**ATIS-0x0000x**

ATIS Technical Report on

**Signature-Based Handling of Asserted Information Using Tokens (SHAKEN): Enterprise Identity and TN allocation utilizing Distributed Ledger Technology for OSP Attestation**

**Alliance for Telecommunications Industry Solutions**

Approved Month DD, YYYY

**Abstract**

Abstract text here.

**Foreword**

The Alliance for Telecommunications Industry Solutions (ATIS) serves the public through improved understanding between carriers, customers, and manufacturers. The DLT Focus Group initiated to validate key aspects of DLT as it applies to real-world challenges facing today’s communications industry. From a number of potential use cases, one was selected for a more in-depth analysis and proof of concept. The Enterprise Identity Network use case addresses current challenges differentiating robocalls from legitimate calls placed by enterprises.

The mandatory requirements are designated by the word *shall,* and recommendations by the name *should*. Where both a mandatory requirement and a recommendation are specified for the same criterion, the recommendation represents a goal currently identifiable as having distinct compatibility or performance advantages. The word *may* denotes an optional capability that could augment the standard. The standard is fully functional without the incorporation of this optional capability.

Suggestions for improvement of this document are welcome. They should be sent to the Alliance for Telecommunications Industry Solutions, [DLT Focus Group], 1200 G Street NW, Suite 500, Washington, DC 20005.

At the time of consensus on this document, DLT Focus Group which was responsible for its development, had the following leadership organizations:

* First Orion
* IBM
* Inteliquent
* Metaswitch
* Numeracle
* TNSI
* T-Mobile

The DLT Focus Group was responsible for the development of this document.

**Revision History**

| **Date** | **Version** | **Description** | **Author** |
| --- | --- | --- | --- |
| 6th August 2020 | 1.0 | Initial Contribution for IPNNI | Ian Deakin |

**Table of Contents**

[1 Scope, Problem 1](#_Toc47627700)

[1.1 Scope 1](#_Toc47627701)

[1.2 Problem Statement 1](#_Toc47627702)

[2 Normative References 3](#_Toc47627703)

[3 Definitions, Acronyms, & Abbreviations 4](#_Toc47627704)

[3.1 Definitions 4](#_Toc47627705)

[3.2 Acronyms & Abbreviations 5](#_Toc47627706)

[4 Overview 7](#_Toc47627707)

[4.1 Enterprise Identity Distributed Ledger Network Interworking with SHAKEN Architecture 10](#_Toc47627708)

[4.2 Overview of Self-Sovereign Identity (SSI) 10](#_Toc47627709)

[4.3 Overview of Decentralized Identifier 11](#_Toc47627710)

[4.4 Overview of Verifiable Credentials 13](#_Toc47627711)

[5 Enterprise Identity Management on Distributed Ledger 14](#_Toc47627712)

[5.1 Creation of a Decentralized Identifier DID and recording of the EIDL 14](#_Toc47627713)

[5.1.1 Identity Creation API 14](#_Toc47627714)

[5.2 Trust Authority Hierarchy 15](#_Toc47627715)

[5.3 Authorization Verifiable Credential 16](#_Toc47627716)

[5.3.1 Organization Vetting API 17](#_Toc47627717)

[6 TN Authorization 18](#_Toc47627718)

[6.1 Allocation of a TN by the Numbering authority to a TNSP 20](#_Toc47627719)

[6.2 Assignment of a TN by TNSP to TNR 22](#_Toc47627720)

[6.3 Delegation of a TN by TNR to Brand / Brand to BPO 24](#_Toc47627721)

[6.3.1 TN Authorization API 26](#_Toc47627722)

[6.4 TN Cancel and Revocation - Credential Claim Status 27](#_Toc47627723)

[7 OSP Attestation of a call using EIDL Network 28](#_Toc47627724)

[7.1 Enterprise PASSPort Encoding 29](#_Toc47627725)

[7.1.1 Use of modified base PASSPorT encoding for Enterprise Identity 29](#_Toc47627726)

[7.1.2 Use of 'rcd' PASSporT encoding for Enterprise Identity 30](#_Toc47627727)

[7.2 Verification of an Enterprise Identity and calling TN authorization by an OSP 31](#_Toc47627728)

[8 ANNEX A : Example of Open API for EIDL 32](#_Toc47627729)

[8.1 Organization Accounts 32](#_Toc47627730)

[8.1.1 createAccount 32](#_Toc47627731)

[8.1.2 getAccount 34](#_Toc47627732)

[8.2 TNAllocation 35](#_Toc47627733)

[8.3 Vetter 38](#_Toc47627734)

[8.3.1 createVettedOrganisationOnDLT 38](#_Toc47627735)

[8.3.2 getVettedOrganisationAuthorization 40](#_Toc47627736)

**Table of Figures**

[INSERT]

**Table of Tables**

[INSERT]

# Scope, Problem

## Scope

This specification extends the capabilities of SHAKEN to enable an enterprise to establish ‘Enterprise Identity’ credentials by applying Distributed Ledger Technology and its cryptographic principles. A verified ‘Enterprise Identity' allows the enterprise to be assigned or delegated 'Trusted' telephone numbers by authorized Telephone Number Service Providers (TNSPs) or Telephone Number Resellers (TNRs). Using a verified ‘Enterprise Identity’ with an authorized telephone number, the enterprise can place calls signed with their ‘Enterprise Identity’ credentials, enabling any Origination Service Provider (OSP) receiving the call, to authenticate the '‘Enterprise Identity'. The OSP can also determine the authorization of the signing entity using the telephone number from the Distributed ledger for the purpose of marking a call with the SHAKEN “Full” or “A-level” attestation. This distributed Ledger infrastructure is called the 'Enterprise Identity Network'.

The ‘Enterprise Identity’ credential is a W3C-standard Decentralized Identifier (DID)[[1]](#footnote-2) document recorded on the distributed ledger, authenticated by Public/Private Key pair cryptography.

All authorized TN assignments or delegations will be recorded on the distributed ledger by the issuing authority using signed verifiable credentials recorded on the distributed ledger according to the W3C standard format.

An enterprise will create a SIP Identity header on its outgoing calls containing a PASSporT signed with their ‘Enterprise Identity’ private key and including a reference to the DID credential. The signature and reference will enable any OSP connected to the distributed ledger to authenticate the signed PASSporT using the DID/public key stored on the DLT. The OSP can then verify that the Originating TN being used for the outgoing call is authorized to be used by the ‘Enterprise Identity' checking the signed verifiable credential for the TN.

When the OSP can authenticate the calling ‘Enterprise Identity' and the originating TN is authorized to be used, the OSP can apply SHAKEN A-level attestation to the call.

## Problem Statement

The Federal Trade Commission (FTC) and Federal Communications Commission (FCC) regularly cite “unwanted and illegal robocalls” as their number-one complaint category, and caller ID spoofing increases the harm from these unwanted calls. The ATIS-SIP Forum IP-NNI Task Force has developed a specification explicitly designed to mitigate unsolicited and unwanted robocalls by reducing the impact of illegitimate caller ID spoofing. This specification, SHAKEN (Signature-based Handling of Asserted information using tokens) that is based on the STIR (Secure Telephone Identity Revisited) protocol developed by IETF, allows the originating service provider to generate a digital signature that securely signals the caller's right to use a phone number to the terminating service provider.

When a call is originated, the originating service provider will use the originating SIP identity header to create a digital signature, or token, that will accompany the call as it is being completed. At call termination, the terminating service provider verifies that nothing was tampered with and ensures that the call came from an entity that has a legitimate right to use that number. The verification from SHAKEN can be displayed directly to the user or fed into a "“call-blocking app"” that provides a rating system that will essentially identify calls as good, questionable, or likely fraudulent. The call-blocking app can take action, on behalf of the user, to stop unwanted calls from getting through.

SHAKEN was never intended to be a complete solution for addressing the robocalling problem. It is an important tool in a multi-layered approach. What SHAKEN specifically does is to allow an OSP to populate a SIP "“Identity"” header on its calls destined to other service providers. This SIP “Identity” header that contains a cryptographically signed token (the SHAKEN PASSporT) that authenticates the OSP's identity for network traceability and non-repudiation, and protects the integrity of SIP parameters such as the calling TN. One of the parameters, or “claims", provided in the SHAKEN PASSporT, is an "“attestation"” level that the OSP marks on the call indicating its relationship to a “customer” entity and the customer's association with the calling TN. A “Full Attestation” level specifically indicates that a call was received from a known and authenticated enterprise and that there is a verified “association” of the calling TN to the call. The SHAKEN framework, ATIS-1000074, defines the criteria for Full Attestation as follows:

**A. Full Attestation:** The signing provider shall satisfy all of the following conditions:

1. Is responsible for the origination of the call onto the IP based service provider voice network.
2. Has a direct authenticated relationship with the customer and can identify the customer.
3. Has established a verified association with the telephone number used for the call.

Many call scenarios today will not satisfy these conditions through information locally available to the OSP and may, therefore, receive "“partial attestation"” (Attestation level “B”) at most. Third-party call centers are a great example of a situation that may not allow full attestation via information locally known to the OSP. Full attestation is not provided in this case due to the original assignment or delegation of a TN to a different entity than the one placing the call. Full attestation is not provided when the assignment and/or delegation of the TN coming from a TNSP that is not also the OSP.

SHAKEN does not define what service providers should do with the delivery of a call. Call delivery is governed by existing regulations that generally (with a few exceptions) require that telephone service providers complete all calls. Note, this does not apply to call-blocking apps, which allow the end-user to opt-in and have the app screen, characterize, and potentially block, calls on the user’s behalf. In March 2020, the FCC issued a Declaratory Ruling allowing service providers to apply call blocking at the network level on an opt-out basis.

SHAKEN does not characterize calls, and therefore it cannot identify calls correctly or incorrectly. SHAKEN merely creates a digital signature at the origin with the information the originating service provider knows about the call (customer, and potentially their right to use the number) and sends this signature to the terminating service provider, which verifies that no one has tampered with the information. SHAKEN does not identify calls as “SPAM". Call blocking apps are the entity that tries to identify the call intent, and as a result, sometimes label “legitimate” calls as spam.

Robocalls can sometimes be quite valuable to customers in communicating important, need-to-know information that otherwise may not have been received in real-time time. A few examples include flight or school cancellations, appointment reminders, and credit card fraud alerts. However, current analytics deployments can label these legal callers as FRAUD/SCAM.

As previously mentioned, these typical call scenarios are complex; an enterprise will contract with a call center provider, who will use another vendor (or vendors) for the calling platform, and obtain TNs for the campaign from a telephone number reseller(s) and originate the calls using multiple OSPs (some examples actually use a dozen OSPs) based on time of day and other factors to minimize costs. With this scenario, the OSP has absolutely no confidence in the phone number in the caller ID, or anything else about the call and can only provide “partial attestation”. A solution is needed to provide the ability for an OSP to associate the calling number of a call with the verified 'Enterprise identity' responsible for or sponsoring the content of the call.

# Normative References

The following standards contain provisions which, through reference in this text, constitute provisions of this Standard. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this Standard are encouraged to investigate the possibility of applying the most recent editions of the standards indicated below.

ATIS-1000074, *Signature-based Handling of Asserted Information using Tokens (SHAKEN).*

ATIS-1000080, *SHAKEN: Governance Model and Certificate Management,*

ATIS-1000076, *Enterprise Identity on Distributed Ledger for Authenticated Caller Use Cases*

IETF RFC 3261, *SIP: Session Initiation Protocol.*3

IETF RFC 3325, *Private Extensions to SIP for Asserted Identity within Trusted Networks.*3

OpenAPI, *OpenAPI Specification (OAS)Specification Version 3.0.3*

RFC 4122, *A Universally Unique Identifier (UUID) URN Namespace.*2

RFC 4949, *Internet Security Glossary, Version 2.*2

RFC 5806, *Diversion Indication in SIP*. 2

RFC 7044, *An Extension to the Session Initiation Protocol (SIP) for Request History Information*. 2

RFC 8224, *Authenticated Identity Management in the Session Initiation Protocol.*2

RFC 8225, *Personal Assertion Token.*[[2]](#footnote-3)

RFC 8226, *Secure Telephone Identity Credentials: Certificates.*2

TS 24.229, IP multimedia call control protocol based on Session Initiation Protocol (SIP) and Session Description Protocol (SDP). [[3]](#footnote-4)

W3C, Decentralized Identifiers (DIDs) v1.0 [[4]](#footnote-5)

W3C, Verifiable Credentials Data Model 1.0 4

ITU-T Recommendation X.811 (04/1995) | ISO/IEC 10181-2:1996, Information technology – Open Systems Interconnection – Security frameworks for open systems: Authentication framework

NIST SP 800-63-3 - NIST Special Publication 800-63-3 Digital Identity Guidelines

# Definitions, Acronyms, & Abbreviations

For a list of common communications terms and definitions, please visit the *ATIS Telecom Glossary*, which is located at < <http://www.atis.org/glossary> >.

## Definitions

**BPO Enterprise:** Is a Business Process Outsource organization (Call Center) placing calls on behalf of a Brand Enterprise.

**Brand Enterprise:** Is the business entity who is representing the calling party TN, even when the call is placed by a BPO, the caller identity in the SIP call is using the Brand Enterprise name and intended purpose for the call when contacting consumers.

**Caller ID:** The originating or calling party’s telephone number used to identify the caller carried either in the P-Asserted-Identity or from header fields in the Session Initiation Protocol (SIP) [RFC 3261] messages.

**Company Code:** A unique four-character alphanumeric code (NXXX) assigned to all Service Providers [ATIS-0300251].

**Decentralized Identifiers:** (DIDs) are a new type of identifier that enables verifiable, decentralized digital identity. (DIDs), and can authenticate using proofs (for example, digital signatures, privacy-preserving biometric protocols, and so on)

**Decentralized Identifier Document:** Also referred to as a DID document, this is a document that is accessible using a verifiable data registry and contains information related to a specific decentralized identifier, such as the associated repository and public key information.

**End-Entity:** An entity that participates in the Public Key Infrastructure (PKI). Usually a Server, Service, Router, or a Person. In the context of this document, an end-entity is a Service Provider, TN Service Provider, or VoIP Entity.

**Enterprise**: For the purposes of this document, a business, governmental body, or non-governmental body that participates in VoIP calling and TN assignments and delegations. In some instances, an “enterprise” may also be an entity providing VoIP calling services and/or infrastructure.

**Fingerprint:** A hash result ("key fingerprint") used to authenticate a public key or other data [RFC 4949].

**Identity:** Unless otherwise qualified, an identifier that unambiguously distinguishes an entity for authentication and other security and policy application purposes (compare to “distinguished identifier” as used in X.811 [Ref ITU-T Rec. X.811 (1995 E)]). For the purposes of this report, an identity may or may not be a "“TN-based caller identity",” depending on the context.

**National/Regional Regulatory Authority (NRRA):** A governmental entity responsible for the oversight/regulation of the telecommunication networks within a specific country or region.

Note: Region is not intended to be a region within a country (e.g., a region is not a state within the US).

**Numbering Authority:** This is the authority that indicates the authoritative assignment of a TN to a TNSP. This will be either, TN Primary Number plan range allocation (NANPA) to a TNSP. Including any port corrected TN data (LNPA).

**Private Key:** In asymmetric cryptography, the private key is kept secret by the end-entity. The private key can be used for both encryption and decryption [RFC 4949].

