



## D4.4

### Discovery of Trust Translation Authorities

Document Identification	
<b>Date</b>	28.05.2018
<b>Status</b>	Final
<b>Version</b>	Version 1.1

<b>Related WP</b>	WP4	<b>Related Deliverable(s)</b>	D3.4, D5.4, D4.1, D4.3
<b>Lead Authors</b>	Javier Presa, Miryam Villegas, Miguel Angel Mateo	<b>Dissemination Level</b>	PU
<b>Lead Participants</b>	ATOS	<b>Contributors</b>	FHG, NLNET, USTUTT
<b>Reviewers</b>	GS, DTU		

This document is issued within the frame and for the purpose of the LIGHT<sup>est</sup> project. LIGHT<sup>est</sup> has received funding from the European Union's Horizon 2020 research and innovation programme under G.A. No 700321.

This document and its content are the property of the *Lightest* Consortium. All rights relevant to this document are determined by the applicable laws. Access to this document does not grant any right or license on the document or its contents. This document or its contents are not to be used or treated in any manner inconsistent with the rights or interests of the *Lightest* Consortium or the Partners detriment and are not to be disclosed externally without prior written consent from the *Lightest* Partners.

Each *Lightest* Partner may use this document in conformity with the *Lightest* Consortium Grant Agreement provisions.

<b>Document name:</b>	Discovery of Trust Translation Authorities	<b>Page:</b>	1 of 32
<b>Dissemination:</b>	PU	<b>Version:</b>	Version 1.1
		<b>Status:</b>	Final



## 1. Executive Summary

The aim of this document is to provide discovery mechanisms for Trust Translation Authorities in order to supply the right information about the trust scheme levels translation lists. The mentioned discovery process is part of the procedure that belongs to the conceptual framework for Trust Scheme Translation presented in the D4.1 [1] where not only the discovery but the publication process is described.

As beginning of this deliverable, Section 5, in order to have a technical perspective of the baseline of the LIGHTest project, it is included the consolidated approach corresponding to the three different technical pillars Publication, Translation and Delegation, but this time is specified for the discovery mechanisms for Trust Translation Authorities. In addition to the publication of trust-related Information, it also includes sections for the discovery of trust declarations as well as the authenticity of Trust Declarations, which are relevant for this deliverable. This approach was published already in the corresponding deliverables D3.3 [2], D4.3 [3], and D5.3 [4]. Therefore, this Section is almost identical to the corresponding sections in those deliverables, in the D3.4 [5] and in the D5.4 [6]. This leads to some duplicity in the deliverables, however it provides for the readers standalone and complete documents.

Following Section 6, it is related to the Discovery of Trust Translation Lists which includes a review of the concept of the Trust Translation Authorities (TTA) but oriented to the explanation of the issues relevant to DNS discovery mechanisms. This includes the three different functionalities: information population, information publication and discovery of the information with special focus on the translation discovery process as presented in the Figure 1

Section 7 describes some examples in order to present the applicability of the consolidated approach to discover Trust Translation Authorities for the different types of Trust Schemes, boolean, ordinal and tuple-based. These examples are aiming to make more understandable the different steps to be happened during the discovery process in different well-known Trust Schemes and how the TTA is been queried in the LIGHTest framework.

For concluding the document, a brief summary can be found where explained the appropriate mechanisms for discovery Trust Translation Authority in the global namespace of domain names.

<b>Document name:</b>	Discovery of Trust Translation Authorities	<b>Page:</b>	2 of 32
<b>Dissemination:</b>	PU	<b>Version:</b>	Version 1.1
		<b>Status:</b>	Final



## 2. Document Information

### 2.1 Contributors

Name	Partner
Miguel Angel Mateo	ATOS
Javier Presa	ATOS
Miryam Villegas	ATOS
Martin Hoffmann	NLNET
Sven Wagner	USTUTT
Heiko Rossnagel	FHG

### 2.2 History

Version	Date	Author	Changes
0.1	21.03.2018	Miguel A. Mateo	Initial draft, Table of Contents
0.2	30.04.2018	Miryam Villegas	Adding the examples for demonstrating the translation queries (eSeal and FIDO). Some sequence diagrams included to explain the TTA within LIGHTest framework.
0.3	03.05.2018	Miryam Villegas	Improving the examples for demonstrating the translation queries.
0.4	07.05.2018	Martin Hoffmann	Added the Consolidated Approach: common chapter to WP3 and WP4.
	07.05.2018	Miryam Villegas	Included a section about DANE resource records in the TTA. Authentication steps added to the examples.
	09.05.2018	Miryam Villegas	Sequence diagram explained. Light customisation changes to the Consolidated Approach.
	09.05.2018	Sven Wagner	Added the example for boolean trust scheme.
0.5	10.05.2018	Miguel Angel Montero	Document re-structuration. Updating sequence diagrams

<b>Document name:</b>	Discovery of Trust Translation Authorities	<b>Page:</b>	3 of 32
<b>Dissemination:</b>	PU	<b>Version:</b>	Version 1.1
		<b>Status:</b>	Final



# Discovery of Trust Translation Authorities



Version	Date	Author	Changes
0.51	11.05.2018	Javier Presa Miryam Villegas	Executive summary. Internal review.
0.6	11.05.2018	Miryam Villegas	References chapter.
0.7	14.05.2018	Sven Wagner	Added the example for another ordinal trust scheme (ISO/IEC 29115).
0.8	14.05.2018	Martin Hoffmann	Added the example for tuple base STORK scheme.
1.0	18.05.2018	Miguel Angel Mateo	Release for internal review
1.1	28.05.2018	Miguel Angel Mateo Javier Presa Miryam Villegas	Final version addressing comments.

<b>Document name:</b>	Discovery of Trust Translation Authorities	<b>Page:</b>	4 of 32
<b>Dissemination:</b>	PU	<b>Version:</b>	Version 1.1
		<b>Status:</b>	Final



## 3. Table of Contents

1. Executive Summary	2
2. Document Information	3
2.1 Contributors .....	3
2.2 History .....	3
3. Table of Contents	5
3.1 Table of Figures.....	6
3.2 Table of Tables.....	6
3.3 Table of Acronyms.....	6
4. Scope of the Deliverable	7
5. A Consolidated Approach to Publishing Trust-related Information in the DNS	8
5.1 Trust Declarations.....	8
5.2 Publication of Trust Declarations .....	9
5.3 Discovering Trust Declarations .....	10
5.4 Authenticity of Trust Declarations .....	11
6. Discovery of Trust Translation Lists	13
6.1 TTA Review.....	13
6.2 About DNS discovery mechanisms.....	16
7. Demonstration for Selected Trust Schemes	19
7.1 Discovery of the translation list for a boolean trust scheme.....	19
7.1.1 eIDAS eTimestamp.....	19
7.2 Discovery of the translation list for an ordinal trust scheme.....	20
7.2.1 eIDAS eSeal .....	20
7.2.2 ISO/IEC 29115 electronic identity .....	22
7.3 Discovery of the translation list for a tuple-based trust scheme.....	24
7.3.1 STORK AQAA (eID+Attributes).....	24
7.3.2 FIDO .....	25
8. Summary and Conclusion	28
9. References	29
10. Project Description	31



## 3.1 Table of Figures

Figure 1: Translation of trust scheme levels ..... 15  
 Figure 2: DNS queries in TTA..... 18

## 3.2 Table of Tables

Table 1: Levels of Trust for eSeal (eIDAS) (Table 16 taken from D4.1)..... 20  
 Table 2: Levels of Assurance for eID of ISO 29115 (Table 2 taken from D4.1)..... 22  
 Table 3: Levels of AQAA for eidentification (STORK 2.0 eID) (Table 22 taken from D4.1) ..... 24

