ATIS-I-0000084

ATIS Technical Report on

ATIS-I-0000084: Enterprise Identity Distributed Ledger Network

Providing Enterprise and Telephone Number Allocation Authentication for Originating Service Provider SHAKEN Attestation

Alliance for Telecommunications Industry Solutions

Approved July 15th, 2021

Abstract

Consumers worldwide have been inundated with illegal robocalls and other unwanted calls to the point that by some estimates, a large percentage no longer answer their phones. This makes it difficult for legitimate businesses to reach consumers in a timely manner with important information, alerts and reminders. Those businesses also bear the expense of follow-up calls and other outreach. Meanwhile, service providers bear costs such as fielding customer complaints and investigating sources of fraudulent traffic.

This Technical Report describes a specification that extends the capabilities of SHAKEN to provide the ecosystem with new options for mitigating illegal robocalls. In addition, it features a distributed ledger infrastructure called the enterprise identity network.

The specification enables an enterprise to establish enterprise identity credentials by applying distributed ledger technology and its cryptographic principles. The enterprise then can place calls signed with its enterprise identity credentials, enabling any originating service provider (OSP) receiving the call to authenticate the enterprise identity of the calling enterprise. When the OSP can authenticate the calling enterprise identity and the originating TN is authorized for use, the OSP can apply SHAKEN A-level attestation to the call.

Foreword

As a leading technology and solutions development organization, the Alliance for Telecommunications Industry Solutions (ATIS) brings together the top global ICT companies to advance the industry's most pressing business priorities. ATIS' nearly 200 member companies are currently working to address the all-internet protocol (IP) transition, 5G, network functions virtualization, big data analytics, cloud services, device solutions, emergency services, M2M, cyber security, network evolution, quality of service, billing support, operations, and much more. These priorities follow a fast-track development lifecycle — from design and innovation through standards, specifications, requirements, business use cases, software toolkits, open-source solutions, and interoperability testing.

ATIS is accredited by the American National Standards Institute (ANSI). The organization is the North American Organizational Partner for the 3rd Generation Partnership Project (3GPP), a founding partner of the oneM2M global initiative, a member of and major U.S. contributor to the International Telecommunication Union (ITU), as well as a member of the Inter-American Telecommunication Commission (CITEL). For more information, visit www.atis.org.

The Alliance for Telecommunications Industry Solutions (ATIS) serves the public through improved understanding between carriers, customers, and manufacturers. The ATIS DLT Focus Group was 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.

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.

Table of Contents

Table of Figures

1 Scope & Problem Statement

1.1 Scope

This specification extends the capabilities of Signature-based Handling of Asserted information using toKENs ((SHAKEN)) to enable an enterprise to establish enterprise identity credentials by applying distributed ledger technology and its cryptographic principles. A Know Your Customer (KYC) verified enterprise identity allows the enterprise to prove its identity to any participant of the VoIP ecosystem to be assigned or delegated trusted telephone numbers (TNs)) by authorized telephone number service providers (TNSPs) or telephone number resellers (TNRs). Using a KYC-verified enterprise identity credential with an authorized telephone number, the enterprise can place calls signed with its enterprise identity credentials, enabling any originating service provider (OSP) receiving the call to authenticate the enterprise identity credential of the calling enterprise. The OSP can also verify the signing entity's authorization for using the TN from the distributed ledger to mark a call with the SHAKEN “Full” or “A-level” attestation. This distributed ledger infrastructure is called the Enterprise Identity Distributed Ledger Network (EIDLN).

The enterprise identity credential is a W3C Decentralized identifier (DID) recorded on the distributed ledger, authenticated by public/private key pair cryptography. The proof that an enterprise identity has been KYC vetted will be recorded on the distributed ledger by the issuing authority using signed, verifiable credentials (VCs), recorded on the distributed ledger according to the W3C Verifiable Credential format.

All authorized TN assignments or delegations will be recorded on the distributed ledger by the issuing authority using signed VCs recorded on the distributed ledger according to the W3C Verifiable Credential format.

An enterprise will create a SIP identity header on its outgoing calls containing a Personal Assertion Token (PASSporT)) signed with its enterprise identity private key and 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 for use by the enterprise identity by checking the signed VC for the TN.

When the OSP can authenticate the calling enterprise identity and the originating TN is authorized for use, the OSP can apply SHAKEN A-level attestation to the call.

1.2 Problem Statement

The Federal Trade Commission (FTC) and Federal Communications Commission (FCC) regularly cite “unwanted and illegal robocalls” as their No. 1 complaint category. Caller ID spoofing increases the harm from these unwanted calls. The ATIS-SIP Forum IP-NNI Task Force has developed a specification, SHAKEN, explicitly designed to mitigate unsolicited and unwanted robocalls by reducing the impact of illegitimate caller ID spoofing. SHAKEN is based on the Secure Telephone Identity Revisited (STIR) protocol developed by IETF. The SHAKEN specification allows the OSP to generate a digital signature that securely signals the caller's right to use a phone number to the TSP.