**Public Key:** The publicly disclosable component of a pair of cryptographic keys used for asymmetric cryptography [RFC 4949].

**Real-world identity**: Identifiers and identifying characteristics of a principal outside of the telecommunications services domain, including but not limited to personal or business name, physical location or postal address, government-issued identifiers, credit information, etc. Compare to the use of “real-life identity” and “real-world identity” in NIST SP 800-63-3 [Ref 8].

**Responsible Organization (Resp Org):** Entity designated as the agent for the Toll-Free subscriber to obtain, manage and administer Toll-Free Numbers and provide routing reference information in the SMS/800 Toll-Free Number Registry.

**Resp Org Identification (Resp Org ID):** A 5-character code that designates or points to the Responsible Organization (Resp Org) associated with a specific Toll-Free number [ATIS-0417001-003].

**Secure Telephone Identity (STI) Certificate:** A public-key key certificate used by a service provider to sign and verify the SHAKEN PASSporT.

**Service Provider Code:** In the context of this document, this term refers to any unique identifier that is allocated by a Regulatory and/or administrative entity to a service provider. In the US and Canada, this would be a Company Code as defined in [ATIS-0300251], or a Resp Org ID assigned to a Resp Org as defined in [ATIS-0417001-003].

Editor’s note: Further analysis is required to determine if Resp Org should be included as part of the service provider code or somewhere else.

**Signature:** Created by signing the message using the private key. It ensures the identity of the sender and the integrity of the data [RFC 4949].

**Telephone Identity:** An identifier associated with an originator of a telephone call. In the context of the SHAKEN framework, this is a SIP identity (e.g. a SIP URI or a TEL URI) from which a telephone number can be derived.

**TN Allocation:** The granting of a TN numbering block or individual TN to a TNSP by a TN numbering authority ( E.g. NANPA, LNPA, SMS/800 number).

**TN Assignment:** The procedurefor a TNSP togrant the right to use a block or individual TN to a TNR, Brand Enterprise or other entity.

**TN Delegation:** The procedurefor a TNR, Brand Enterprise or other TN assignee togrant the right to use an individual TN to a Brand Enterprise, BPO or other entity.

**Trust Authority:** The Trust Authority in the Enterprise Identity network is an assigned 3rd party providing three functions. 1. Vetting and authorization of the Enterprise Identity TNSPs and KYC vetters. 2. Vetting and authorization of the EIDL network nodes. 3. Implements the TN authorization rules and process on the EIDL according to the Numbering Authority.

**Verifiable Credential:** are the electronic equivalent of the physical credentials that we all possess today, such as: plastic cards, passports, driving licenses, qualifications, and awards, etc. Verifiable Credentials (VC) may be expressed using [JSON](https://en.wikipedia.org/wiki/JSON). A VC is typically composed of: the context of the VC, the identity of the issuer, the date and time of issuing, the expiry date and time, the type of VC, the subject of the VC, one or more identity attributes of the VC subject, and finally the cryptographic proof created by the issuer to ensure the integrity and authenticity of the VC.

**VoIP Entity:** A non-STI-authorized customer entity that purchases (or otherwise obtains) delegated telephone numbers from a TNSP

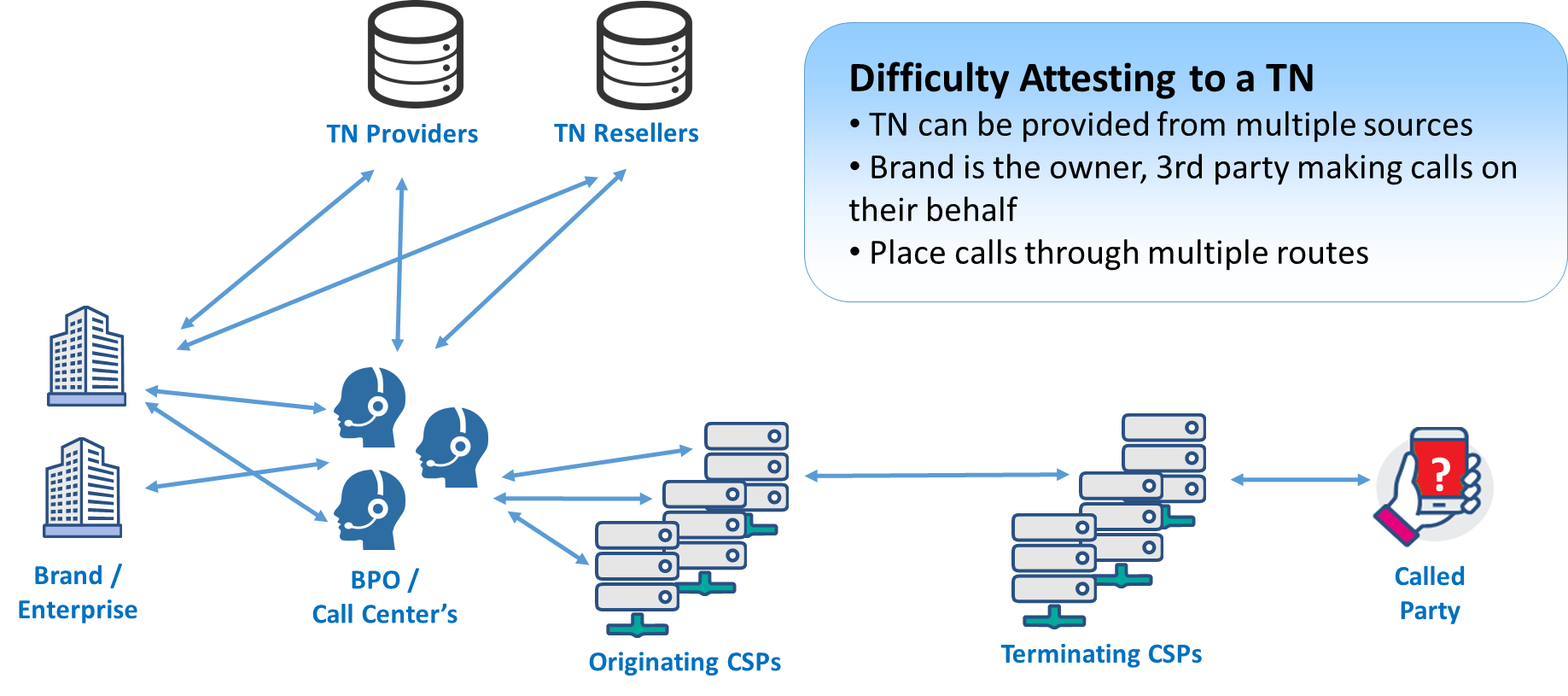
## Acronyms & Abbreviations

|  |  |
| --- | --- |
| 3GPP | 3rd Generation Partnership Project |
| ATIS | Alliance for Telecommunications Industry Solutions |
| B2BUA | Back-to-Back User Agent |
|  |  |
| CSCF | Call Session Control Function |
| CVT | Call Validation Treatment |
| DID  DLT | Decentralized Identifier  Distributed Ledger Technology |
| HTTPS | Hypertext Transfer Protocol Secure |
| IBCF | Interconnection Border Control Function |
| IETF | Internet Engineering Task Force |
| IMS | IP Multimedia Subsystem |
| IP | Internet Protocol |
| JSON | JavaScript Object Notation |
| JWS | JSON Web Signature |
| NNI | Network-to-Network Interface |
|  |  |
| OSP | Originating Service Provider |
| PASSporT | Persona Assertion Token |
| PBX | Private Branch Exchange |
| PKI | Public Key Infrastructure |
| SHAKEN | Signature-based Handling of Asserted information using toKENs |
| SIP | Session Initiation Protocol |
| SKS | Secure Key Store |
| SP | Service Provider |
| SPID | Service Provider Identifier |
| STI | Secure Telephone Identity |
| STI-AS | Secure Telephone Identity Authentication Service |
| STI-CA | Secure Telephone Identity Certification Authority |
| STI-CR | Secure Telephone Identity Certificate Repository |
| STI-VS | Secure Telephone Identity Verification Service |
| STIR | Secure Telephone Identity Revisited |
| TLS | Transport Layer Security |
| TN | Telephone Number |
| TNSP | TN Service Provider |
| TSP | Terminating Service Provider |
| UA | User Agent |
| URI | Uniform Resource Identifier |
| UUID | Universally Unique Identifier |
| VC | Verifiable Credential |
| VoIP | Voice over Internet Protocol |

# Overview

This document presents a method for participants in the VoIP service provider network ecosystem, including TNSPs, TNRs, OSPs, Enterprises, KYC Vetters, and governance and trust authorities to record identity information and TN assignment and delegation information on a distributed ledger. The TN assignment and delegation information may be read from an instance of the ledger to determine a SHAKEN attestation level for a call from an entity whose information is recorded on the ledger. The document contains an example PASSporT format that may be used to assert an Enterprise Identity and authorization of the holder of that identity to use a calling TN, and the references in the PASSporT may be used by OSP to resolve that information from the distributed ledger. The method will be illustrated by a complex use case involving multiple layers of TN assignment and delegation and calling initiated by a party without a direct relationship to the TNSP or OSPs. This scenario demonstrates a range of capabilities that this method supports, although the method may be used for many different scenarios. The outline of the scenario is as follows:

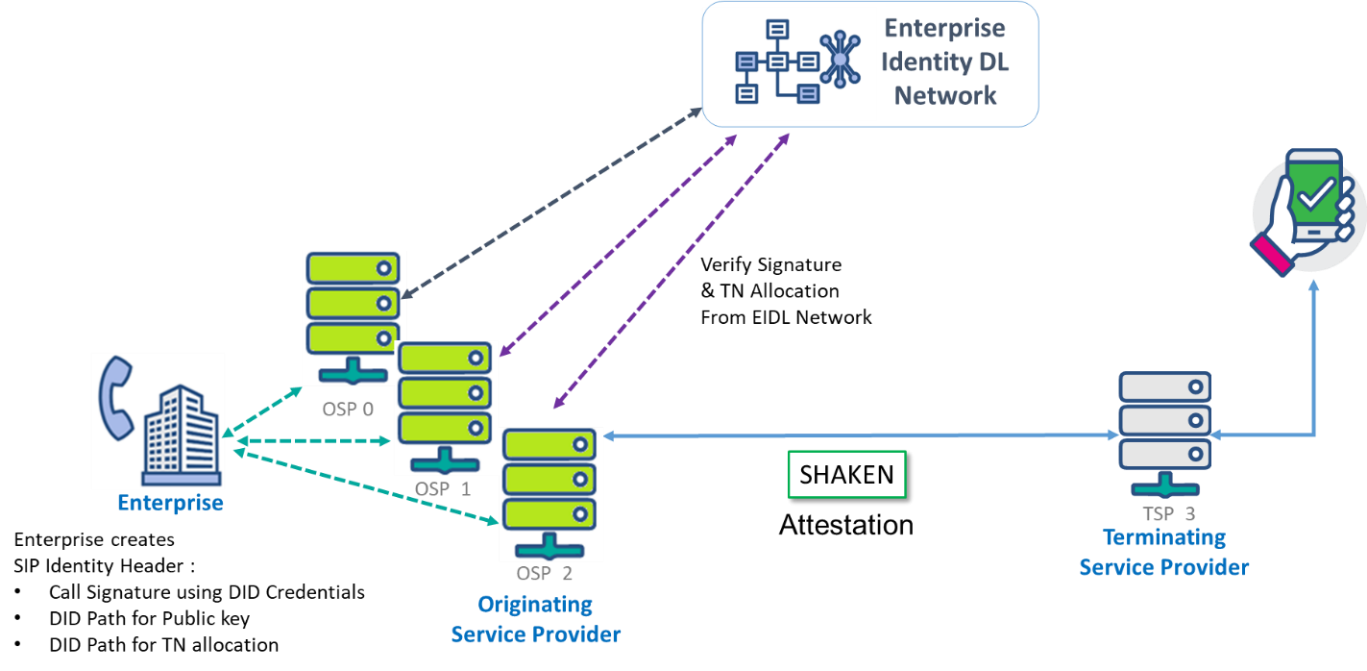
1. A TNSP assigns TNs to a TNR
2. The TNR delegates one or more TNs to a Brand/Enterprise for the Brand’s own general calling purposes
3. The Brand/Enterprise further delegates one or more TNs to a Business Process Outsourcing (BPO) Enterprise to conduct an outbound calling campaign on the Brand’s behalf.
4. The BPO initiates calls to one or more OSPs, each of which will make a SHAKEN attestation decision based on the contents of a PASSporT that the BPO populates on each call and the through resolution of information from the EIDL. The BPO may place calls indirectly to the OSPs through another entity such as a Call Processing as a Service (CPaaS) platform provider, which holds the direct relationship with the OSPs.
5. Based on the verification of the BPO’s PASSporT and identity and TN authorization information read from the EIDL, an OSP determines that the BPO’s calls marked with the Brand’s calling TN are entitled to Full Attestation marking designation in an outgoing SHAKEN PASSporT.



***Figure 4.1 Problem to attest a Telephone Number***

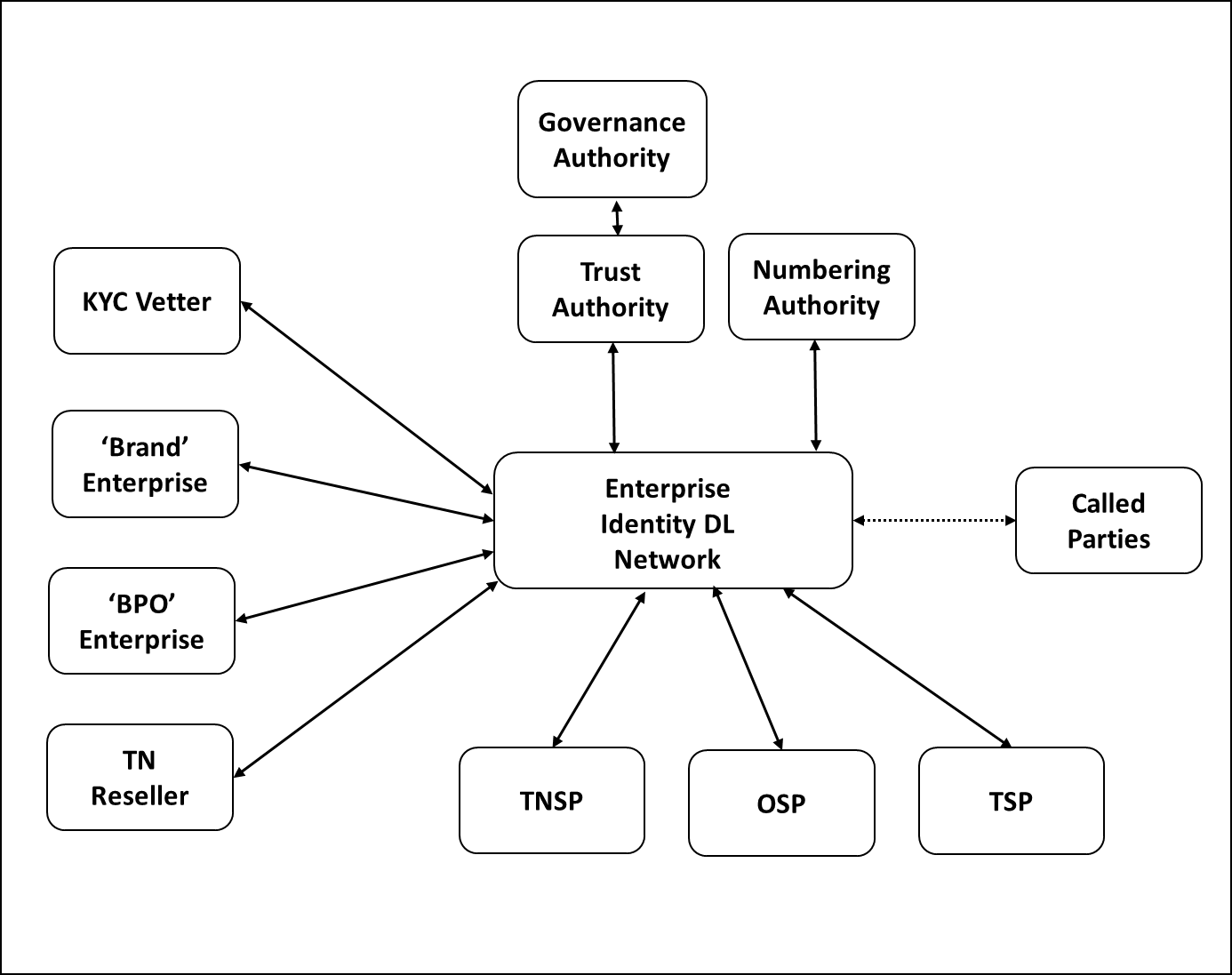
**Enterprise Identity Distributed Ledger Network**

By using distributed ledger technology, the Enterprise Identity Distributed Ledger (EIDL) Network provides control and scale for the establishment of an Enterprise Identity digital trust foundation, or an identity trust fabric (ITF) to all organizations. The EIDL Network enables any OSP in the call path to fully attest to a calling enterprise’s SIP call, with an authorized EIDL identity using an authorized TN assignment or delegation. A full description of the EIDL Network is explained in ATIS-1000076, *Enterprise Identity on Distributed Ledger for Authenticated Caller Use Cases.*



***Figure 4.2 Enterprise Identity Distributed Ledger Network to attest a Telephone Number***

The following actors have roles in the EIDL network's Enterprise authentication and calling TN authorization process.