## 3.3 Table of Acronyms

AAID	Authenticator Attestation Identifier (FIDO protocol)
API	Application Programming Interface
AQAA	Attribute QAA scheme
ATV	Automated Trust Verifier
CA	Certification Authority
DANE	DNS-based Authentication of Named Entities
DNS	Domain Name System
DNSSEC	Domain Name System SECURITY extensions
eIDAS	Electronic IDentification And Signature (Regulation (EU) No 910/2014 on electronic identification (eID) and trust services for electronic transactions in the internal market)
FIDO	Fast IDentity Online protocol
FIDO UAF	FIDO Universal Authentication Framework
HTTP(S)	Hypertext Transfer Protocol (Secure)
ID	Identity, identification
LoA	Level of Assurance
NFC	Near-field communication
QAA	Quality of Authentication Assurance
REST	Representational State Transfer (service)
SE	Secure Element
SMIMEA	DNS resource record (RR) is used to associate an end entity certificate or public key with the associated email address, thus forming a "SMIMEA certificate association".
STORK	Secure idenTity acrOss boRders linKed
TEE	Trusted Execution Environment
TPL	Trust Policy Language
TSPA	Trust Scheme Publication Authority
TTA	Trust scheme Translation Authority
TTL	Trust Translation List
U2F	Universal 2nd Factor. It is an open authentication standard that strengthens and simplifies two-factor authentication (2FA) using specialized USB or NFC devices based on similar security technology found in smart cards.
URI	Uniform Resource Identifier
USB	Universal Serial Bus



## 4. Scope of the Deliverable

As previously said in D4.3 [3], the LIGHTest Trust Translation Authority uses a DNSSEC software with an enhancing layer to aim the LIGHTest purposes. Mechanisms to publish the trust translation schemes using the existing ones of the DNS system were defined in that deliverable.

Discovery structures based on DNS are studied in order to be used by verifiers to discover Trust Translation Authorities. The goal is TTA to directly provide information in the form of trust translation lists instead of third parties discovering this information; the TTA is the unique interface to access the trust translation list, meaning nobody could discover any information without asking the TTA.

In relation to publish to a TTA, different approaches were needed for simple and historical Trust Schemes. A suitable sub-domain structure was designed in order to facilitate querying Trust Translation data (see [3]). The design uses the standard DNS delegation mechanisms for support of the subsidiarity principle which is often applied by Trust Scheme Authorities.

Related to the discovery structures, the necessity of finding the appropriate Trust Translation Authority in the global namespace of domain names makes the creation of discovery mechanisms essential. These will include pointers from Trust Scheme Publication Authorities as well as index domains operated by Trust Translation Authorities.

The utilities to construct DNS-based discovery structures for Trust Translation Authorities will be presented and described in this deliverable. One of the tasks that perform these utilities is the design, construction and maintenance of index domains or zones. Another task will be the management of the pointers to the TSPA in order to access to the corresponding TTA.

<b>Document name:</b>	Discovery of Trust Translation Authorities	<b>Page:</b>	7 of 32
<b>Dissemination:</b>	PU	<b>Version:</b>	Version 1.1
		<b>Status:</b>	Final



## 5. A Consolidated Approach to Publishing Trust-related Information in the DNS

Regarding the Discovery functionality in LIGHTest, it also has many similarities within WP3, WP4 and WP5. As in the case of Publication functionality and deliverables D3.3 [7], D4.3 [3] and D5.3 [4] a consolidated approach has been elaborated and is introduced in the three deliverables in charge of describing the Discovery of Publication, Translation and Delegation of Trust Schemes.

The following text in this chapter will be also added as it is or with small modifications in D3.4 [5] and D5.4 [6].

The trust framework created by LIGHTest verifies the trustworthiness of an electronic transaction by attempting to establish a chain of trust from a set of pre-configured, well-known trust sources to this transaction. The links in this chain are assurances that the trust into a source can be extended to another entity. This section looks at the basic properties these assurances exhibit independently of their concrete contents and introduces an underlying, fundamental framework.

Within the LIGHTest project, three working packages look at different aspects of such assurances. WP3 examines the most basic form where a trust source known as a trust scheme declares that some issuer of trust services, such as certificates or time stamps, conforms to the conditions and rules set out by the scheme and thus extend trust placed into it onto this trust service. WP4 assesses the relationship between multiple trust schemes allowing a trust scheme or some other trusted entity to declare if and how trust into one scheme extends to trust into another scheme. WP5, finally, looks into how individual entities can empower other entities to act on their behalf– extending trust into themselves onto that other entity within certain well-defined limits.

### 5.1 Trust Declarations

However different they may appear, each of these aspects follows a similar pattern: some entity makes a *trust declaration* stating that trust into a certain entity extends to another entity, possibly providing conditions and limits of such an extension of trust. To simplify further discussion, it will be helpful to label the three entities involved in the process. The entity issuing the declaration shall be the **originator**, the entity that is already trusted is the **source** and the entity trusted as a result of the declaration is the **target**.

Within the aspects discussed as part of the LIGHTest project, in many cases the originator of a declaration is identical to the source. This is certainly true for trust membership publication in WP3, where the trust scheme itself declares which trust services are members. In the aspects of the other two work packages, similarly originator and source are most often identical. However, there may be use cases where this is not the case. For instance, a third party may declare a trust translation independently of the trust schemes that are source or target of this declaration. Similarly, a third party may declare a trust delegation. For instance, business registries often

<b>Document name:</b>	Discovery of Trust Translation Authorities	<b>Page:</b>	8 of 32
<b>Dissemination:</b>	PU	<b>Version:</b>	Version 1.1
		<b>Status:</b>	Final





provide information about individuals that are allowed to sign on behalf of a company. Such information can be modeled as third-party trust delegation.

This definition specifies a single declaration to extend trust from exactly one source to exactly one target. In practice, the document formats used often contain multiple declarations according to this definition. For instance, the trusted lists defined in ETSI TS 119 612 [8] used for the declarations in WP3 contain a list of all the targets a trust scheme as a source wishes to extend trust to. This published form of declarations shall be called *trust declaration documents*. At least in principle each such document can contain one or many declarations with any number of sources and targets.

For a trust declaration to be considered when building the chain of trust during validation of an electronic transaction, the declaration itself needs to be trusted. If a declaration document is treated like any other electronic transaction, this trust can in turn be established through verification using the LIGHTest framework. That is, there needs to be a chain of trust from pre-configured trust sources to the originator of the declaration document for the particular aspect of the declaration in question.

Note that in most cases where the originator of the declaration is identical to the source of the declaration, this chain exists implicitly and no extra checks are needed. It may, however, be possible that conditions for trusting the source as such and trusting declarations made by it are different and thus need to be verified independently.

## 5.2 Publication of Trust Declarations

When using trust declarations for verifying an electronic transaction, a validator needs to construct a chain of trust declarations leading from any of the trusted entities to those entities appearing in the transaction. It does so by recursively finding and adding applicable trust declarations that have an originator that is either a trusted entity or the target of the declaration is already part of the chain. In order to do this in an unaided, automatic way, the validator needs a way to gain access to all declarations that are potentially usable in this process.

There are two fundamental strategies for the verifier to find declarations when they are needed: a declaration could either be actively supplied as part of the input or configuration or it is left to the verifier itself to discover it.

The prime example for the former case is that a declaration is provided as part of the electronic transaction to be verified. This is particularly useful if the creator of the transaction is aware that the declaration is necessary for verification. For instance, if an entity signs a document in their function as a proxy, the transaction will only ever verify if the declaration of trust delegation— the mandate— is known to the verifier. The proxy may very well include the mandate in the transaction right away.

For declarations that are potentially applicable to a large amount of transactions or if the sender of the transaction does not know the trusted entities the receiver will base their verification on, such a strategy is not very practical. Instead, it is better to make the declarations available

<b>Document name:</b>	Discovery of Trust Translation Authorities	<b>Page:</b>	9 of 32
<b>Dissemination:</b>	PU	<b>Version:</b>	Version 1.1
		<b>Status:</b>	Final



publicly and provide means for a verifier to discover how and where it can retrieve them. The verifier can then decide itself which declarations it needs and try and find them as needed.