When a call is originated, the OSP will use the SIP identity header to contain a digital signature or token that will accompany the call as it is being completed. At call termination, the TSP 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 whose rating system tells the receiving caller whether the incoming number is good, questionable or likely fraudulent. The app also can act on the user’s behalf to automatically stop unwanted calls from getting through.

SHAKEN was never intended to be a complete solution for addressing the robocalling problem. Instead, it is an important tool in a multi-layered approach. SHAKEN specifically allows an OSP to populate a SIP identity header on its calls destined to other service providers. This SIP identity header 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 indicator (attest) 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 customer 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:

A. Is responsible for the origination of the call onto the IP-based service provider voice network.
B. Has a direct authenticated relationship with the customer and can identify the customer.
C. Has established a verified association with the TN used for the call.

Many call scenarios today will not satisfy these conditions through information locally available to the OSP. As a result, they may receive “partial attestation" (“B” Attestation level) at best. 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 may not be provided when the assignment and/or delegation of the TN is 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 TSPs to complete all calls. Note that the FCC call delivery regulations do 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 SPs 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 OSP knows about the call (customer, and potentially its right to use the number) and sends this signature to the TSP, which verifies that no one has tampered with the information. SHAKEN does not identify calls as spam. Call validation treatment (CVT) is the logical function 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. A few examples include flight or school cancellations, appointment reminders and credit card fraud alerts. However, current analytics deployments can mistakenly label these legal callers as fraud/scam.

As previously mentioned, these typical call scenarios are complex. An enterprise may contract with a call center provider, which will use one or more vendors for the calling platform, obtain TNs for the campaign from a TNR and originate the calls using multiple 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.

2 References

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

ATIS-10000074, *Signature-based Handling of Asserted Information using Tokens (SHAKEN)*[1]

ATIS-10000080, *SHAKEN: Governance Model and Certificate Management* [1]

ATIS-10000089v2, SHAKEN: Study of Full Attestation Alternatives for Enterprises and Business Entities with Multi-Homing and Other Arrangements[1]

ATIS-10000094, SHAKEN: Calling Name and Rich Call Data Handling Procedures[1]

ATIS-I-0000076, *Enterprise Identity on Distributed Ledger for Authenticated Caller Use Cases*[1]

OpenAPI, *OpenAPI Specification (OAS) Version 3.0.3*[2]

RFC 3261, *SIP: Session Initiation Protocol*[3]

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

RFC 4949, *Internet Security Glossary, Version 2*[2]

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

RFC 7095, *jCard: The JSON Format for vCard*[2]

RFC 7515, *JSON Web Signature (JWS)*[2]

W3C, Decentralized Identifiers (DIDs) v1.0[4]

W3C, Verifiable Credentials Data Model 1.0[3]

W3C, Verifiable Credentials Use Cases[3]

ITU-T Recommendation X.811 (04/1995) | ISO/IEC 10181-2:1996, Information technology – Open Systems Interconnection – Security frameworks for open systems: Authentication framework[5]

NIST SP 800-63-3 - NIST Special Publication 800-63-3 Digital Identity Guidelines[6]

CA/Browser Forum Guidelines for The Issuance and management of Extended Validation Certificates Version 1.6.8. [7]

---

[1] Available from ATIS at https://www.atis.org

[2] Available from Swagger at https://swagger.io/specification/

[3] Available from IETF at https://www.ietf.org

[4] Available from W3C at https://www.w3.org/TR/did-core/

[5] Available from ITU-T at https://www.itu.int/ITU-T/recommendations/rec.aspx?rec=3107

[6] Available from NIST at https://csrc.nist.gov/publications/detail/sp/800-63/3/final

[7] Available from CA/Browser Forum at https://cabforum.org/documents/ - Extended-Validation-Guidelines

3 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 >.

3.1 Definitions

BPO Enterprise: A business process outsource (BPO) organization (call center) placing calls on behalf of a brand enterprise.

Brand Enterprise: The business entity 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 TN used to identify the caller carried either in the P-Asserted-Identity or From header fields in the SIP [RFC 3261] messages.

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

Decentralized Identifiers: DIDs are new type of identifier that enables verifiable, decentralized DIDs. It can be authenticated using proofs such as digital signatures and privacy-preserving biometric protocols.

Distributed Ledger Technology: A protocol that enables the secure functioning of a decentralized digital database. Distributed networks eliminate the need for a central authority to keep a check against manipulation. DLT enables storage of all information in a secure and accurate manner using cryptography. Information is stored and verified on the DLT by using asymmetric-key cryptographic signatures. Once the information is stored, it becomes an immutable database and is governed by the rules of the network.

Decentralized Identifier Document: Also referred to as a DID document, it is accessible using a verifiable data registry. It 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 PKI. Usually it is a server, service, router or a person. In the context of this document, an end-entity is an SP, TNSP or VoIP entity.