***Figure 4.3 Enterprise Identity (EIDL) Network Actors***

For a full description of the EIDL actor roles and functions, refer to ATIS-1000076.

In addition to enterprise callers, all organization actors of the EIDL network must have an ‘Enterprise Identity‘ to transact on the network, not only the enterprise callers. An organization's ‘Enterprise identity' on the EIDL is implemented using distributed identities according to the W3C Distributed Identity (DID) specification.[[5]](#footnote-6) (DID W3C Reference <https://www.w3.org/TR/did-core/#definition> ).

The distributed Identity credential is generated with a public/private key pair by the organization client.

The benefit of using DID for identities enables the underlying distributed ledger technology to be agnostic, but also allows interoperability across other identity networks as a federated identity service to support cross border identity and TN authentication.

Once an Enterprise organization DID is created and registered on the EIDL, the enterprise can use a network of KYC vetters to verify and prove their identity. When the KYC Vetter has validated the real-world identity of the Enterprise according to the governance policy of the EIDL, the KYC Vetter will sign a verifiable claim for the organization’s DID indicating they are now 'approved'.

With an approved DID, an Enterprise making calls can be assigned or delegated TNs by an authorized TNSP, TN Reseller, or other enterprise entity that can be used to place calls.

All TN allocations (assignments or delegation) on the EIDL are represented and contained within verifiable credential (VC) claims (W3C reference <https://www.w3.org/TR/vc-use-cases/> [[6]](#footnote-7)). These verifiable credential claims are digitally signed assignments or delegation of TN from the authorized TNSP to a TNR to a Brand Enterprise and to one or more BPO (call Centers). Each TN is encapsulated in a single VC, as the TN is assigned from the TNSP to TNR, TNR to Brand, Brand to BPO, each organization assignment or delegation is cryptographically signed within the VC to prove that claim. From a single TN, VC provides the full chain of custody for the TN; who assigned it at what time and by who over its history. As these Verifiable Credential claims are on the distributed ledger, they are all true at any point in time as an authoritative source for all stakeholders to against which they can verify.

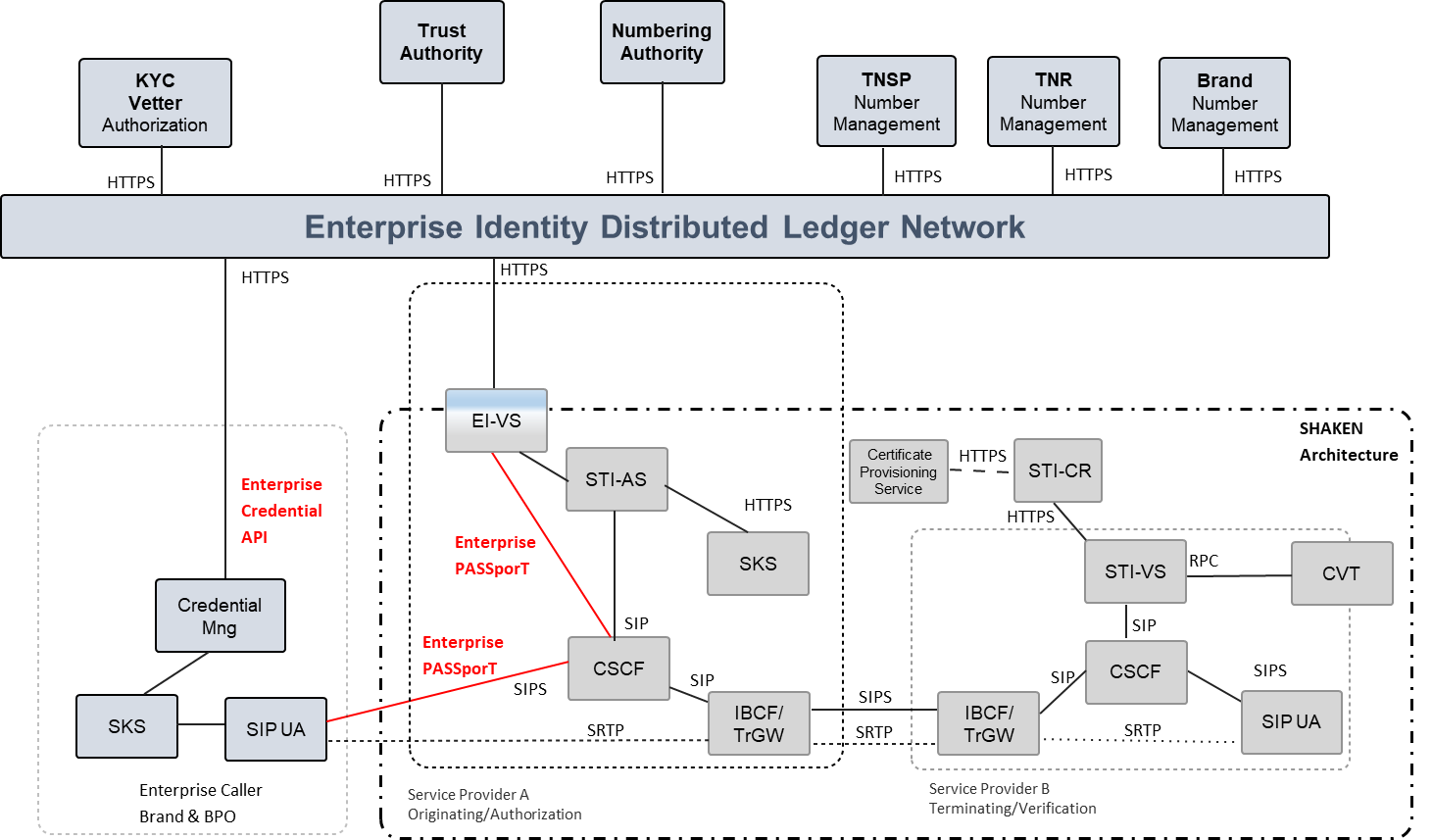
Using their DID, a KYC vetted enterprise will sign originating phone calls using the SIP Identity header. This signature enables any originating Service Provider (OSP) connected to the EIDL Network to verify the DID of the calling Enterprise and prove that the call is being placed only by a 'Trusted' enterprise.  Using this proof of identity, the OSP can then verify that the originating TN being used by the Enterprise has the authoritative right to use it by checking the verifiable credential for the TN on EIDL Network. This VC enables an OSP to authenticate an Enterprise Identity and prove that they are entitled to make calls from the calling TN.  With this proof of identity and the number being used, the originating service provider can attest to the call using SHAKEN A attestation.

An 'Enterprise Identity' implemented with distributed ledger technology, using DID and Verifiable credentials, will significantly enhance the capability for Originating Service Providers to provide validation and authentication of the calling party identity and the authority to place a call with an authorized TN. OSP’s will be able, to provide SHAKEN attestation for enterprise calls in complex the call scenarios described above.

This document will cover the implementation of the DID for Enterprise identity verification and TN authorization. For more information on the EIDL implementation, see the ATIS-I-0000076, *Enterprise Identity on Distributed Ledger for Authenticated Caller Use Cases,* technical report.

## Enterprise Identity Distributed Ledger Network Interworking with SHAKEN Architecture

The following architecture illustrates the Enterprise Identity Distributed Ledger network architecture interworking with the SHAKEN architecture to provide '‘Enterprise Identity' services to originating service providers.



***Figure 4.4 Enterprise Identity (EIDL) Network Interworking with SHAKEN Architecture***

## Overview of Self-Sovereign Identity (SSI)

Self-sovereign identity (SSI) is a decentralized user-centric model that gives the user full control of their own identity and information. In other words, the user is the owner and the administrator of his digital identity and their identity is independent of any service provider. The implementation of SSI requires three concepts, decentralized identifiers, decentralized identifiers documents, and verifiable claims. SSI implies that an individual or organization that has one or more identifiers or DIDs, can present specific claims or credentials relating to those DIDs without having to go through an intermediary. An SSI is not bound to a single identity issuer, therefore enabling the use of SSI to be portable across multiple credential issuers.

An EIDL Network Actor SSI can create an account on the EIDL, generating a decentralized identifier credential with a public -private key pair, where the DID and Public key are published to the EIDL for authentication when transacting with other actors on the EIDL network.

## Overview of Decentralized Identifier

The concept of a globally unique decentralized identifier is not new. Universally Unique Identifiers (UUIDs) were first developed in the 1980s and later became a standard feature of the Open Software Foundation’s Distributed Computing Environment. UUIDs achieve global uniqueness without a centralized registry service by using an algorithm that generates 128-bit values with sufficient entropy that the chance of collision is infinitesimally small. UUIDs are formally specified in [RFC4122] as a specific type of Unified Resource Name (URN).

A DID is similar to a UUID except that:

* Like a URL, it can be resolved or dereferenced to a standard resource describing the subject. That is, a DID document.
* Unlike most resources returned when dereferencing URLs, a DID document usually contains cryptographic material enabling authentication of the DID subject.

A Decentralized Identifier (DID) is a globally unique identifier that does not require a centralized registration authority and is created in a common trust domain called an Identity Trust Fabric (ITF). The ITF stores the proof of identities and their verifiable credentials cryptographically and immutably on the blockchain. The ITF is where supply-chain partners can verify the authenticity of an identity as well as related verifiable credentials. The ITF is the component that circumvents the need for a central identity provider to manage trust. Once a decentralized identity is established, any supply-chain partner can verify relevant attributes regarding another supply-chain partner with which it is about to engage in a business interaction, either by granting access or conducting a transaction.

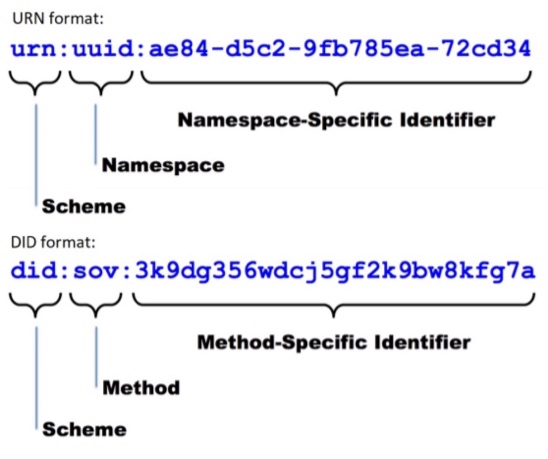
Entities are identified by decentralized identifiers (DIDs), and can authenticate using proofs (for example, digital signatures, privacy-preserving biometric protocols, and so on). DIDs point to DID documents. A DID document contains a set of service endpoints for interacting with the entity that the DID identifies (that is, the DID subject). Following the guidelines of Privacy by Design, any entity can have as many DIDs (and corresponding DID documents and service endpoints) as necessary to respect the entity’s desired separation of identities, personas, and contexts (in the everyday sense of these words).

DID methods are the mechanism by which a DID and its associated DID document are created, read, updated, and deactivated on a specific distributed ledger or network. DID methods are defined using separate DID method specifications.

This design eliminates dependence on centralized registries for identifiers as well as centralized certificate authorities for key management, which is the standard in hierarchical PKI (public key infrastructure). In cases where the DID registry is a distributed ledger, each entity can serve as its own root authority.

**DID Path format**

DID format is similar to the Uniform Resource Name (URN) format. The key difference is that we are specifying a method instead of a namespace in the DID format. The method is referring to the specification ‘W3C, Decentralized Identifiers (DIDs) v1.0' used to register and manage DIDs on a particular blockchain or DLT (Distributed Ledger Technology).



***Figure 4.5: DID format (Example Sovrin network method) compared to URN format***

**DID Document**

We can see DID infrastructure as a global key-value database where the database consists of all DID-compatible distributed ledgers. In this global and distributed database, the key is a DID, and the value is a DID document. A DID document is a JSON-LD file which stands for JavaScript Object Notation for Linked Data. The main advantage of JSON-LD is interoperability.

The DID specified in the DID Document includes the following six parts:

1. The DID itself, so the DID document is fully self-describing, and we can link other documents to this DID.

2. A set of public keys or other proofs that can be used for authentication or interaction with the identified entity.

3. A set of service endpoints that describe where and how to interact with the identified entity. A service endpoint could, for example, link to the mail service of the DID’s owner or to his storage service.

4. The owner of the DID can also decide to authorize other entities to make changes to the DID Document. This can become particularly important and useful in the case of private key loss.

5. Timestamps can be added for auditing the date of creation or update of a particular DID.

6. A JSON-LD signature is added if there is a need to verify the integrity of the document.

The format of an enterprise identity DID document is:-

"@context": ["https://www.w3.org/ns/did/v1", "https://w3id.org/security/v1"],

"id": "did:ibm:123456789abcdefghi",

"publicKey": [{

"id": "did:ibm:123456789abcdefghi#key-1",

"type": "Ed25519VerificationKey2018",

"controller": "did:ibm:123456789abcdefghi",

"publicKeyBase58": "H3C2AVvLMv6gmMNam3uVAjZpfkcJCwDwnZn6z3wXmqPV"

}]

}

***Note:*** in this format, the controller is the DID, being that it is self-sovereign.

## Overview of Verifiable Credentials

A verifiable claim is a qualification, achievement, quality, or piece of information about an entity's background, such as a name, government ID, payment provider, home address, or university degree. Such a claim describes a quality or qualities, property, or properties of an entity which establish its existence and uniqueness. The use cases outlined here are provided in order to make progress toward possible future standardization and interoperability of both low and high-stakes claims with the goals of storing, transmitting, and receiving digitally verifiable proof of attributes such as qualifications and achievements.