While the most likely method for publishing currently is the Hypertext Transfer Protocol (HTTP), other protocols may become available in the future. An extensible standard exists to describe both the method used for accessing a resource and all necessary parameters for a successful retrieval in the form of a Uniform Resource Identifier (URI). It encodes all information into a single string which can be easily stored or transmitted.

## 5.3 Discovering Trust Declarations

In order to build a chain of trust from published declarations, the verifier needs to be able to discover their existence. Given a transaction and a set of already trusted entities, this chain can be built from two sides: either the verifier starts with a trusted entity, tries to discover all the declarations that have this entity as their source, and repeats this process until it arrives at a declaration that includes the transactions as its target or runs out of declarations to apply. Alternatively, it can start with the transaction, attempts to discover all declarations that have the transaction as their target, and continues by recursively trying to find declarations that have the sources of already discovered declarations as their target until it arrives at an already trusted entity as a declaration's source or, again, runs out of declarations.

In both cases, the verifier needs a mechanism to search for the URI of a declaration based on a given entity. Such a mechanism will have to take some information that identifies that entity as input. Given that entities are typically identified by X.509 certificates, it should be possible to include such identifying information in the entity's certificate. One of the possible options is to use a domain name as the entity's identifier. This has the advantage that the name can be used directly as a search input for the DNS, allowing using the DNS as a global, highly available, distributed, and independently managed data store.

A domain name can be stored in a certificate either in the subject alternative name or issuer alternative name extensions. The subject alternative name indicates the domain name identifying the entity using the certificate while the issuer alternative name identifies the entity having issued the certificate.

Using this domain name as input, the URIs to the declarations for differing aspects should be stored in the DNS. As the DNS uses record types to distinguish between different types of information stored for a domain name, one option is to register an individual record type for each aspect of trust declarations. However, this would clutter the space of types. As the data stored is the same in every case – a URI – an alternative approach can be used whereby the aspect is encoded as a prefix to the domain name. This is already used for instance with the SRV record type for discovering the host name and port where a certain networking service is available for a domain. As such services sometimes are available over different transport protocols, a two-layer prefix of the form `_service._protocol` is used where the latter describes the transport protocol to be used and the former the networking service.

<b>Document name:</b>	Discovery of Trust Translation Authorities	<b>Page:</b>	10 of 32
<b>Dissemination:</b>	PU	<b>Version:</b>	Version 1.1
		<b>Status:</b>	Final



In keeping with this concept, LIGHTest proposes to use the DNS for providing pointers to all kinds of declarations via a pair of prefixes of the form *\_aspect.\_application*. Here, the type of application for which a declaration is published is the second part of the prefix while the first part defines the particular aspect within that application. For LIGHTest itself, the application is, of course, trust-related declarations, which is identified via using the literal label *\_trust* as the second part. Each of the aspects of trust declarations defines its own label to be used as the first part.

Under the name constructed by concatenating the prefix with the entity's own identifying domain name, the entity can now publish pointers to trust declarations of the corresponding aspect that relate to it. It can use the already existing URI resource record type for this purpose. This type is defined in section 4 of RFC 7553 [9]. Its record data contains exactly one URI. While section 5 of the RFC describes a different use of the record, this use is limited by a different set of prefixes, allowing its reuse for declaration publication based on the prefixes defined above.

If the entity in question is not the originator of a declaration it may not control the URI under which the declaration is published. In this case, it may be beneficial to only point to the originating entity rather than burden itself with tracking whether the originators URI has changed. This can be done easily if the originator is an entity identified by a domain name, too. In this case, instead of publishing URI resource records under the domain name prefixed by the declaration aspect, the entity will publish PTR resource records. Such record types, part of the original DNS specification in RFC 1035 [10], contain another domain name as their record data. If such records are present, they instruct the verifier to continue discovery for declaration at the entity identified by these domain names. Note that as these names in the PTR record's data refer to the specific application and aspect and as such are already prefixed with the correct declaration prefix. While not used in the LIGHTest framework as yet, this would allow to provide references between different aspects.<sup>1</sup>

## 5.4 Authenticity of Trust Declarations

If a verifier retrieves a declaration from somewhere in the network, it needs to make sure that the data it received is indeed the declaration made by the originator. Since the URI resource records are stored under the domain name identifying the originator, it is reasonable to assume they are authentic if DNSSEC validation succeeds, guaranteeing that the URI is indeed the one intended by the originator.

In the next step, where the verifier contacts the server indicated in the URI, it needs to ensure that it communicates with the correct server. When using encrypted transport via the TLS protocol, the server will identify itself via a certificate. In order to deal with shortcomings of the

---

<sup>1</sup> The initial version of the Consolidated Framework as presented in deliverable D3.3 proposed to let the PTR records refer to the domain name identifying the entity. After discussion, it was decided that the updated approach presented here is both more correct as it points to the exact name where discovery continues and, as mentioned, more flexible as a generic means of discovery.

<b>Document name:</b>	Discovery of Trust Translation Authorities	<b>Page:</b>	11 of 32
<b>Dissemination:</b>	PU	<b>Version:</b>	Version 1.1
		<b>Status:</b>	Final



traditional method of certificate verification, a protocol called DNS-based Authentication of Named Entities, or DANE for short, allows a server operator to publish information about the certificates used in DNS.

Since, however, the server is not necessarily operated under authority of the originator of the declaration – for instance because the declarations are hosted by a third party that provides better availability, this does not guarantee that the declaration received is indeed the one that the originator intended.

This final link can be provided if the declaration itself is a signed document. The originator can then publish the certificates that it uses for signing declarations of a certain aspect using a slightly adapted version of the DANE protocol.

To do so, it adds SMIMEA resource records under the same domain name it placed the URI records pointing to the declarations. These records define conditions a certificate has to fulfill to be accepted. By placing these records, the originator declares that all documents retrieved via the pointers have to verify considering these conditions. See D2.7 [11] for more detail.

<b>Document name:</b>	Discovery of Trust Translation Authorities	<b>Page:</b>	12 of 32
<b>Dissemination:</b>	PU	<b>Version:</b>	Version 1.1
		<b>Status:</b>	Final



## 6. Discovery of Trust Translation Lists

Discovery for the Trust Translation Authority is the function that allows users, in LIGHTest project environment it means the ATV, to look for and retrieve information regarding agreements of equivalence between two Trust Schemas.

### 6.1 TTA Review

As it has been introduced in D4.1 [1] that “the TTA provides the translation between trust schemes, by enabling trust scheme providers to indicate which trust schemes are recognized as trustworthy by their trust scheme. Any Trust Scheme Provider can hereby negotiate with other Trust Scheme Provider on whether their schemes trust each other, and in what way”. This definition was later extended in D4.3 [3] to define it as the unidirectional translation between two Levels of Assurance of two different Trust Schemes.

This task is divided into three different functionalities: information population, information publication and discovery of the information.

Population is understood as the process required for feeding the TTA system with valid information. To this end it will be provided a public REST API. This information is stored and translated to the formats required by the other two functionalities. Publication is understood as the process of providing the information in committed formats, while discovery is the functionality that let an operator, in this case the ATV to find the information it needs.

The two first functionalities (public API and publication) are implemented by the Trust Translation List Provider, which receive the information about Trust Scheme Translations, and produce the required data in order to let a user discover and retrieve the information.

The input information consists of Trust Translation Declaration as expression of the agreements among Trust Entities. This is, when a Trust provider declares that there is a specific equivalence between one specific trust level of a service and another trust level of a service published by another trust entity. This relation of equivalence should be provided to the TTA in the form of a contract expressing the specific circumstances under which the agreement is valid.

Two different types of information results from the work of the Trust Translation provider; a file containing the equivalence and its characteristics, and the structure of data required to let ATV discover this file.

In the following sequence diagram (Figure 1), the process of translating a trust level is shown in a high level view:

- Steps 1-3: An electronic business transaction (eBusiness, or eTransaction) comes to the Verifier to be analysed whether it is valid or not according to some policies defined previously. The Verifier, with the ATV, tries to match the policies with the incoming eTransaction. Thus, a request for the trustworthy of the eTransaction is submitted to the ATV.

<b>Document name:</b>	Discovery of Trust Translation Authorities	<b>Page:</b>	13 of 32
<b>Dissemination:</b>	PU	<b>Version:</b>	Version 1.1
		<b>Status:</b>	Final



- Steps 4-8: The ATV analyses the transaction and asks the TSPA for the related trust scheme. The TSPA returns the information with the trust scheme corresponding to that transaction. The ATV can compare the information just provided by the TSPA with the trust policy. Let's suppose ATV, as result of the validation, does not find matching between the trust scheme of the eTransaction and the local policies, so, ATV takes the decision of looking for the translations of the trust level (the local one or the incoming one), in order to get any trust level equivalence.
- Steps 9-13: Firstly, the ATV needs to know which trust scheme the incoming trust level belongs to. If such information is not locally available, the ATV requests it to the TSPA, which provides it: the domain name to build the right query to the TTA, is an information that is maintained by the TSPA. The ATV can build the related query with the level name plus the trust scheme name, in order to send the query to the TTA. The corresponding TTA will return the list of the trust levels equivalents to the one requested.
- Steps 14-15: The ATV, which has just received that list, analyses it and gets a result of trustworthy (true or false). This answer is sent to the Verifier.
- Step 16: The Verifier has received the trustworthy answer, and it can be processed as the use case is defined.

<b>Document name:</b>	Discovery of Trust Translation Authorities	<b>Page:</b>	14 of 32
<b>Dissemination:</b>	PU	<b>Version:</b>	Version 1.1
		<b>Status:</b>	Final



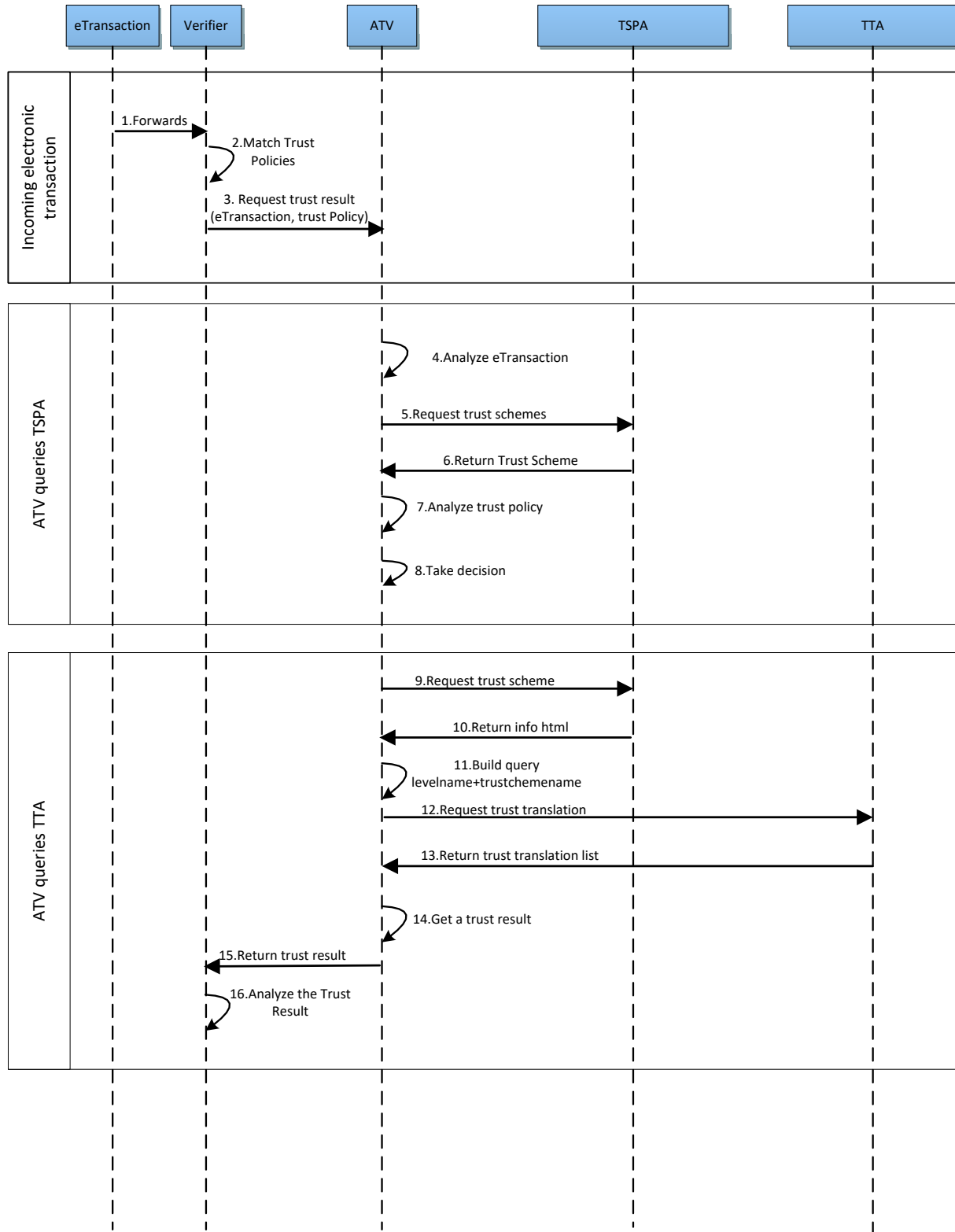


Figure 1: Translation of trust scheme levels



## 6.2 About DNS discovery mechanisms

Whenever the ATV needs to request a translation for a given level of a trust scheme, the construction of the request is linked to the information it has received from TSPA as it is this authority where trust scheme, their names and characteristics are hosted.

We should say at this point that TTA considers trust translations are levelled at Trust Scheme Level of Assurance. For instance, in the case of eSignature service into eIDAS, TTA considers three cases corresponding to its LoA: Basic, Qualified and Advanced. This is quick to characterise in the case of Boolean where there is only one possibility of name, or in ordinal schemes in which levels are defined a priori.

In the case of Tuple Base Trust Schemes there is no such a specification and LoA depends on other factors that are expressed by means of the values of attributes. To be able to provide translations for this kind of schemas there are two possibilities:

- To set a translation at a higher level, this is, at trust scheme level (*eSignature.Fido = eSignature.eIDAS*).
- To specify at the TSPA a level name for each set of attribute and their values. This name has to be use then when creating the translation declaration at the TTA.

In the former case the responsibility of matching attribute values with a Level of Assurance relies on the ATV, this option impoverishes the concept of LIGHTest as it delegates the responsibility to the policies defined at ATV which are independent of translation agreements signed.

In the latter case, TTA needs TSPA to provide or host a name of each set of values for the attributes.

According to the consolidated approach in Section 5, pointers to declarations, are formed as a composition of two prefixes plus the domain name identifying then entity or Trust Scheme Level. The prefixes for the TTA are set as “*\_translate*” for the aspect and “*\_trust*” for the application. Resulting pointers will have a form like:

*\_translate.\_trust.qualified.esignature.eidas.eu*

By means of URI resource record type, these pointers are published into the DNS to address where to find the corresponding translation declarations, which are signed trust translations list documents.

These documents contain unidirectional equivalences in two different formats. XML formats which is a list of the Trust Scheme + Level as registered in TSPA to let ATV query the TSPA about this scheme, and TPL format that contains not only the name of the scheme but also a description in TPL language to indicate how ATV has to interpret the equivalence. This format is more expressive in order to evaluate equivalence in the case of tuple base schemes.