There are four roles supported by verifiable credentials: Issuer, Verifier, Subject, and Holder.

**Issuer:** The entity that creates a claim and associates it with a particular subject.

**Verifier:** The entity verifying a claim about a given subject.

**Subject:** The entity about whom a claim is issued.

**Holder:** A role an entity may perform by possessing one or more verifiable credentials. A holder is usually, but not always, the subject of the verifiable credentials that they are holding. Holders store their credentials in credential repositories.

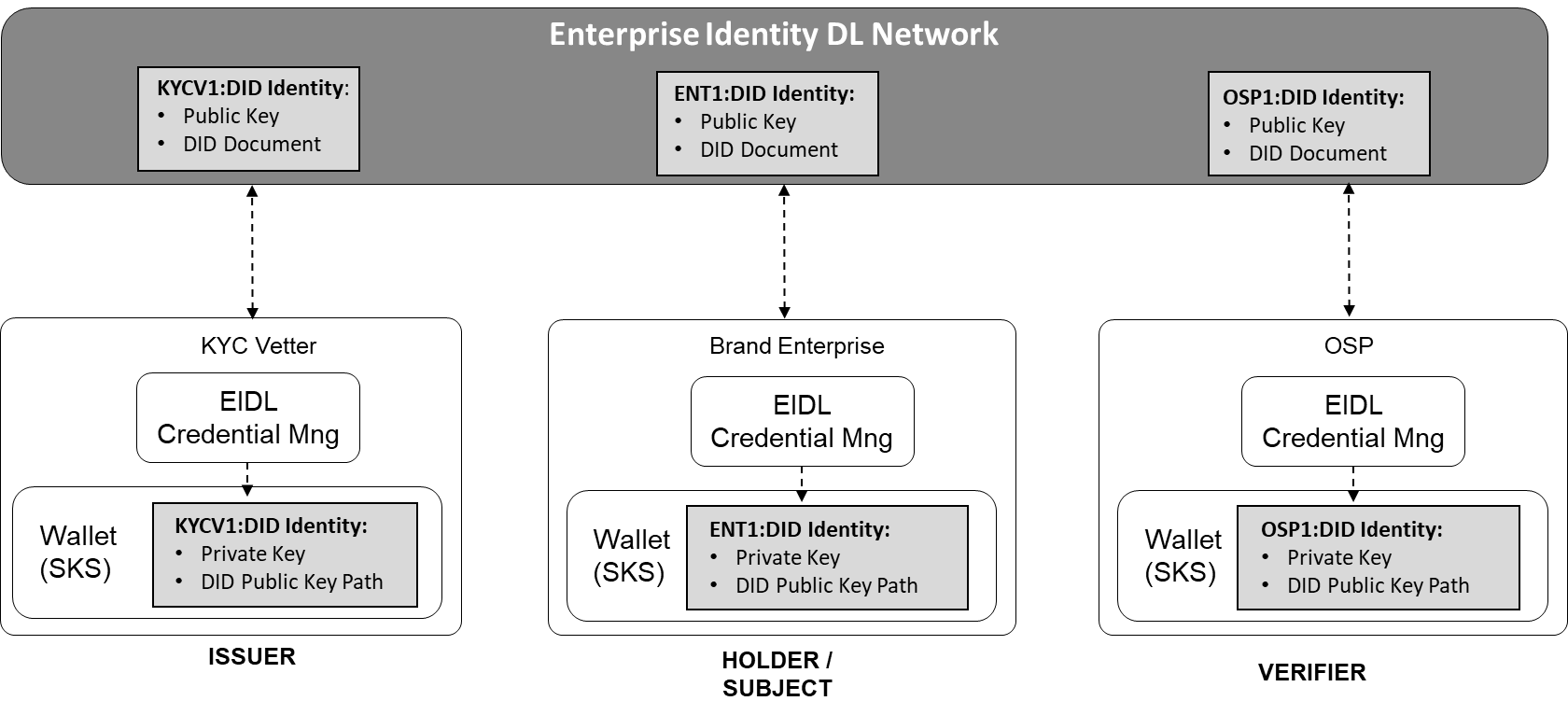
For more details on the Verifiable credential claim syntax see 'W3C, Verifiable Credentials Data Model 1.0.'

# Enterprise Identity Management on Distributed Ledger

This section describes the specific implementation of Decentralized Identifiers and Verifiable Credentials on the Enterprise Identity distributed ledger network to provide Enterprise caller TN attestation for the OSP. For more details on the Enterprise identity architecture, actors and functional flows, see ATIS-1000076.

## Creation of a Decentralized Identifier DID and recording of the EIDL

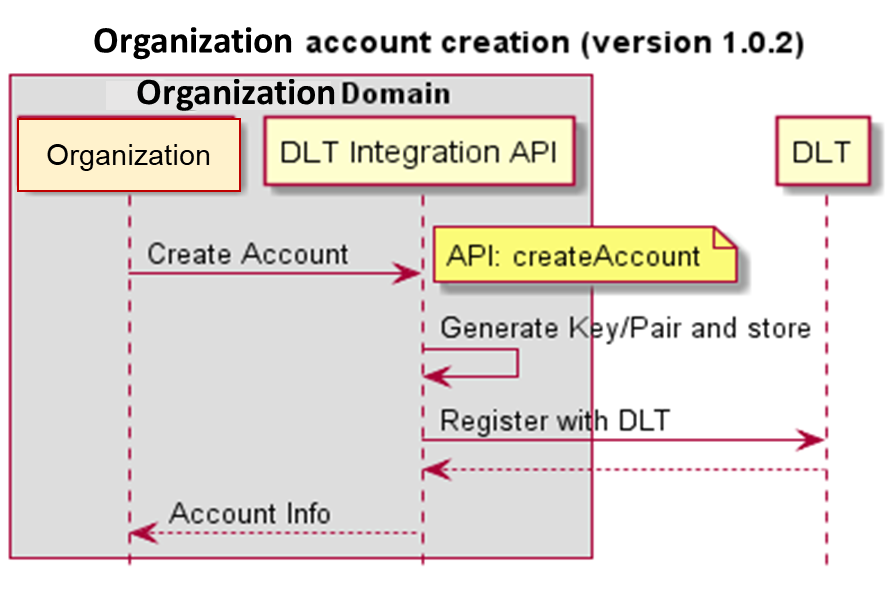
Each actor of the EIDL Network must generate a Decentralized Identifier. The Actor will create a DID identity and publish this onto the EIDL as a DID Document with its public key. The EIDL will return the DID Path, which is the address for the public key on the EIDL.



***Figure 5.1 Creation of DID on EIDL Network***

### Identity Creation API

Using the Enterprise Identity API definition described in Annex A of this document, an organization's client-side side credential manager will integrate with the EIDL to generate their DID credentials and a corresponding key pair. Here is an example of the organization's client-side API generating a key pair and registering with the EIDL.



***Figure 5.2 Creation of DID on EIDL Network***

***Note:*** the OSP must be a registered identity of the EIDL, permissioned to connect and query Enterprise Identity authorization status and TN assignments or delegation.

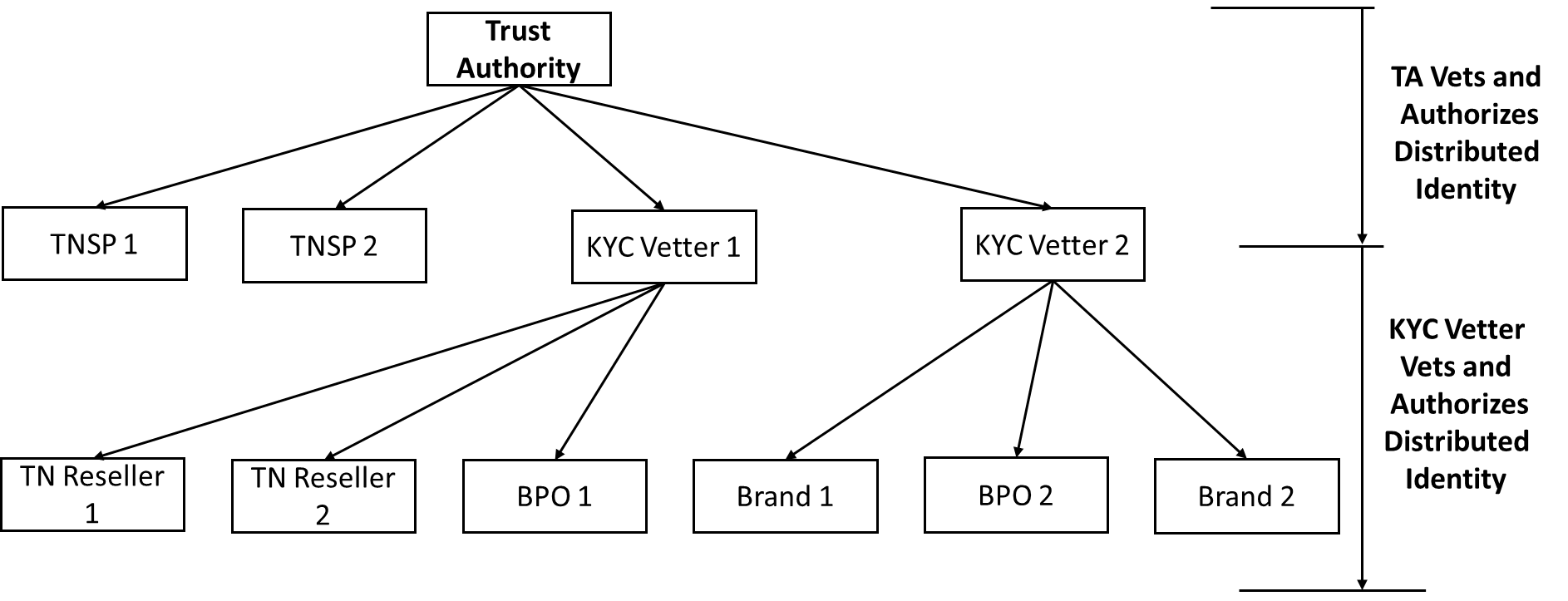
## Trust Authority Hierarchy

Every actor of the EIDL Network must have their DID authorized to participate in the Enterprise Identity service. Using the organization DID, the requesting ORG will apply to the appropriate vetting authority to have their DID authorized to participate.

The role of the Trust Authority (TA) is to verify the organizations that represent the TNSP and KYC Vetter actors of the EIDL Network. The vetting authorization process is according to the policy defined by the EIDL Network Governance Authority. When the TA has authorized an organization, the TA will create a verifiable credential claim on the EIDL for that organization's DID to indicate that they are authorized.

The role of the KYC Vetter is to verify the organizations that represent the TNR, Brand Enterprise, and BPO Enterprise actors of the EIDL Network. The vetting authorization process is according to the policy defined by the EIDL Network Governance Authority. When the KYC Vetter has authorized an organization, the KYC will create a verifiable credential claim on the EIDL for that organization's DID to indicate that they are authorized.

***Note:*** a KYC Vetter must be approved by the TA before they themselves can authorize other organization



***Figure 5.3 EIDL Network Trust Hierarchy***

The issuer ( TA or KYC Vetter ) of the Verifiable claim for the organization will contain data attributes that indicate the organization's business function TBD, for example :-

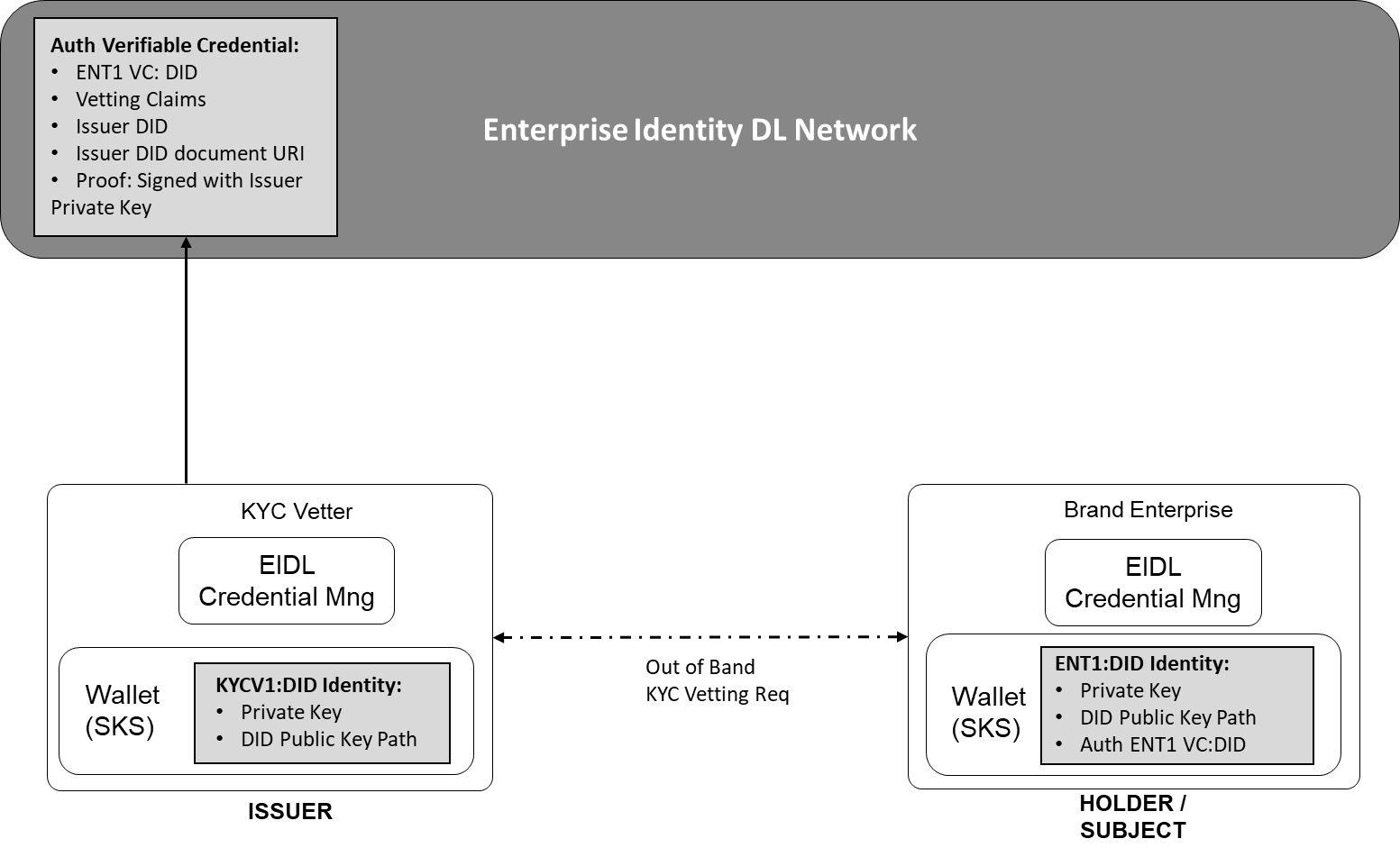
* Name of Business = “Company Name” – Assigned in KYC Process
* Nature of Business ( TNSP, KYC Vetter, TN Reseller, Brand, BPO )
* Vetting Identity = Identity ( Unique Allocation i.e. UUID ) – Assigned by DLT
* Current Authentication Status = Authorized, Not, Blacklisted for example
* Vetter Name = "“Vetter Business Name"” – Assigned by TA

## Authorization Verifiable Credential

As indicated in the Trust Hierarchy, the TA must vet and authorize a TNSP, KYC vetter, and OSP to perform their function on the EIDL network. Only an Authorized KYC vetter can then vet and authorize TNR, Brand Enterprise, and BPO enterprise actors to perform their function on the EIDL.