Trust translation list documents, XML or TPL formats, are signed by the TTA with X.509 certificates. Also, in order to let ATV ensure these documents are valid, a set of constraints to

<b>Document name:</b>	Discovery of Trust Translation Authorities	<b>Page:</b>	16 of 32
<b>Dissemination:</b>	PU	<b>Version:</b>	Version 1.1
		<b>Status:</b>	Final





validate certificates is set in the DNS with SMIMEA resource record. Performed by setting constraints to the accepted certificates. How to set these constraints is explained in section 9.1.2 of D3.4 [5].

According to the whole process depicted in Figure 1, section “ATV queries TSPA” corresponds to the Figure 2 of D3.4 where ATV queries and verifies the Trust Scheme the entity claims to be under. If the ATV determines that according to the policy applied the result of the verification is not valid, the verifier can then query TTA in order to look for an available translation of the trust scheme to one for which the verification may succeed.

The Figure 2 below shows an example of the process of querying Trust Translations Lists to the TTA. This process corresponds to the section “ATV queries TTA” of Figure 1.

1. ATV queries with the Scheme name to the DNS for the translation provider. It should be mentioned that the ATV, at this point, should have all the information regarding the scheme name plus trust level as described above. This information is derived from information received from the TSPA: step 6 in Figure 1 or a new request if needed.
2. The DNS delivers the record for the Trust Translation Provider which contains the pointer to the Trust Translation List.
3. The ATV requests the TTA provider with the Scheme Name + Level in the form of domain name as specified above.
4. TTA provider checks if there is an available list of translation for the given name. If found, the provider delivers the requested signed list document.
5. The ATV queries the DNS with the given name in order to retrieve the certificate constraints.
6. DNS provides certificate constraints.
7. ATV verifies that the document is signed with a valid certificate.

<b>Document name:</b>	Discovery of Trust Translation Authorities	<b>Page:</b>	17 of 32
<b>Dissemination:</b>	PU	<b>Version:</b>	Version 1.1
		<b>Status:</b>	Final



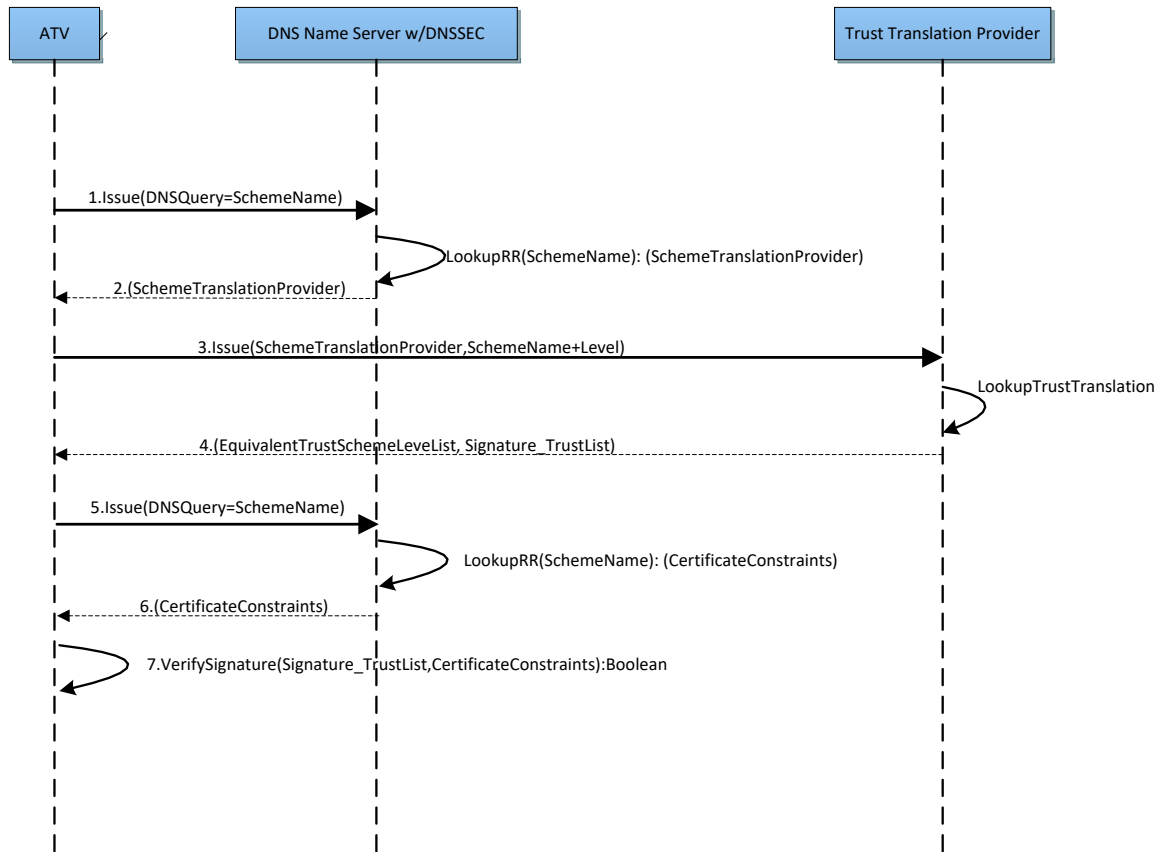


Figure 2: DNS queries in TTA



## 7. Demonstration for Selected Trust Schemes

The following examples are proofs of concept of the discovery mechanisms explained in the previous chapters. All the translation lists given here are the result of scenarios with *hypothetical* bilateral agreements between the authorities responsible for each trust scheme.

The trust schemes shown in this chapter have been analysed in previous deliverables (see D3.1 [12] and D4.1 [1] for more information.)

In order to demonstrate the authenticity of the trust declarations, please refer to section 5.4

### 7.1 Discovery of the translation list for a boolean trust scheme

#### 7.1.1 eIDAS eTimestamp

The electronic Timestamp trust service in the eIDAS trust scheme [13] is an example for boolean trust schemes with a qualified timestamp or not. In the next paragraphs it can be seen how the TTA is queried in the LIGHTest framework for boolean trust schemes.

Note the proper Trust Translation Authority has been selected (the eIDAS as the trust scheme), and then the trust service (electronic timestamp). The ATV builds the complete domain name to be translated, i.e., to query the TTA for such translation. Next, the ATV sends a query to the TTA to know about equivalent other trust schemes. The verifier needs to check this claim by locating the trust translation declaration:

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

;; ANSWER SECTION: from the TTA
_translate._trust.etimesamp.eidas.eu. IN URI
                        https://lightest.eu/ttl_qualifiedTimestampEidas1.tpl
                        ...
_translate._trust.etimesamp.eidas.eu. IN URI
                        https://lightest.eu/ttl_qualifiedTimestampEidasN.tpl

_translate._trust.etimesamp.eidas.eu. IN URI
                        https://lightest.eu/ttl_qualifiedTimestampEidas1.xml
                        ...
_translate._trust.etimesamp.eidas.eu. IN URI
                        https://lightest.eu/ttl_qualifiedTimestampEidasN.xml
```

The equivalent trust levels are given by the files returned. Note that a trust translation for a given trust level can be characterized by more than one file: the TTA module provides with a file for each recognized trust level.

Besides, notice there are two kinds of files, TPL files and XML files, depending on the client to manage the answers.

<b>Document name:</b>	Discovery of Trust Translation Authorities	<b>Page:</b>	19 of 32
<b>Dissemination:</b>	PU	<b>Version:</b>	Version 1.1
		<b>Status:</b>	Final



Since the content of those files is signed, it will also check whether the certificate used for signing the files is valid.

```
;; QUESTION SECTION: Verifying authenticity
;_translate._trust.etimestamp.eidas.eu. IN SMIMEA

;; ANSWER SECTION:
_translate._trust.etimestamp.eidas.eu. IN SMIMEA <SMIMEA record data>
```

If validation succeeds, the verifier is sure that the list of the translations obtained is authentic: the publishing originator has been authenticated.

## 7.2 Discovery of the translation list for an ordinal trust scheme

### 7.2.1 eIDAS eSeal

The electronic eSeal trust service in the eIDAS trust scheme [13] provides two levels of trust, namely, Advanced and Qualified as described in the next table. In the next paragraphs it can be seen how the LIGHTest framework is queried for getting the list of all trust levels equivalent to the Qualified level of the eIDAS eSeal.

eSEAL	Uniquely linked to the creator of the seal	Identification process	Creation data with high level of confidence	Changes on data sealed are detectable	Created by a qualified eSeal creation device	Based on a qualified certificate for eSeal
Advanced	✓	✓	✓	✓	Any of these not fulfilled	
Qualified	✓	✓	✓	✓	✓	✓

Table 1: Levels of Trust for eSeal (eIDAS) (Table 16 taken from D4.1)

First, the names of the levels just published by the TSPA have to be retrieved from the TSPA, in order to build the right domain name for asking for the translation. Note the proper Trust Translation Authority has been selected (the eIDAS as the trust scheme), and then the trust service (electronic seal):

```
;; QUESTION SECTION: Client/ATV to the TSPA
;_scheme._trust.eseal.eidas.eu. IN URI

;; ANSWER SECTION: from the TSPA
_scheme._trust.eseal.eidas.eu. IN URI
\ https://lightest.eu/eIDAS_eSeal.xml
```



The TSPA publishes the name of the trust level corresponding to the right domain name. The **ATV** expects this behaviour in order to be able to build the complete domain name of the level of trust to be translated, i.e., to query the TTA for such translation.

Once the ATV has built the right domain name for the *qualified* level, for example, a query is sent to the TTA to know about its equivalent levels in other trust schemes. The verifier needs to check this claim by locating the trust translation declaration:

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

;; ANSWER SECTION: from the TTA
_translate._trust.qualified.esel.eidas.eu. IN URI
    https://lightest.eu/ttl_qualifiedSealEidas1.tpl
    ...
_translate._trust.qualified.esel.eidas.eu. IN URI
    https://lightest.eu/ttl_qualifiedSealEidasN.tpl

_translate._trust.qualified.esel.eidas.eu. IN URI
    https://lightest.eu/ttl_qualifiedSealEidas1.xml
    ...
_translate._trust.qualified.esel.eidas.eu. IN URI
    https://lightest.eu/ttl_qualifiedSealEidasN.xml
```

The equivalent levels are given by several files returned. Note that a trust translation level for a given trust level can be characterized by more than one file: the TTA module provides with a file for each recognized trust level.

Besides, notice there are two kinds of files, TPL files and XML files, depending on the client to manage the answers. For example, the ATV is expecting the results in TPL format. Other clients could get the answers in XML format.

Since the content of those files is signed, it will also check whether the certificate used for signing the files is valid according with the content of SMIMEA resource record.

```
;; QUESTION SECTION: Verifying authenticity
;_translate._trust.qualified.esel.eidas.eu. IN SMIMEA

;; ANSWER SECTION:
_translate._trust.qualified.esel.eidas.eu. IN SMIMEA <SMIMEA record
    data>
```

If validation succeeds, the verifier is sure that the list of the translations obtained is authentic: the publishing originator has been authenticated.

<b>Document name:</b>	Discovery of Trust Translation Authorities	<b>Page:</b>	21 of 32
<b>Dissemination:</b>	PU	<b>Version:</b>	Version 1.1
		<b>Status:</b>	Final



## 7.2.2 ISO/IEC 29115 electronic identity

With the ISO/IEC 29115 standard “Information Technology – Security techniques – Entity authentication assurance” [14] another example for the discovery of the translation list for an ordinal trust scheme is shown. The framework consists of 3 technical phases (enrolment phase, credential management phase, entity authentication phase). For each phase, 4 levels of assurance for entity authentication are defined, which determine the degree of confidence in the process (see Figure 3, which contains Table 2 from D4.1 [1] as an example of LoA). In the next paragraphs it can be seen how the LIGHTest framework is queried for getting the list of all trust levels equivalent to the LoA4 level of ISO/IEC 29115.

IDENTITY	Enrolment	Credential management	Authentication
LoA 4	✓	✓	✓
LoA 3	If any of the above requirements is LoA 3 and none is LoA 2 or 1		
LoA 2	If any of the above requirements is LoA 2 and none is LoA 1		
LoA 1	If any of the above requirements is LoA 1		

Table 2: Levels of Assurance for eID of ISO 29115 (Table 2 taken from D4.1)

First, the names of the levels just published by the TSPA have to be retrieved from the TSPA, in order to build the right domain name for asking for the translation. Note the proper Trust Translation Authority has been selected (the ISO/IEC 29115 as the trust scheme), and then the trust service (electronic ID):

```
;; QUESTION SECTION: Client/ATV to the TSPA
;_scheme._trust.eid.iso29115.org. IN URI
;; ANSWER SECTION: from the TSPA
_scheme._trust.eid.iso29115.org. IN URI
    \ https://lightest.eu/iso29115_eID.xml
```

The TSPA publishes the name of the trust level corresponding to the right domain name. The **ATV** expects this behaviour in order to be able to build the complete domain name of the level of trust to be translated, i.e., to query the TTA for such translation.

Once the ATV has built the right domain name for the *LoA4* level, for example, a query is sent to the TTA to know about its equivalent levels in other trust schemes. The verifier needs to check this claim by locating the trust translation declaration:

```
;; QUESTION SECTION: Client/ATV to the TTA
;_translate._trust.loa4.eid.iso29115.org. IN URI
;; ANSWER SECTION: from the TTA
_translate._trust.loa4.eid.iso29115.org. IN URI
    https://lightest.eu/ttl_LoA4iso29115_1.tpl
_translate._trust.loa4.eid.iso29115.org. IN URI
    https://lightest.eu/ttl_LoA4iso29115_2.tpl
```

<b>Document name:</b>	Discovery of Trust Translation Authorities	<b>Page:</b>	22 of 32
<b>Dissemination:</b>	PU	<b>Version:</b>	Version 1.1
		<b>Status:</b>	Final



```
...
_translate._trust.loa4.eid.iso29115.org.  IN  URI
      https://lightest.eu/ttl_ LoA4iso29115_N.tpl

_translate._trust.loa4.eid.iso29115.org.  IN  URI
      https://lightest.eu/ttl_ LoA4iso291151.xml
_translate._trust.loa4.eid.iso29115.org.  IN  URI
      https://lightest.eu/ttl_ LoA4iso29115_2.xml
...
_translate._trust.loa4.eid.iso29115.org.  IN  URI
      https://lightest.eu/ttl_ LoA4iso29115N.xml
```

The equivalent levels are given by several files returned. Note that a trust translation level for a given trust level can be characterized by more than one file: the TTA module provides with a file for each recognized trust level.

Besides, notice there are two kinds of files, TPL files and XML files, depending on the client to manage the answers. For example, the ATV is expecting the results in TPL format. Other clients could get the answers in XML format.

Since the content of those files is signed, it will also check whether the certificate used for signing the files is valid according with the content of SMIMEA resource record.

```
;; QUESTION SECTION: Verifying authenticity

;_translate._trust.LoA4.eID.iso29115.org.  IN  SMIMEA

;; ANSWER SECTION:

_translate._trust.LoA4.eID.iso29115.org.  IN  SMIMEA <SMIMEA record data>
```

If validation succeeds, the verifier is sure that the list of the translations obtained is authentic: the publishing originator has been authenticated.