Using an out-of-band (OOB) request ( as indicated in option 1 below) , the Organization will request the Vetter Actor, to perform vetting of their business for authorization to use the EIDL Network services, according to the agreed policy of the EIDL ( out of scope for this document at this time ).

The vetting authority ( TA or KYC Vetter) , will perform vetting of the requesting organization according to the agreed EIDL policy. If authorized, the vetting authority will generate and sign a Verifiable Credential (VC) claim, indicating the organization DID and that their authorization status in now 'authorized.' The vetter returns the status back to the requesting organization together with the VC claim identity to support this.



***Figure 5.4 Request Org to be KYC Vetted and create Vetted VC Claim***

In the example outlined above, the Brand Enterprise (ENT1) has been authorized by the KYC Vetter (KYC1). KYC Vetter (KYC1) has generated a signed VC Claim to indicate this on the EIDL which can be checked by any other actor on the EIDL. As part of the authorization process, the Vetter will verify the Enterprise Brand Name that will be used to identify them. This is also included in the VC claim.

**Organization Vetting Authorization Verifiable Credential Claim**

Example of a W3C verifiable credential claim representing the vetting status of an organization.

{

"@context": [

"https://www.w3.org/2018/credentials/v1",

"https://access.atis.org/apps/group\_public/dlt/v1/"

],

"id": "did:ibm:<**ENT1**>",

"type": ["VerifiableCredential", "VettingStatus"],

"issuer": "did:ibm:<**KYC1**>",

"issuanceDate": "2020-01-01T19:73:24Z",

"credentialSubject": {

"id": "did:iota:<**Brand Name of Enterprise**>",

"organisationVettingStatus": "Authorized"

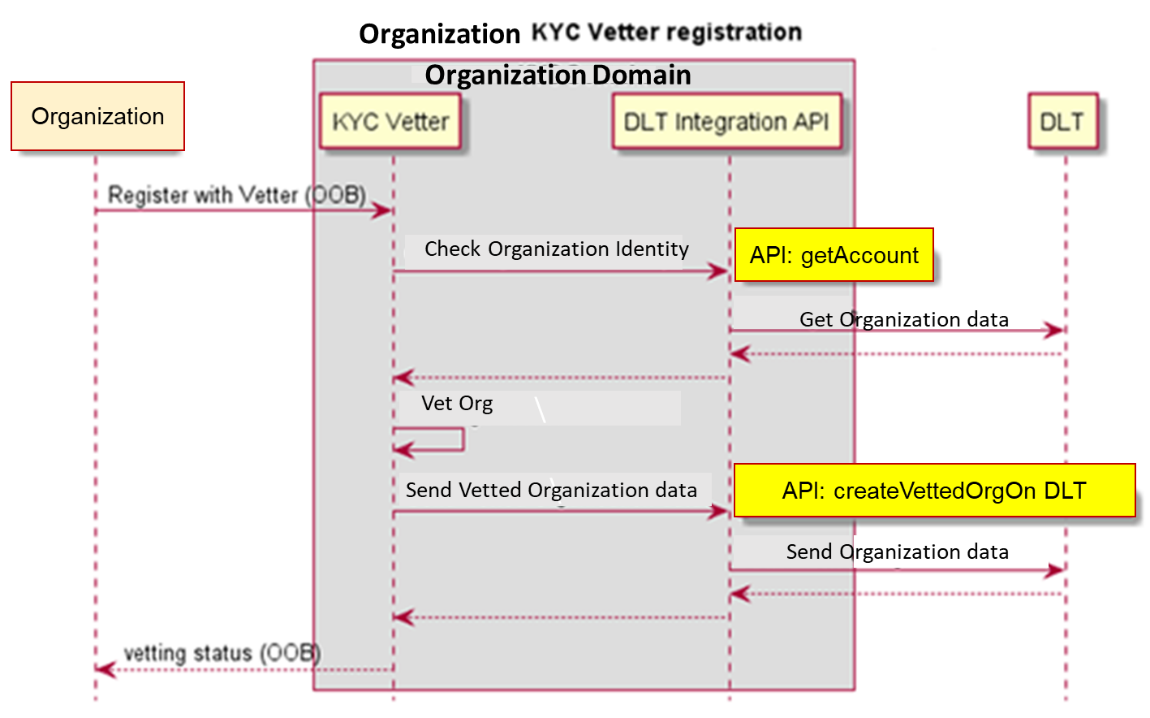
},

"proof": {...}

}

### Organization Vetting API

Using the Enterprise Identity API definition described in Annex A of this document, an organizations will request vetting from a vetter that will generate the VC claim to support the vetting status.

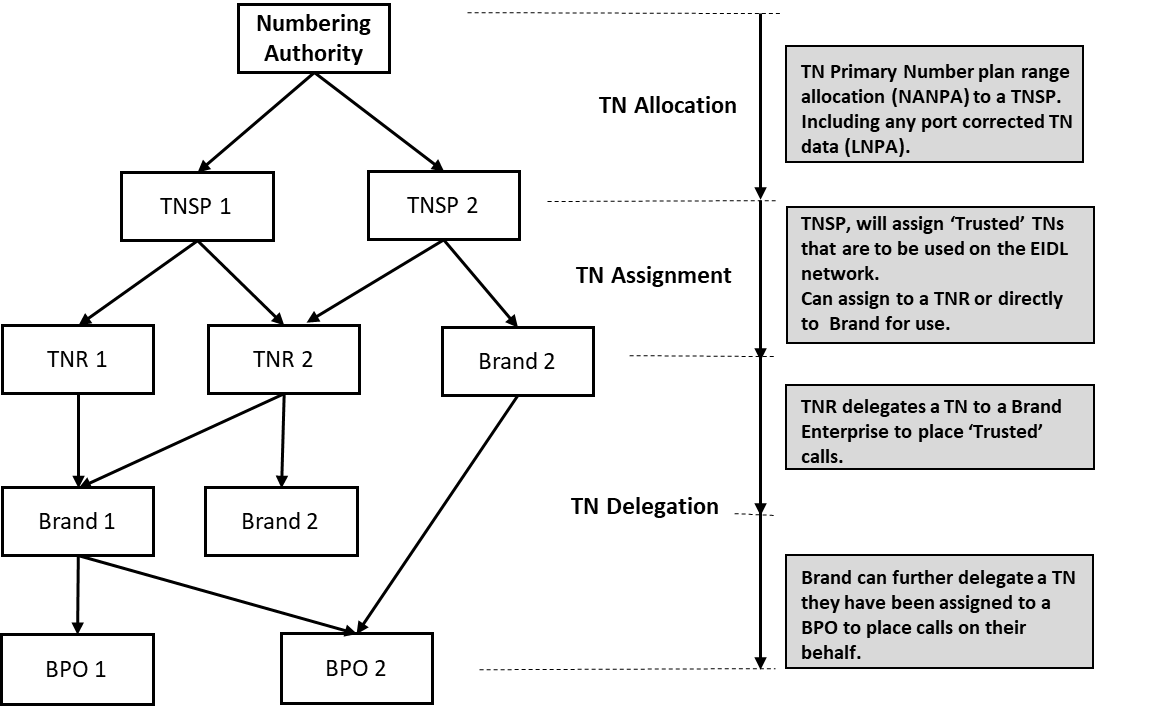


***Figure 5.5 API Flow for Out of Band (OOB) request to be vetted***

# TN Authorization

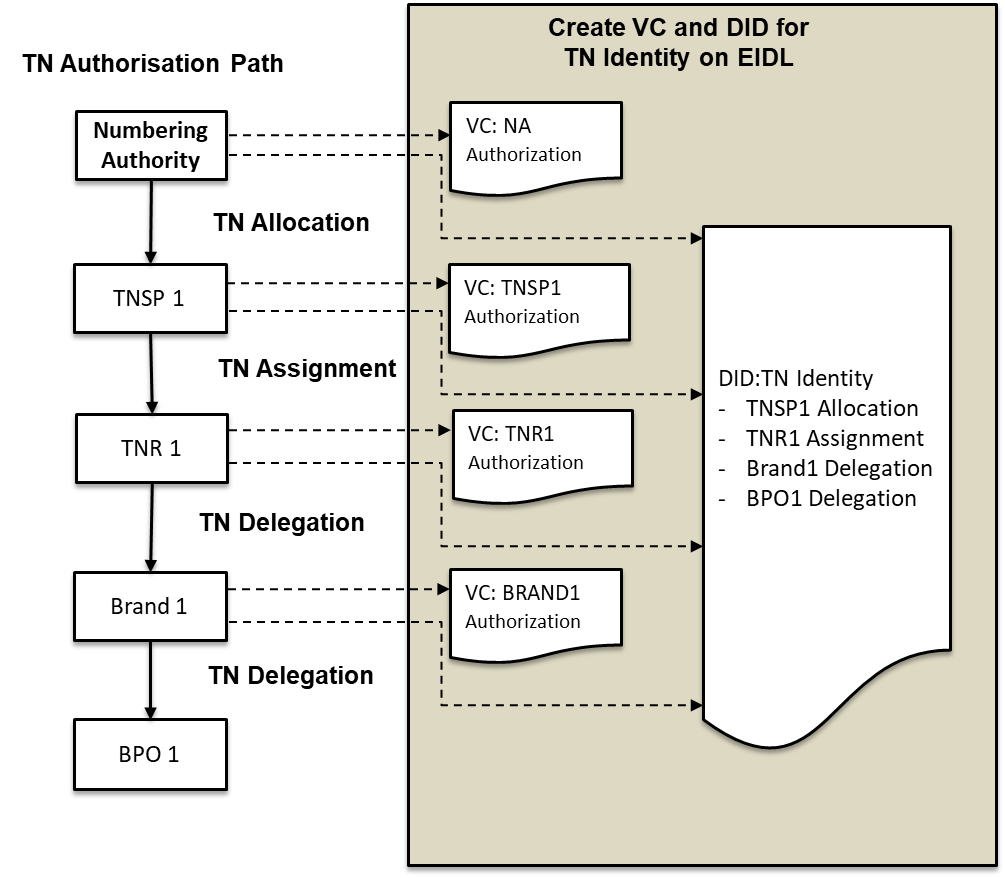
The authorized allocations (assignments or delegations) of 'Trusted' TNs from a trusted TNSP, assignee, or delegator of the TN to another 'Authorized' actor are represented in WC3 Verifiable Credential claim 'Proofs' on the EIDL Network.

The hierarchy for which TNs can be allocated from the top-level level Numbering Authority to a TNSP, assigned to a TNR, which can be delegated to a Brand Enterprise and then further delegated to a BPO, is outlined in figure 6.1. An allocation, assignment, or delegation can only be made to an actor with vetted status 'Authorized'. Each Layer of the TN allocation, assignment, or delegation must be contained within a TN Verifiable Credential Claim and recorded to the EIDL network. This will prove the authority of the TN has made the allocation, assignment, or delegation to the recipient organization identity, which is signed by the organization identify who is allocating, assigning, or delegating it to them.



***Figure 6.1 TN Authorization Hierarchy***

When an authorized actor makes a TN authorization, they will generate a corresponding verifiable credential proof of the authorized use. The first TN allocation of a TN identity will generate a TN identity DID used to verify and authenticate the use of the TN. As the TN is further assigned and delegated by actors, the DID for the TN identity is appended with the authorized verifiable credential claim. The DID for a TN identity represents the current chain of authorization for the given TN.



***Figure 6.2 TN Authorization process on EIDL***

Should the TN authorization be canceled or revoked; the verifiable credential status can be updated with the reason code to indicate that the TN is no longer authorized for the actor. The TN Identity DID is updated accordingly to show the chain of authorization that currently exists for the TN. In the event that this is from the top-level actor, i.e. number porting from one TNSP to another TNSP, the chain of custody to the new TNSP will require the creation of verifiable credential claims to support the new chain of authorization. These verifiable claims will need to be signed by each authorizing actor in the chain to support the new chain of authorization. See section below on TN cancellation and revocation.

## Allocation of a TN by the Numbering authority to a TNSP

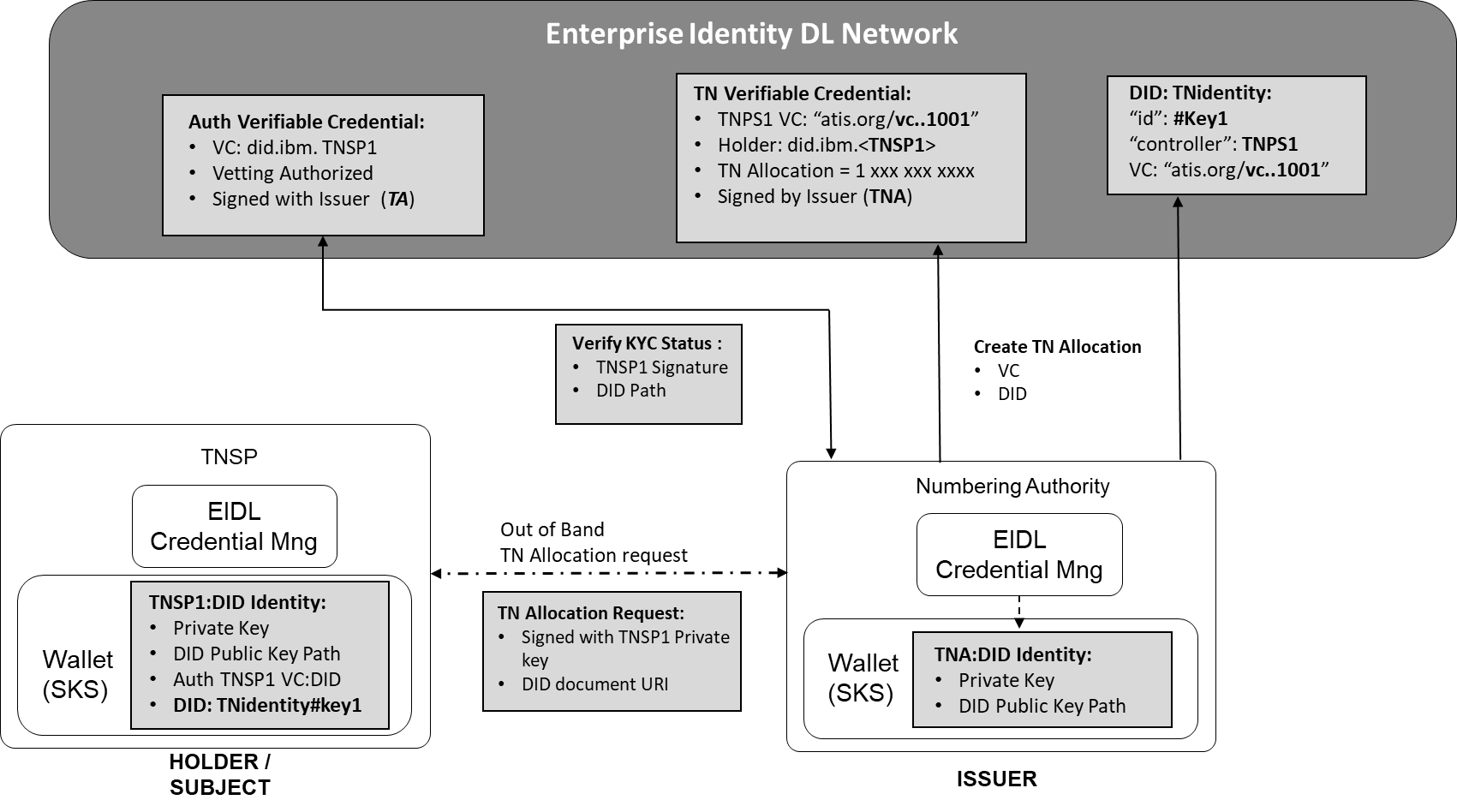
The Numbering Authority will only allocate 'Trusted' TNs to a TNSP on the EIDL network. TNs not being verified by the EIDL network are not allocated on the EIDL to the TNSP.

A TN allocation request is made by the TNSP to the Numbering Authority (the result of NANPA allocation or LNPA port request) as an out of band process, signing the request with the Enterprise Identity. The Numbering Authority will first verify that the TNSP is currently of the status '‘authorized/vetted' by checking the TNSP authorization VC claim, before allocating the TN to the TNSP.

If the TNSP is status '‘Authorized/Vetted', the Numbering Authority will create a Verifiable Credential claim for the specific TN, that is signed by the Numbering Authority and recorded on the EIDL network for proof. They then generate the DID supporting the TN identity indicating that the TNSP as a holder to use this identity.

If the TNSP Status is not '‘Authorized/Vetted', then the TN cannot be allocated.

.



***Figure 6.3 Numbering Authority TN Allocation – Verifiable Credential***

**Numbering Authority TN Allocation Verifiable Credential Claim**

Example of a W3C verifiable credential claim representing the TN allocation by the Numbering Authority to a TNSP.

{

"@context": [

"https://www.w3.org/2018/credentials/v1",

"https://access.atis.org/apps/group\_public/dlt/v1/"

],

"id": "**https://access.atis.org/credentials/1001**",

"type": ["VerifiableCredential", "TN Authorization"],

"holder": "did:ibm:<**TNSP1**>",

"issuer": "did:ibm:<**TNA**>",

"issuanceDate": "2020-01-01T19:73:24Z",

"credentialSubject": {

"id": "did:iota:<**TNIDENTITY**>",

"TN Authorization": {

"TN": "**12002001234**",

"Type": "**Allocation**",

}

},

"credentialStatus": {

"id": "https:// access.atis.org /status/24,

"type": "CredentialStatusList2017"

},

"proof": {...}

}

**Creating the TN Identity DID with the TN Allocation**

Example of a W3C DID Document used to verify and authenticate the TN allocation to the TNSP.

{

"@context": [

"https://www.w3.org/2018/credentials/v1",

"https://access.atis.org/apps/group\_public/dlt/v1/"

],

"id":"did:ibm:<**TNIDENTITY**>",

“authentication" :[{

“id”:”did:ibm:<**TNIDENTITY#Key1**>”,

“controller”:”did.ibm:<**TNPS1**>”,

"publicKey": ""H3C2AVv……."

}],

"Service" :{

“id”:”did:ibm:<**TNIDENTITY#Key1**>”,

“type": "**TN Authorization**",

"tn":"**12002001234**",

“serviceEndpoint":**"https://access.atis.org/credentials/1001"**

},

"proof": {...}

}

## Assignment of a TN by TNSP to TNR

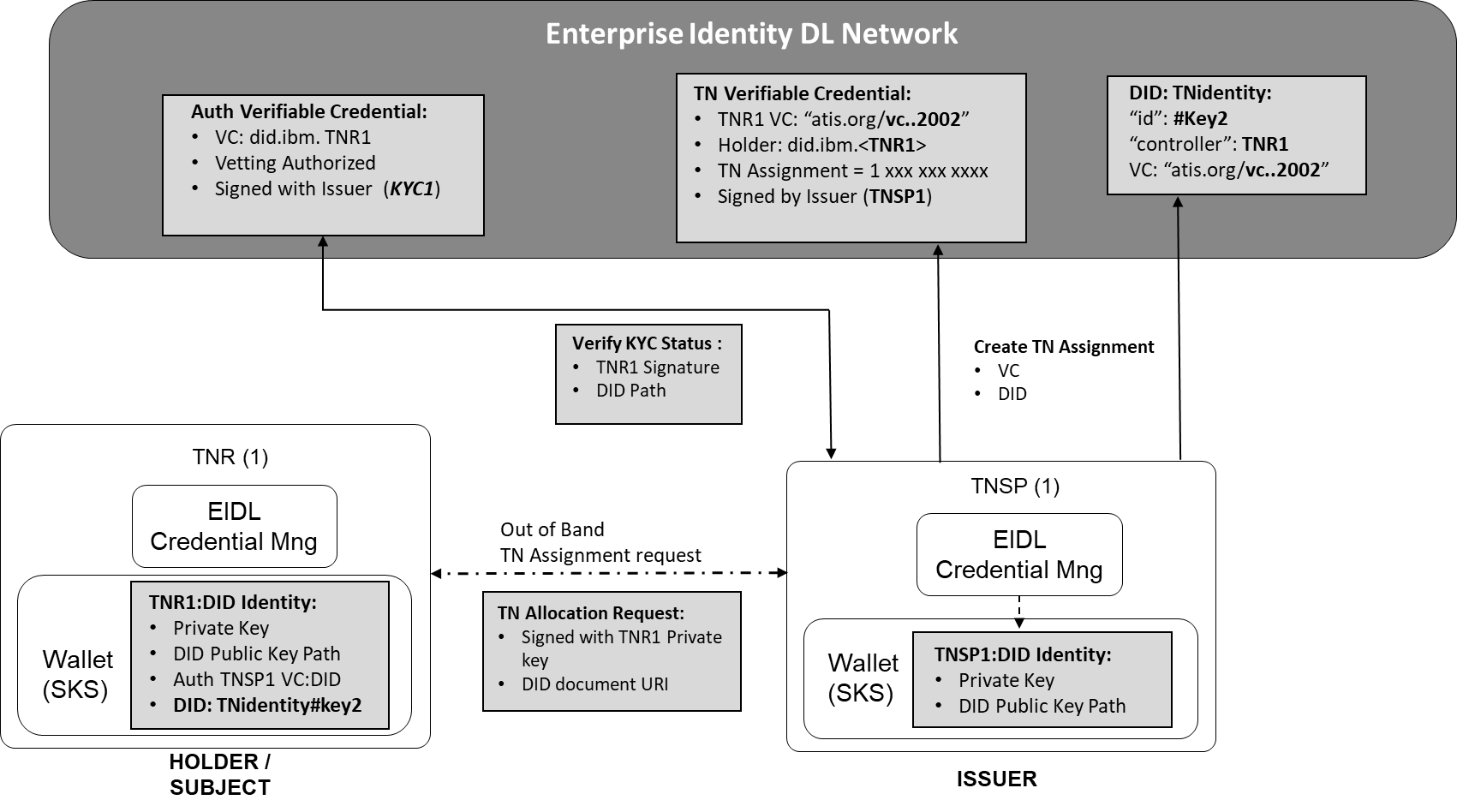
A TN assignment request is made by the TNR to the TNSP as an out of band process, signing the request with the Enterprise Identity.

***Note:*** The TNSP can only assign a TN that is under their authorized control, by a supporting active VC claim.

The TNSP will verify that the TNR is currently of the status '‘authorized/vetted' by checking the TNR authorization VC claim, before allocating the TN to the TNR.

If the TNR is status '‘Authorized/Vetted', the TNSP will create a Verifiable Credential claim for the specific TN assignment to the TNR, signed by the TNSP, and recorded on the EIDL network for proof. The TN Identity DID is updated with a key to indicate that the TNR is now a holder and can use this identity.

If the TNR Status is not '‘Authorized/Vetted', then the TN cannot be delegated.



***Figure 6.4 TNSP to TNR TN Assignment – Verifiable Credential***

**TNSP TN Assignment Verifiable Credential Claim**

Example of a W3C verifiable credential claim representing the TN assignment by a TNSP to a TNR.

{

"@context": [

"https://www.w3.org/2018/credentials/v1",

"https://access.atis.org/apps/group\_public/dlt/v1/"

],

"id": **"https://access.atis.org/credentials/2002"**,

"type": ["VerifiableCredential", "TN Authorization"],

"holder": "did:ibm:<**TNR1**>",

"issuer": "did:ibm:<**TNSP1**>",

"issuanceDate": "2020-01-01T19:73:24Z",

"credentialSubject": {

"id": "did:iota:<**TNIDENTITY**>",

"TN Authorization": {

"TN": "**12002001234**",

"Type": "**Assignment**",

}

},

"credentialStatus": {

"id": "https:// access.atis.org /status/24,

"type": "CredentialStatusList2017"

},

"proof": {...}

}

**Adding the TN Assignment to the TN Identity DID**

Example of a W3C DID Document used to verify and authenticate the TN allocation to the TNR.

{

"@context": [

"https://www.w3.org/2018/credentials/v1",

"https://access.atis.org/apps/group\_public/dlt/v1/"

],

"id": "did:ibm:<**TNIDENTITY**>",

“authentication" :[{

“id”:”did:ibm:<**TNIDENTITY#Key1**>”,

“controller”:”did.ibm:<**TNPS1**>”,

"publicKey": ""H3C2AVv……."},{

“id”:”did:ibm:<**TNIDENTITY#Key2**>”,

“controller":" did.ibm:<**TNR1**>",

“level":" **TopRestricted**",

"publicKey": ""LMv6g…………." }],

"Service" :[{

“id”:”did:ibm:<**TNIDENTITY#Key1**>”,

“type": ""**TN Authorization**",

"tn":"**12002001234**",

“serviceEndpoint":**"https://access.atis.org/credentials/1001"** },{

“id”:”did:ibm:<**TNIDENTITY#Key2**>”,

“type”: "**TN Authorization**",

"tn":"**12002001234**",

“serviceEndpoint”:**”https://access.atis.org/credentials/2002”**

}],

"proof": {...}

}

## Delegation of a TN by TNR to Brand / Brand to BPO

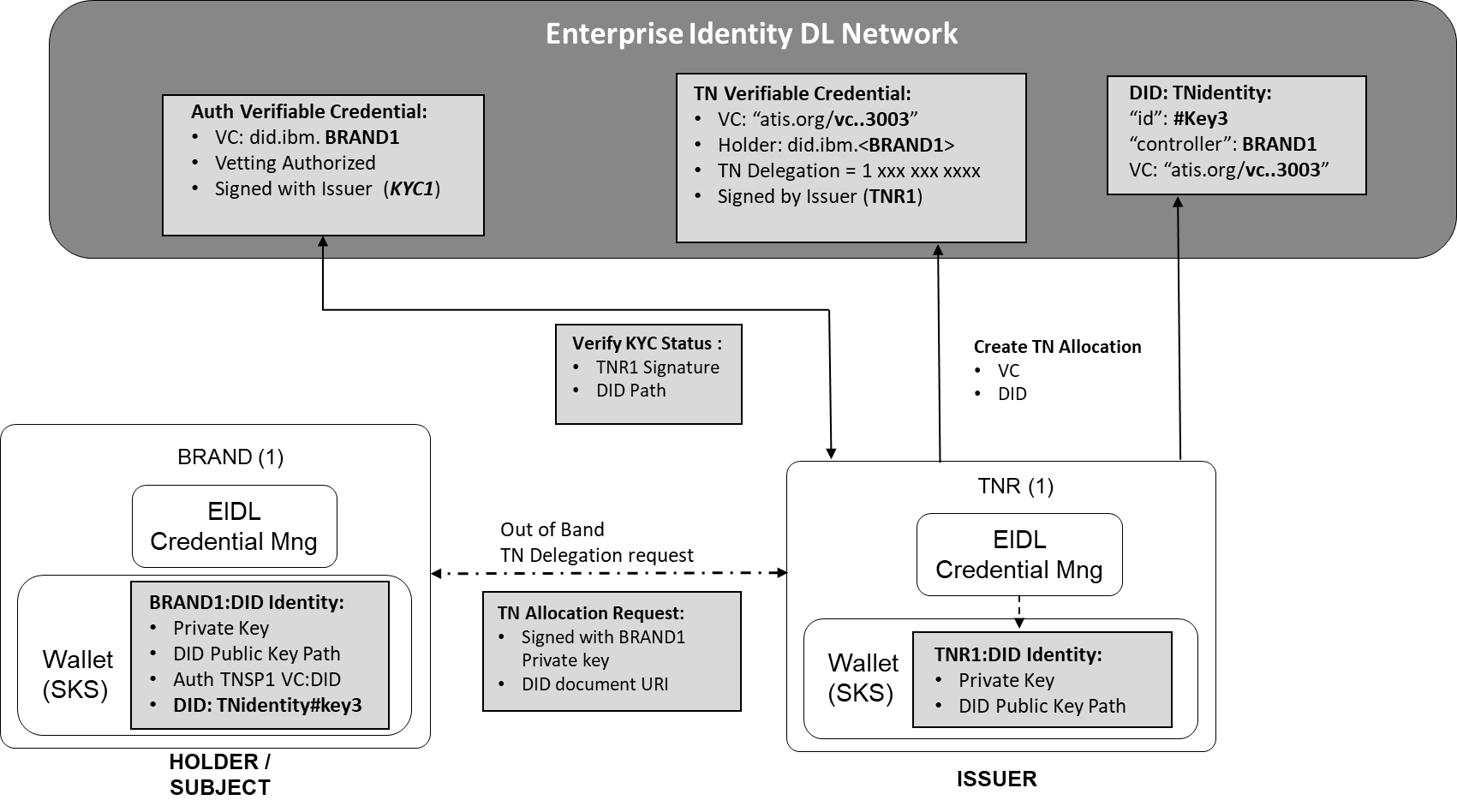
A TN delegation request is made by the BRAND to the TNR or BRAND to BPO, as an out of band process, signing the request with the Enterprise Identity.

***Note:*** The TNR can only delegate a TN that is under their authorized control, by a supporting active VC.

The TNR will verify that the BRAND is currently of the status '‘authorized/vetted' by checking the BRAND authorization VC claim, before allocating the TN to the BRAND.

If the BRAND is status '‘Authorized/Vetted', the TNR will create a Verifiable Credential claim for the specific TN assignment to the BRAND, signed by the TNR, and recorded on the EIDL network for proof. The TN Identity DID is updated with a key to indicating that the BRAND is now a holder to use this identity.

If the BRAND Status is not '‘Authorized/Vetted', then the TN cannot be delegated.



***Figure 6.5 TNR to BRAND TN Delegation – Verifiable Credential***

**TNR TN Delegation Verifiable Credential Claim**

Example of a W3C verifiable credential claim representing the TN delegation by a TNR to a BRAND enterprise. In the claim by the TNR to the BRAND, the claims may include the brand identity and intended reason for the call. It is possible to include within the verifiable claim other constraints about the use of the TN. In the example below the TN is allocated the Brand ‘*Big Bank’* for the intended purpose of making calls for *‘Loan Application’*.