<b>Document name:</b>	Discovery of Trust Translation Authorities	<b>Page:</b>	23 of 32
<b>Dissemination:</b>	PU	<b>Version:</b>	Version 1.1
		<b>Status:</b>	Final



## 7.3 Discovery of the translation list for a tuple-based trust scheme

In the previous examples with boolean and ordinal trust schemes, the identifiers for each trust level were easily interpreted, almost the expected identifiers. In the case of the tuple-based trust schemes, the TSPA will have to fix a name for each list of tuples (attribute name, attribute value).

### 7.3.1 STORK AQAA (eID+Attributes)

STORK AQAA assigns an attribute quality assurance level to a group of attributes provides as part of an electronic ID. These assurance levels are derived from both the quality assurance level of the eID itself as well as the maximum of the quality assurance levels of each of the attributes in the group. Table 22 of D4.1 [1] provides an overview of the AQAA levels and how they relate to the eID and attributes.

STORK 2.0 attrib. assertion AQAA level	Maximum eID quality (QAA level)	Maximum attribute quality (AQAA level)
4	✓	✓
3	If any of the above is level 3 and none is 2 or 1.	
2	If any of the above is level 2 and none is 1.	
1	If any of the above is level 1.	

Table 3: Levels of AQAA for eidentification (STORK 2.0 eID) (Table 22 taken from D4.1)

A trust scheme can provide assurance levels for one or more groups of attributes. It publishes these as part of its trust scheme publication via its TSPA.

For each of these groups, the trust scheme can also provide a list of equivalent schemes for the various quality levels, i.e., trust schemes that provide the same set of attributes with that quality of assurance for their electronic IDs. In order for ATV to discover these trust translation lists, the TSPA provides a symbolic name for each group it defines as part of its trust scheme status list.

When attempting to discover the trust translation lists, the ATV constructs a domain name from the trust scheme's name, the symbolic name of the group in question, and the well-defined prefix for trust translation documents. Under this name, the TTA will publish a pointer to the trust translation list for the group in the form of a series URI resource records.

For example, let's assume a fictional eID scheme *eid.lightest.eu* defines an attribute group *name-and-day-of-birth* that contains the attributes for the name and day of birth of the holder of the eID. To find schemes that provide an equivalent assurance, the ATV wants to discover the

<b>Document name:</b>	Discovery of Trust Translation Authorities	<b>Page:</b>	24 of 32
<b>Dissemination:</b>	PU	<b>Version:</b>	Version 1.1
		<b>Status:</b>	Final





trust translation list for this assurance group. It constructs the domain name to query as `_translate._trust.name-and-day-of-birth.lightest.eu` and queries for URI records:

```
;; QUESTION SECTION: Client/ATV to the TSPA
;_translate._trust.name-and-day-of-birth.eid.lightes.eu. IN URI

;; ANSWER SECTION: from the TSPA
_translate._trust.name-and-day-of-birth.eid.lightes.eu. IN URI \
    https://eid.lightest.eu/name-and-day-of-birth/ttl-1.xml
_translate._trust.name-and-day-of-birth.eid.lightes.eu. IN URI \
    https://eid.lightest.eu/name-and-day-of-birth/ttl-1.tpl
_translate._trust.name-and-day-of-birth.eid.lightes.eu. IN URI \
    https://eid.lightest.eu/name-and-day-of-birth/ttl-2.xml
_translate._trust.name-and-day-of-birth.eid.lightes.eu. IN URI \
    https://eid.lightest.eu/name-and-day-of-birth/ttl-2.tpl
```

In the response, the TTA provides four trust translation lists. These are in two different formats, XML and TPL, to leave the ATV the choice of the format it prefers.

For each format, the TTA provides one list each for the two AQAA levels 1 and 2. These documents contain the equivalent schemes that provide eIDs with attributes for name and date of birth with an AQAA level of 1 and 2, respectively.

In addition to these URI records, the TTA also publishes an SMIMEA resource record under the same name. These records describe requirements for the certificates used for signing the trust translation lists.

```
;; QUESTION SECTION: Client/ATV to the TSPA
;_translate._trust.name-and-day-of-birth.eid.lightes.eu. IN SMIMEA

;; ANSWER SECTION: from the TSPA
_translate._trust.name-and-day-of-birth.eid.lightes.eu. IN SMIMEA ...
```

When verifying the signatures of all of the trust translation lists referenced by the URI records, the ATV must apply the requirements described in the record to determine, whether the certificate used for signing the documents is authorised according with the content of SMIMEA resource record. This process works in the same way as with the other examples provided.

## 7.3.2 FIDO

FIDO authenticators may be different among them. The trust that an application puts on an authentication could depend on the authenticator used. To enable the verification of such authenticators, an attestation process takes place based on certain information that defines the level of trust of the authenticator.

FIDO UAF authenticators are described by a tuple-based trust scheme by means of these attributes, among others (see [15]):

- Authenticator Version
- Authenticator ID (AAID)
- Attestation Certificate
- User verification method such as fingerprint biometrics

<b>Document name:</b>	Discovery of Trust Translation Authorities	<b>Page:</b>	25 of 32
<b>Dissemination:</b>	PU	<b>Version:</b>	Version 1.1
		<b>Status:</b>	Final



- Whether keys are protected by Trusted Execution Environment (TEE) or Secure Element (SE)
- Whether biometrics are protected by TEE

On the other hand, FIDO U2F authenticators have their own attributes for establishing the levels of trust too [15].

A trust translation process on a certain level of trust of FIDO UAF authenticator is shown following, as an example of a tuple-based trust level being translated. For simplicity, the example has been extracted from [15].

The trust level to be translated is defined by the following tuples:

- authenticatorVersion 2.
- Fingerprint based user verification allowing up to 5 registered fingers, with false acceptance rate of 0.002% and rate limiting attempts for 30 seconds after 5 false trials.
- Authenticator is embedded with the FIDO User device.
- The authentication keys are protected by TEE and are restricted to sign valid FIDO sign assertions only.
- The (fingerprint) matcher is implemented in TEE.
- The Transaction Confirmation Display is implemented in a TEE.
- The Transaction Confirmation Display supports display of "image/png" objects only.
- Display has a width of 320 and a height of 480 pixels. A bit depth of 16 bits per pixel offering True Color (=Color Type 2). The zlib compression method (0). It doesn't support filtering (i.e. filter type of=0) and no interlacing support (interlace method=0).
- The Authenticator can act as first factor or as second factor, i.e. isSecondFactorOnly = false.
- It supports the "UAFV1TLV" assertion scheme.
- It uses the ALG\_SIGN\_SECP256R1\_ECDSA\_SHA256\_RAW authentication algorithm.
- It uses the ALG\_KEY\_ECC\_X962\_RAW public key format (0x100=256 decimal).
- It only implements the TAG\_ATTESTATION\_BASIC\_FULL method (0x3E07=15879 decimal).
- It implements UAF protocol version (upv) 1.0 and 1.1.

Such a tuple-based trust level can be referenced with a name given by the TSPA, for example *fido2fingerprint*. This name is going to be retrieved at first place by the ATV:

The names of the levels just published by the TSPA have to be retrieved from the TSPA, in order to build the right domain name for asking for the translation. Note the proper Trust Translation Authority has been selected (the FIDOUAF as the trust scheme, [16]), and then the trust service (attestation):

<b>Document name:</b>	Discovery of Trust Translation Authorities	<b>Page:</b>	26 of 32
<b>Dissemination:</b>	PU	<b>Version:</b>	Version 1.1
		<b>Status:</b>	Final



```
;; QUESTION SECTION: Client/ATV to the TSPA
_scheme._trust.attestation.fidouaf.eu. IN URI

;; ANSWER SECTION: from the TSPA
_scheme._trust.attestation.fidouaf.eu. IN URI
 \ https://lightest.eu/FIDOUAF_attestation.xml
```

The TSPA publishes the name of the trust level corresponding to the right domain name. The **ATV** expects this behaviour in order to be able to build the complete domain name of the level of trust to be translated, i.e., to query the TTA for such translation.