{

"@context": [

"https://www.w3.org/2018/credentials/v1",

<https://access.atis.org/apps/group_public/dlt/v1/> ],

"id": **"https://access.atis.org/credentials/3003"**,

"type": ["VerifiableCredential", "TN Authorization"],

"holder": "did:ibm:<**BRAND1**>",

"issuer": "did:ibm:<**TNR1**>",

"issuanceDate": "2020-01-01T19:73:24Z",

"credentialSubject": {

"id": "did:iota:<**TNIDENTITY**>",

"TN Authorization": {

"TN": "**12002001234**",

"Type": "**Delegation**",

“CallerName: “**Big Bank**”,

“Purpose": “**Loan Application**"

}},

"credentialStatus": {

"id": "https:// access.atis.org /status/24",

"type": "CredentialStatusList2017"},

"proof": {...}

}

**Adding the TN Delegation to the TN Identity DID**

Example of a W3C DID Document used to verify and authenticate the TN allocation to the BRAND.

{

"@context": [

"https://www.w3.org/2018/credentials/v1",

"https://access.atis.org/apps/group\_public/dlt/v1/"

],

"id": "did:ibm:<**TNIDENTITY**>",

“authentication" :[{

“id”:”did:ibm:<**TNIDENTITY#Key1**>”,

“controller”:”did.ibm:<**TNPS1**>”,

"publicKey": "H3C2AVv……."

},{

“id”:”did:ibm:<**TNIDENTITY#Key2**>”,

“controller":" did.ibm:<**TNR1**>",

“level":" **TopRestricted**",

"publicKey": "LMv6g…………."

},{

“id”:”did:ibm:<**TNIDENTITY#Key3**>”,

“controller”:”did.ibm:<**BRAND1**>”,

“level":"**TopRestricted**",

"publicKey": "LMv6g…………."

}],

"Service" :[{

“id”:”did:ibm:<**TNIDENTITY#Key1**>”,

“type": ""**TN Authorization**",

"tn":"**12002001234**",

“serviceEndpoint":**" https://access.atis.org/credentials/1001"**

},{

“id”:”did:ibm:<**TNIDENTITY#Key2**>”,

“type”: "**TN Authorization**",

"tn":"**12002001234**",

“serviceEndpoint”:**”https://access.atis.org/credentials/2002”**

},{

“id”:”did:ibm:<**TNIDENTITY#Key3**>”,

“type”: "**TN Authorization**"

"tn":"**12002001234**",

“serviceEndpoint”:**”https://access.atis.org/credentials/3003”**

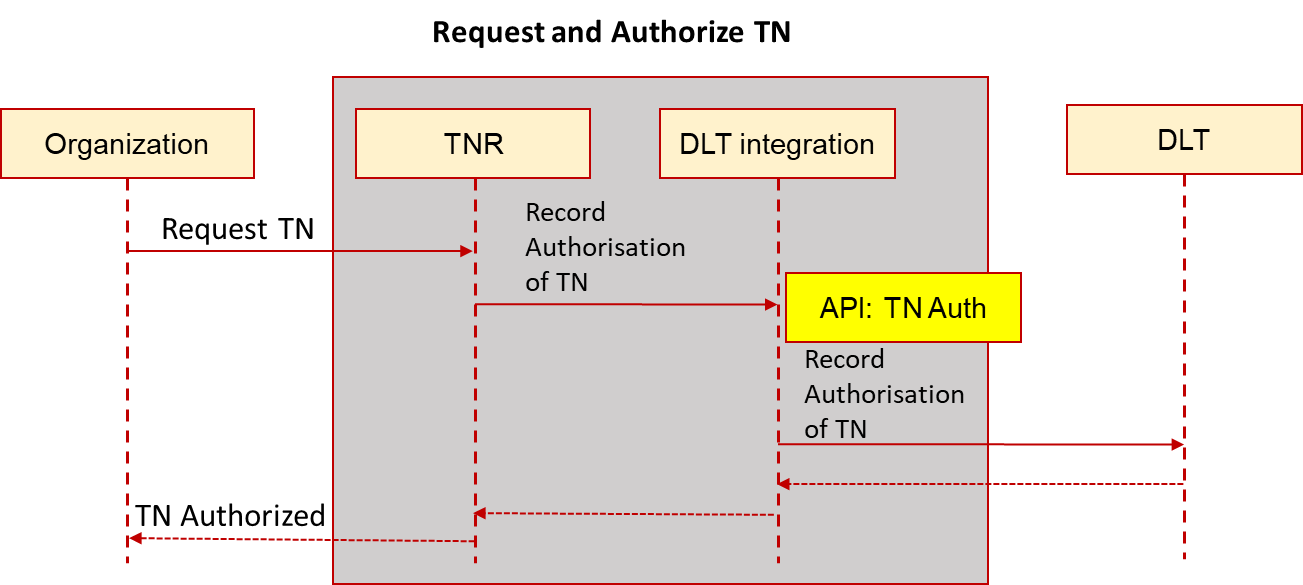
}],

"proof": {...}

}

### TN Authorization API

Using the Enterprise Identity API definition described in Annex A of this document, an organization will request the TN authorization from an assignee organization that will generate the verifiable credential claim to support the TN allocation / assignment / delegation on the EIDL.



***Figure 6.6 API Flow for TN Authorization***

## TN Cancel and Revocation - Credential Claim Status

When a TN allocation, assignment, or delegation credential needs to be canceled or revoked. A VC issuer can update the 'credentialStatus' property for the discovery of information about a verifiable credential. The ‘credentialStatus’ property for the discovery of information about a verifiable credential can be updated a VC issuer.

The following example of the credential claim status file is for a TN being ported from a TNSP1 and a canceled number from being used by TNR1 .

{

"id": "https://access.atis.org/status/24,

""description": ""Status of TN Allocation credentials."

"verifiableCredential": [{

"claim": {

"id": "**https://access.atis.org/credentials/1001**>",

"currentStatus": "**Revoked**",

"statusReason": "**Number Port**"

},

"issuer": " did:ibm:<**TNSP1**>",

"issued": "2020-02-04T14:27:42Z",

"proof": { ... }

}, {

"claim": {

"id": "**https://access.atis.org/credentials/2002**>",

"currentStatus": "**Revoked**",

"statusReason": "**Canceled**"

},

"issuer": " did:ibm:<**TNR1**>",

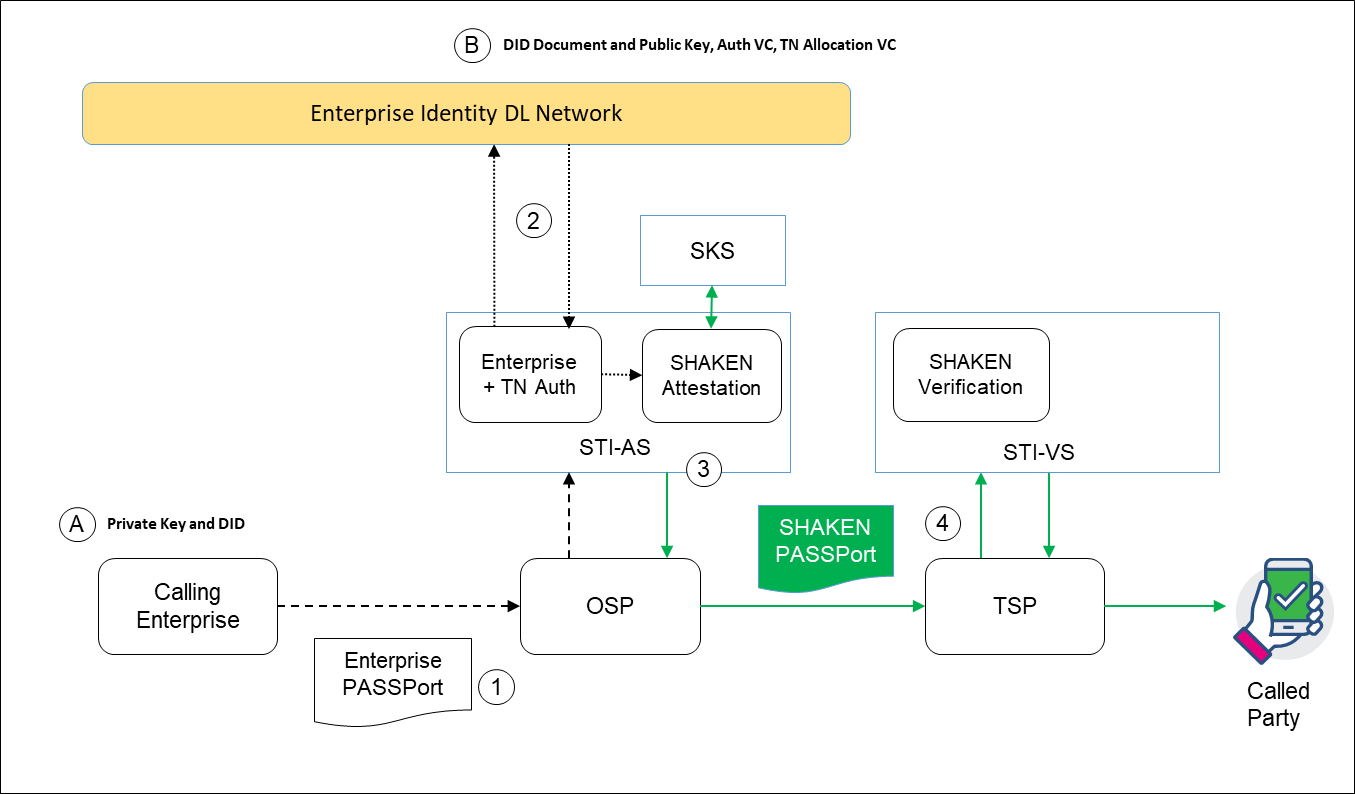
"issued": "2020-02-04T14:27:42Z",

"proof": { ... }

}]

# OSP Attestation of a call using EIDL Network

The sections describes how the EIDL network integrates with the STIR/SHAKEN Architecture to enable an OSP to use the Enterprise Identity to perform verification and authentication of the calling organization identity and proof that they have the authority to place calls with the called TN.



***Figure 7.1 Enterprise Identity from EIDL Network to provide OSP A Attestation***

The call scenario in figure 7.1 illustrates where a KYC vetted Enterprise originates a call to an OSP, including an ‘Enterprise PASSPorT’ token in a SIP Identity header. In this scenario, Where the OSP canis able to use the information included in the Enterprise PASSPorT token to qualify the caller identity and authorization to place this call from the called TN.

1. The Enterprise will use the Private Key to generate the Enterprise PASSPorT for the call being placed and include it in a SIP identity header. The PASSPorT will include the Enterprise DID.
2. The OSP adapted STI-AS will verify the ‘Enterprise PASSPorT' from the EIDL Network to authenticate the Enterprise and check the authorized use of the TN.
   1. Using the Enterprise DID Public Key to verify the token
   2. Verify the TN has assignment or delegation authorization to this enterprise identity from the VC.
3. If the 'Enterprise PASSPorT' is verified from the EIDL, then the STI-AS can mark the call with the “Full Attestation” level in a SIP Identity header containing a SHAKEN PASSporT delivered to an intermediate service provider or TSP according to the SHAKEN Specification ATIS-1000074.
4. The TSP will verify the SHAKEN PASSporT according to the SHAKEN Specification ATIS-1000074.

## Enterprise PASSPort Encoding

It is proposed that the ‘Enterprise PASSPorT’ token will be encoded using either a modified base PASSporT, as shown below, or by using an extended 'rcd' PASSporT with the same modifications. Note that the proposed modifications include a different reference type for the JWT authentication credentials ("jku" versus "x5u"), and an extension of the payload claims to include a reference to the TN authorization document on the DL ("tnauth" claim) as described in the following sub-clause.

### Use of modified base PASSPorT encoding for Enterprise Identity

This is an example of a modified base PASSporT encoded for Enterprise Identity and TN authorization.

Protected Header

{

"typ":"passport",

"alg":"ES256",

**“jku":”did:ibm:098765432abcdefghi”**

}

Payload

{

"dest":{"tn":"12125551213 "},

"iat":1443208345,

"orig":"tn":"12155551212"},

**“tnauth":“uri":["did:ibm:123456789hutnigk"]**

}

Description of the header fields and payload claims:-

The “jku" indicates the URI of the DID of the of the caller identity. Use "jku" URI format is defined in RFC 7515 sec 4.1.2

The 'orig' claim and 'dest' claim shall be of type 'tn'.

The "iat" to an encoding of the value of the SIP Date header field

The “tnauth”: claim is the DID URI to the signed Verifiable Credential claim to support the authorization to use the TN defined in the 'orig' claim

### Use of 'rcd' PASSporT encoding for Enterprise Identity

This is an example of an "rcd" extension PASSporT with modifications to resolve the Enterprise Identity and TN authorization:

Protected Header

{

"alg":"ES256",

"typ":"passport",

“ppt”:”rcd”,

**“jku":”did:ibm:098765432abcdefghi”**

}

Payload

{

"dest":{“tn”:"12155551213"}

"iat":1443208345,

"orig":{“tn”:"12155551212"},

"rcd":{"nam":"Dentist Office"}

**“tnauth":“uri":["did:ibm:123456789hutnigk"]**

}

Description of the header fields and payload claims:-

The “jku” indicates the URI of the DID of the of the caller identity. Use "jku" URI format is defined in RFC 7515 sec 4.1.2

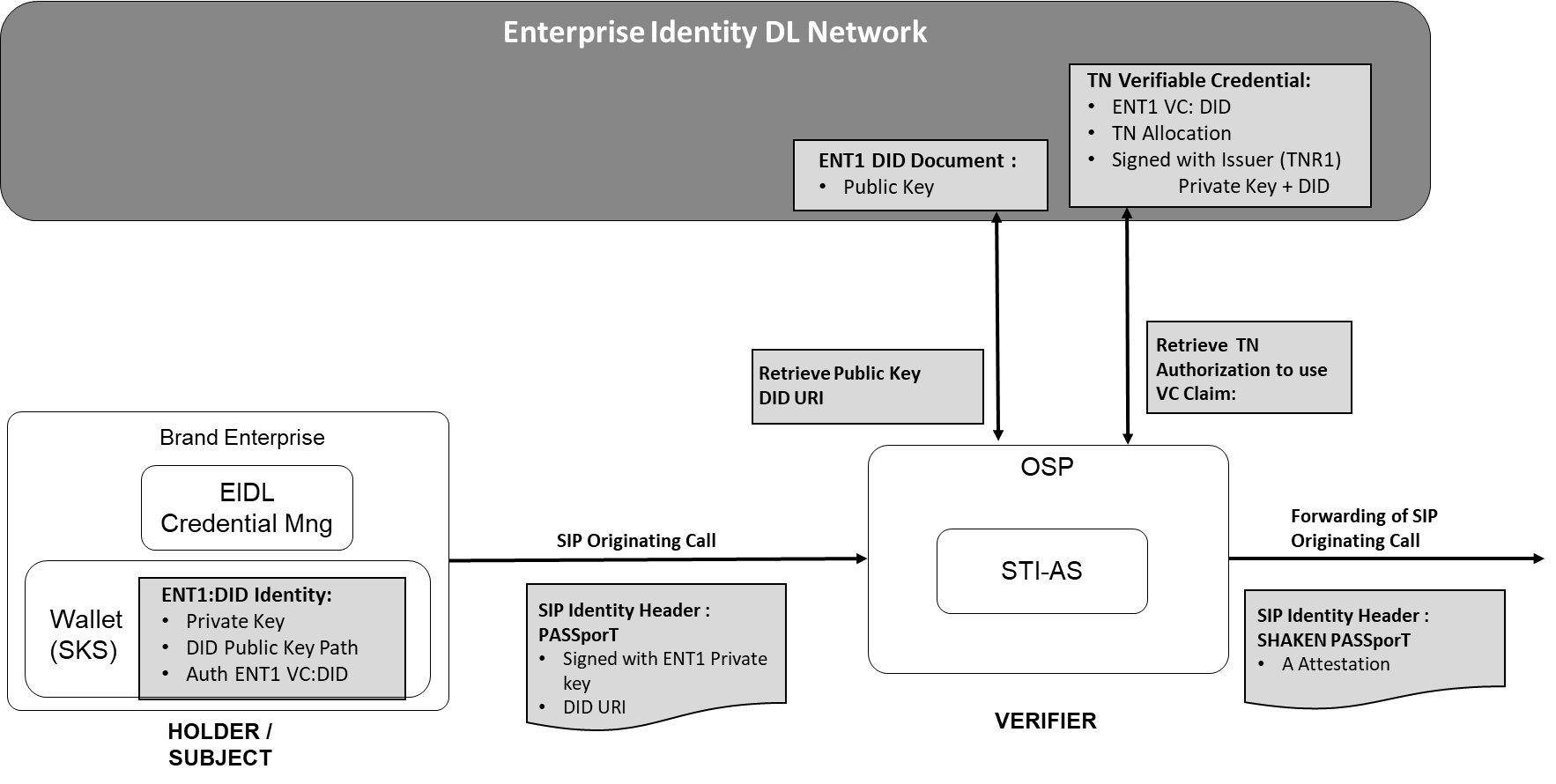
The 'orig' claim and 'dest' claim shall be of type 'tn'.