Once the ATV has built the right domain name for the *fido2fingerprint* level, for example, a query is sent to the TTA to know about its equivalent levels in other trust schemes. The verifier needs to check this claim by locating the trust translation declaration:

```
;; QUESTION SECTION: Client/ATV to the TTA
_translate._trust.fido2fingerprint.attestation.fidouaf.eu. IN URI

;; ANSWER SECTION: from the TTA
_translate._trust.fido2fingerprint.attestation.fidouaf.eu. IN URI
 https://lightest.eu/ttl_fido2fingerprintAttestation1.tpl
...
_translate._trust.fido2fingerprint.attestation.fidouaf.eu. IN URI
 https://lightest.eu/ttl_fido2fingerprintAttestationN.tpl

_translate._trust.fido2fingerprint.attestation.fidouaf.eu. IN URI
 https://lightest.eu/ttl_fido2fingerprintAttestation1.xml
...
_translate._trust.fido2fingerprint.attestation.fidouaf.eu. IN URI
 https://lightest.eu/ttl_fido2fingerprintAttestationN.xml
```

The equivalent levels are given by several files returned. Note that a trust translation level for a given trust level can be characterized by more than one file: the TTA module provides with a file for each recognized trust level.


Besides, notice there are two kinds of files, TPL files and XML files, depending on the client to manage the answers. For example, the ATV is expecting the results in TPL format. Other clients could get the answers in XML format.

Since the content of those files is signed, it will also check whether the certificate used for signing the files is valid according with the content of SMIMEA resource record.

```
;; QUESTION SECTION: Verifying authenticity
_translate._trust.fido2fingerprint.attestation.fidouaf.eu. IN SMIMEA

;; ANSWER SECTION:
_translate._trust.fido2fingerprint.attestation.fidouaf.eu. IN SMIMEA
 <SMIMEA record data>
```

If validation succeeds, the verifier is sure that the list of the translations obtained is authentic: the publishing originator has been authenticated.

<b>Document name:</b>	Discovery of Trust Translation Authorities	<b>Page:</b>	27 of 32	
<b>Dissemination:</b>	PU	<b>Version:</b>	Version 1.1	

## 8. Summary and Conclusion

This deliverable describe the discovery mechanism for the Trust Translation Authority, which has been designed based on D4.1 “Conceptual Framework for Trust Scheme Translation” of LIGHTest, and according to the consolidate approach for the publication of Trust related information. It includes mechanism to verify the authenticity of the information by means of DNSSEC and DANE extensions.

The discovering of Trust Translation declarations is illustrated by mean of several examples for the different types of Trust Schemes; boolean, ordinal and tuple based. Extending examples provided in D4.3.

<b>Document name:</b>	Discovery of Trust Translation Authorities	<b>Page:</b>	28 of 32
<b>Dissemination:</b>	PU	<b>Version:</b>	Version 1.1
		<b>Status:</b>	Final



## 9. References

- [1] The LIGHTest Project, “D4.1- Conceptual Framework for Trust Scheme Translation (1),” Project Deliverable, 2017.
- [2] D. The LIGHTest Project, DNS-based Publication of Trust Schemes, Project Deliverable, 2018.
- [3] The LIGHTest Project, D4.3 - DNS-based Publication of Trust Translation Schemes, Project Deliverable, 2017.
- [4] The LIGHTest Project, “D5.3- Publication of Delegations,” Project Deliverable, 2018.
- [5] The LIGHTest Project, “D3.4 - Discovery of Trust Scheme Publication Authorities,” Project Deliverable, 2018.
- [6] The LIGHTest Project, “D5.4 - Discovery of Delegations,” Project Deliverable, 2018.
- [7] The LIGHTest Project, “D3.3 - DNS-based publication of Trust Schemes,” Project Deliverable, 2018.
- [8] ETSI TS 119 612, “Electronic Signatures and Infrastructures (ESI);Trusted Lists,” European Telecommunications Standards Institute; Technical Specification, Sophia Antipolis Cedex, V2.1.1 (2015-07), 2015.
- [9] O. K. P. Falstrom, The Uniform Resource Identifier (URI) DNS Resource, RFC 7553, Internet Engineering Task Force, June 2015.
- [10] P. Mockapetris, Domain names – implementation and specification, RFC 1035, Internet Engineering Task Force, November 1987.
- [11] T. L. Project, D2.7 -Relevant DNSSEC Concepts and Basic Building Blocks, Project Deliverable, 2017.
- [12] The LIGHTest Project, “D3.1 - Conceptual Framework for Trust Schemes,” Project Deliverable, 2017.
- [13] European Parliament, Regulation (EU) No 910/2014 of the European Parliament and of the Council of 23 July 2014 on electronic identification and trust services for electronic transactions in the internal market and repealing Directive 1999/93/EC, European Parliament, Brussels, Belgium, Regulation 910/201, 2014.

<b>Document name:</b>	Discovery of Trust Translation Authorities	<b>Page:</b>	29 of 32
<b>Dissemination:</b>	PU	<b>Version:</b>	Version 1.1
		<b>Status:</b>	Final



- [14] ISO, "ISO/IEC 29115:2013 Information technology -- Security techniques -- Entity authentication assurance," 2013. [Online]. Available: [http://www.iso.org/iso/catalogue\\_detail.htm?csnumber=45138](http://www.iso.org/iso/catalogue_detail.htm?csnumber=45138). [Accessed 2016].
- [15] FIDO Alliance, "FIDO Metadata Statements," 02 02 2017. [Online]. Available: <https://fidoalliance.org/specs/fido-uaf-v1.1-id-20170202/fido-metadata-statement-v1.1-id-20170202.html>. [Accessed 09 01 2018].
- [16] FIDO Alliance, "FIDO UAF Protocol Specification," 02 02 2017. [Online]. Available: <https://fidoalliance.org/specs/fido-uaf-v1.1-id-20170202/fido-uaf-protocol-v1.1-id-20170202.html>. [Accessed 21 06 2017].
- [17] J. S. P. Hoffmann, The DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS) Protocol: TLSA, RFC 6698, Internet Engineering Task Force, August 2012.
- [18] O. Gudmundsson, Adding Acronyms to Simplify Conversations about DNS-based Authentication of Named Entities (DANE), RFC 7218, Internet Task Force, April 2014.
- [19] S. Wagner, S. Kurowski, U. Laufs and H. Rossnagel, "A Mechanism for Discovery and Verification of Trust Scheme Memberships: The LIGHTest Reference Architecture," in *L. Fritsch et al. (Eds.): Open Identity Summit, Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn, 2017*.
- [20] FIDO Alliance, "FIDO Metadata Service," 02 02 2017. [Online]. Available: <https://fidoalliance.org/specs/fido-uaf-v1.1-id-20170202/fido-metadata-service-v1.1-id-20170202.html>. [Accessed 09 01 2018].
- [21] J. S. P. Hoffmann, Using Secure DNS to Associate Certificates with Domain Names for S/MIME, RFC8162, Internet Engineering Task Force, May 2017.

<b>Document name:</b>	Discovery of Trust Translation Authorities	<b>Page:</b>	30 of 32
<b>Dissemination:</b>	PU	<b>Version:</b>	Version 1.1
		<b>Status:</b>	Final



## 10. 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

<b>Document name:</b>	Discovery of Trust Translation Authorities	<b>Page:</b>	31 of 32
<b>Dissemination:</b>	PU	<b>Version:</b>	Version 1.1
		<b>Status:</b>	Final



# Discovery of Trust Translation Authorities



Identity Exchange (GB), NLNet Labs (NL), CORREOS (ES), IBM Danmark (DK) and Ubisecure (GS). 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.

<b>Document name:</b>	Discovery of Trust Translation Authorities	<b>Page:</b>	32 of 32
<b>Dissemination:</b>	PU	<b>Version:</b>	Version 1.1
		<b>Status:</b>	Final