The ""iat"" to an encoding of the value of the SIP Date header field

The “tnauth”: claim is the DID URI to the signed Verifiable Credential claim to support the authorization to use the TN defined in the 'orig' claim.

## Verification of an Enterprise Identity and calling TN authorization by an OSP

When a caller places a SIP call to an OSP (directly or indirectly), they encode the PASSporT included in the SIP identity header as defined in section 7 of this document. The OSP receiving the call over a user-to-network interface from a customer will first validate the PASSporT by retrieving the caller's public key from the EIDL by the DID URI. The OSP will also verify that the used TN by the calling TN is authorized to before used by the caller DID by retrieving the Verifiable Credential for the TN being used. The OSP is able to both verify that the caller is who they say they are and has the authorization to place calls with the TN being used. Where these two are valid, the OSP can apply A attestation to the forwarding of the call over SHAKEN infrastructure.



***Figure 77.2 OSP validation of an originating caller and the used TN***

ANNEX A : Open API for Enterprise STIR/SHAKEN

# ANNEX A : Example of Open API for EIDL

This annex provided an example of the REST API definitions used in the ATIS Enterprise Identity DLT PoC, used to integrate organizations with the Enterprise Identity DLT Network solution. By using the *OpenAPI Specification (OAS)Specification Version 3.0.3* for defining the API, it is possible to integrate with the DLT using the following programming applications, Curl, Java, Android, Obj-C, JavaScript, C#, PHP, Perl or Python. For the purpose of the API described in this section the examples provided are using Curl.

***Note:*** The Enterprise Identity DL network OpenAPI definition format, is available to downloadable from ATIS (( Location TBD )

## Organization Accounts

These APIs are used for organization account integration with the DLT.

### createAccount

Create an account for an organization includes a TNSP, a TNR, a Brand Enterprise, a BPO, or a KYC Vetter.

This will generate an account identity and a Key Pair. These will be stored in a secure fashion by the organization client DLT Integration API. The DID will be returned in the response payload and will be used to identify the organization going forward in both the integration API and in the DLT

/accounts

**Usage and SDK Samples**

curl -X POST <http://localhost/v1/accounts>

**Parameters**

Body parameters

|  |  |
| --- | --- |
| **Name** | **Description** |
| body \* | {  An account to represent a given organization  Required: name,type  did: *string*  The DID/UUID of the account. This is assigned during the creation (POST) process and should be left blank for POSTs.  name: *string*  type: *integer (int32)*  Type of organization. 0 - KYC Vetter, 1 - TNR, 2 - Brand, 3 – BPO, 4 – TNSP, 5 – CSP, 6 - TA  publicKey: *string*  The DID document URI for the current public key for this account  createdTS: *integer (int64)*  Time that this account was created in UTC milliseconds from EPOCH  updatedTS: *integer (int64)*  Time that this account was last updated in UTC milliseconds from EPOCH  } |

**Responses**

**Status: 201 - Expected response to a valid request**

*Schema*

{

*An account to represent a given organization*

did: *string*

Decentralized identity of the account. This is the UUID assigned during the creation (POST) process. This is used going forward to identify the organization.

name: *string*

type: integer (int32)

Type of organization. 0 - KYC Vetter, 1 - TNR, 2 - Brand, 3 – BPO, 4 – TNSP, 5 – CSP, 6- TA

publicKey: *string*

The DID document URI reference on the DLT of the current public key for this account

createdTS:  *integer (int64)*

Time that this account was created in UTC milliseconds from EPOCH

updatedTS: *integer (int64)*

Time that this account was last updated in UTC milliseconds from EPOCH

}

**Status: default - unexpected error**

Schema

 {

Required: code,message

code: integer (int32)

message: string

}

### getAccount

Retrieve account information from the Enterprise Identity DL Network, based on the DID/UUID for the account.

/accounts/{orgIdParam}

**Usage and SDK Samples**

curl -X GET "http://localhost/v1/accounts/{orgIdParam}"

**Parameters**

Path parameters

|  |  |
| --- | --- |
| **Name** | **Description** |
| orgIdParam\* | String  *The DID/UUID of the organization*  Required |

**Responses**

**Status: 200 - Expected response to a valid request**

* [Schema](file:///H:\Innov8\ATIS\DLT%20PoC\API\API%201.0.2-SNAPSHOT.html#responses-getAccount-200-schema)

 {

*The account information to represent a given organization DID/UUID*

did: *string*

Decentralized identity of the account. This is the UUID assigned during the creation (POST) process. This is used going forward to identify the organization.

name: *string*

type: integer (int32)

Type of organization. 0 - KYC Vetter, 1 - TNR, 2 - Brand, 3 – BPO, 4 – TNSP,

5 – CSP, 6 - TA

publicKey: *string*

The DID document URI reference on the DLT of the current public key for this account

createdTS:  *integer (int64)*

Time that this account was created in UTC milliseconds from EPOCH

updatedTS: *integer (int64)*

Time that this account was last updated in UTC milliseconds from EPOCH

**Status: default - unexpected error**

Schema

 {

Required: code,message

code: integer (int32)

message: string

}

## TNAllocation

Allows an authorized organization to assign or delegate a TN to another organization on the Enterprise Identity DL Network.

***Note:*** ther term ALLOCATION in the API call refers to TN allocation from the Numbering AuthorityAuthorty to a TNSP, assignment of a TN fromtfrom a TNSP to a TNR , and Delegation of a TN to a Brand or BPO.

/tnallocation

**Usage and SDK Samples**

curl -X POST "http://localhost/v1/tnallocation"

**Parameters**

Body parameters

|  |  |
| --- | --- |
| **Name** | **Description** |
| body | *{*  *tnAllocation: [*  *{*  *Required: allocatedByOrganisationId,allocatedToOrganisationId,validEndDate,validStartDate*  tnId: *string*  *Identifier of the telephone number.*  issuerId: *string*  *DID of the Organization that is allocating the TN. T*  subjectId: *string*  *DID of the Organization that the TN is being allocated to.*  intendedUseOfAllocation: *string*  *Text field that allows you to specify the intended purpose the TNs in the allocation will be used for.*  tnIdType*: integer (int32)*  *Type of telephone number identifier. (0-E.164, 1-UUID, 2-DID, 3-Verifiable Credential). This field will allow us some flexibility as we grow the solution to allow different types of identifiers for TNs.*  *allocationStartDate:  string (date-time)*  *Time that this TN Allocation becomes a valid allocation. Must be a future date.*  allocationEndDate*:  string (date-time)*  *Time that this TN Allocation will be revoked. Must be a future date. If this is not specified, it is assumed there is no end date*  *}*  *]*  *}* |
|  |  |

**Responses**

**Status: 201 - Return the newly created TNAllocation**

* [Schema](file:///H:\Innov8\ATIS\DLT%20PoC\API\API%201.0.2-SNAPSHOT.html#responses-tnAllocation-201-schema)

 {

tnAllocationResponse:[

{

verifiableCredential:

  {

@context:[

*Sets the context, which establishes the special terms we will use like issuer and alumniOf*

id:  string

*Identifier for the credential*

type: string

*Credential types, which declare what data to expect in the credential*

issuer: string

*Entity DID that issued the credential*

issuanceDate:  string (date-time)

*Time that this verifiable credential was issued*

}

proof:

 {

*Digital proof that makes the credential tamper-evident.*

type:   string

*The cryptographic signature suite that was used to generate the signature*

created: string (date-time)

*Time that this signature was created.*

proofPurpose:  string

*Purpose of this proof*

verificationMethod:  string

*The identifier of the public key that can verify the signature*

jws:  string

*The digital signature value*

}

**all of:**

 {

Required: allocatedByOrganisationId,allocatedToOrganisationId,validEndDate,validStartDate

tnId: *string*

*Identifier of the telephone number.*

issuerId: *string*

*DID of the Organization that is allocating the TN. T*

subjectId: *string*

*DID of the Organization that the TN is being allocated to.*

intendedUseOfAllocation: *string*

*Text field that allows you to specify the intended purpose the TNs in the allocation will be used for.*

tnIdType*: integer (int32)*

*Type of telephone number identifier. (0-E.164, 1-UUID, 2-DID, 3-Verifiable Credential). This field will allow us some flexibility as we grow the solution to allow different types of identifiers for TNs.*

*allocationStartDate:  string (date-time)*

*Time that this TN Allocation becomes a valid allocation. Must be a future date.*

allocationEndDate*:  string (date-time)*

*Time that this TN Allocation will be revoked. Must be a future date. If this is not specified, it is assumed there is no end date*

*}*

]

}

**Status: default - unexpected error**

Schema

 {

Required: code,message

code: integer (int32)

message: string

}

## Vetter

### createVettedOrganisationOnDLT

Allows a KYC vetter to update the vetted organization authorization status to the DLT

/vetterAUthorization/{orgIdParam}/vetter/{vetterIdParam}

**Usage and SDK Samples**

curl -X POST "http://localhost/v1/vetterAUthorization/{orgIdParam}/vetter/{vetterIdParam}"

**Parameters**

Path parameters

|  |  |
| --- | --- |
| **Name** | **Description** |
| orgIdParam\* | String  *The DID/UUID of the subject organization vetted status change*  Required |
| vetterIdParam\* | String  *The DID/UUID of the vetter*  Required |

Body parameters

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | |
| body \* | }  status:  integer (int32)  *Status of the vetting process. 0 - In Progress, 1 - Complete*  result:  integer (int32)  *Result of the vetting process. 0 - Authorized, 1 - Not Authorized, 2 - Blacklisted*  vettedTS:  string (date-time)  *Time that this authorization became effective*  version: integer (int32)  *In the case that an authorization is updated on the ledger, we will update the version. This will help to identify and order related authorizations.*  } |

**Responses**

**Status: 201 - Null response**

Schema

 {

Required: organization ,status ,vetter

organization:  {

*The DID/UUID of the subject organization vetted status change*

did:   string

Decentralized *identity of the account. This is assigned during the creation (POST) process. This is used going forward to identify the organization.*

name: string

type:  integer (int32)

*Type of organization. 1 - TNR, 2 - Brand, 3 – BPO*

createdTS:  string (date-time)

*Time that this account was created*

updatedTS:  string (date-time)

*Time that this account was last updated*

}

vetter:  {

*An account to represent a given organization*

did:  string

*Decentralized identity of the TA / KYC vetter account.*

name: string

type:  integer (int32)

*Type of organizationorganisation. 0 - KYC Vetter, 6 - TA*

createdTS:  string (date-time)

*Time that this account was created*

updatedTS: string (date-time)

*Time that this account was last updated*

}

status:  integer (int32)

*Status of the vetting process. 0 - In Progress, 1 - Complete*

result:  integer (int32)

*Result of the vetting process. 0 - Authorized, 1 - Not Authorized, 2 - Blacklisted*

vettedTS:  string (date-time)

*Time that this authorization became effective*

version: integer (int32)

*In the case that an authorization is updated on the ledger, we will update the version. This will help to identify and order related authorizations.*

}

**Status: default - unexpected error**

Schema

 {

Required: code,message

code: integer (int32)

message: string

}

### getVettedOrganisationAuthorization

Retrieve the vetted authorization status and history for a given organization

/vetterAUthorization/{orgIdParam}/vetter/{vetterIdParam}

**Usage and SDK Samples**

curl -X GET "http://localhost/v1/vetterAUthorization/{orgIdParam}/history"

**Parameters**

Path parameters

|  |  |
| --- | --- |
| **Name** | **Description** |
| body\* | Id: String  *The DID/UUID of the subject organization vetted status change*  Required  status: integer(int32)  *Type of status response . 0 - Current, 1 - History* |
|  |  |

**Responses**

**Status: 201 - Null response**

Schema

 {

Required: organization ,status ,vetter

organization:  {

*The DID/UUID of the subject organization vetted status change*

did:   string

*Decentralizedidentity of the account. This is assigned during the creation (POST) process. This is used going forward to identify the organization.*

name: string

type:  integer (int32)

*Type of organization. 1 - TNR, 2 - Brand, 3 - BPO*

createdTS:  string (date-time)

*Time that this account was created*

updatedTS:  string (date-time)

*Time that this account was last updated*

}

**all of:[ NOTE: :List authorization history if Status = 1(History )**

vetter:  {

*An account to represent a given organization*

did:  string

*Distributed identity of the TA/ KYC vetter account.*

name: string

type:  integer (int32)

*Type of organizationorganisation. 0 - KYC Vetter, 6 - TA*

createdTS:  string (date-time)

*Time that this account was created*

updatedTS: string (date-time)

*Time that this account was last updated*

}

status:  integer (int32)

*Status of the vetting process. 0 - In Progress, 1 - Complete*

result:  integer (int32)

*Result of the vetting process. 0 - Authorized, 1 - Not Authorized, 2 - Blacklisted*

vettedTS:  string (date-time)

*Time that this authorization became effective*

version: integer (int32)

*In the case that an authorization is updated on the ledger, we will update the version. This will help to identify and order related authorizations.*

]

}

**Status: default - unexpected error**

Schema

 {

Required: code,message

code: integer (int32)

message: string

}

1. https://www.w3.org/TR/did-core/ [↑](#footnote-ref-2)
2. Available from the Internet Engineering Task Force (IETF) at: < <https://www.ietf.org/> >. [↑](#footnote-ref-3)
3. Available from 3rd Generation Partnership Project (3GPP) at: < [https://www.3gpp.org](http://www.3gpp.org) > [↑](#footnote-ref-4)
4. Available from W3C at : <<https://www.w3.org/>> [↑](#footnote-ref-5)
5. <https://www.w3.org/TR/did-core/#definition> [↑](#footnote-ref-6)
6. <https://www.w3.org/TR/vc-use-cases/> [↑](#footnote-ref-7)