Enterprise: For the purposes of this document, this is 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.

Identity: Unless otherwise qualified, an identifier that unambiguously distinguishes an entity for authentication and other security and policy application purposes (compare to the 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.

KYC Vetter: This is the business function used to verify a client's business identity. The KYC vetter is a trusted actor of the EIDLN managed by the EIDLN Governance Authority (GA) to perform verification services according to the EIDLN service's policy. The KYC process is to verify the identity of clients before doing business with those entities and on an ongoing basis with them. An EIDLN KYC vetter will perform the process of KYC on all TNRs, enterprise brands and BPOs with an enterprise identity to verify that they are compliant with KYC policies of the EIDLN governance. A KYC vetter provides a qualified enterprise identity that can be authenticated by all other actors of the EIDLN. This equates to the security management process of "Identity Proofing" as referenced in *NIST SP 800-63-3 - NIST Special Publication 800-63-3 Digital Identity Guidelines, and CA/Browser Forum Guidelines For The Issuance And Management Of Extended Validation Certificates Version 1.6.8).*

Numbering Authority: The entity designated as authoritative over the allocation of a TN to a TNSP for its assignment to end users or other customer entities such as TN resellers or VoIP providers. It is the North American Numbering Plan Administrator (NANPA) under contract to the FCC for full codes and thousands blocks. For toll-free numbers (TFNs), it is the Toll-Free Number Administrator (TFNA), which manages the TFN registry and the

allocation of TFNs to a RespOrg (for their assignment to end users or other customer entities such as TN resellers or other entities), under a tariff from the FCC. For local number portability, it is the Local Number Portability Administrator (LNPA), under contract from the North American Portability Management (NAPM), LLC., and with oversight from the North American Numbering Council (NANC).

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 include, but are not limited to, personal or business name, physical location or postal address, government-issued identifiers and credit information. Compare to the use of “real-life identity” and “real-world identity” in [NIST SP 800-63-3].

RespOrg: This entity is designated as the agent for the toll-free subscriber to obtain, manage and administer TFNs and provide routing reference information in the SMS/800 Toll-Free Number Registry. RespOrgs are the only parties that assign, manage and administer TFNs in the Toll-Free Number Registry.

RespOrg Identification (RespOrg ID): A five-character code that designates or points to the RespOrg associated with a specific TFN [ATIS-0417001-003].

Rich Call Data: The mechanism used for authentication, verification, and transport of calling name and other enhanced caller identity information (e.g., images, logos) and call reason, and describing how they are handled in various call origination and termination scenario, using the rcd PASSporT extension with "rcd" claims.

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 U.S. and Canada, this would be a Company Code as defined in [ATIS-0300251], or a RespOrg ID assigned to a RespOrg as defined in [ATIS-0417001-003].

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 TN 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, TFNA).

TN Assignment: The procedure for a TNSP to grant the right to use a block or individual TN to a TNR, brand enterprise or other entity.

TN Delegation: The procedure for a TNSP, TNR, brand enterprise or other TN assignee to grant the right to use an individual TN to a brand enterprise, BPO or other entity.

Toll-Free Number Administrator (TFNA): An entity that is authoritative over the assigning, reserving and releasing of TFNs for public use.

Toll-Free Number Registry (TFNR): The main administrative support system of toll-free service. It is used to create and update subscriber toll-free records that are then downloaded to service control points (SCPs) for handling subscriber’s toll-free calls. RespOrgs also use the system to reserve and assign TFNs.

Trust Authority: The TA in the enterprise identity network is an assigned third party providing three functions:

- Vetting and authorization of the enterprise identity TNSPs and KYC vetters.
- Vetting and authorization of the EIDLN (EIDLN) nodes.
- Implementing the TN authorization rules and process on the EIDLN.

VCs: These are the electronic equivalent of the physical credentials such as plastic ID and credit cards, passports, driver's licenses, qualifications and award. Verifiable credentials (VC) may be expressed using JSON. A VC is typically composed of the context of the VC, the identity of the issuer, the date and time of issuing, the expiration date and time, the type of VC, the subject of the VC and one or more identity attributes of the VC subject. The VC also includes 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 TNs from a TNSP.

3.2 Acronyms & Abbreviations

3GPP	3rd Generation Partnership Project
ATIS	Alliance for Telecommunications Industry Solutions
BPO	Business Process Outsource
CSCF	Call Session Control Function
CVT	Call Validation Treatment
DID	Decentralized Identifier
DLT	Distributed Ledger Technology
EIDLN	Enterprise Identity Distributed Ledger Network
HTTPS	Hypertext Transfer Protocol Secure
IETF	Internet Engineering Task Force
IP	Internet Protocol
JSON	JavaScript Object Notation
JWS	JSON Web Signature
KYC	Know Your Customer
LNPA	Local Number Portability
NANPA	North American Numbering Plan Administrator
NNI	Network-to-Network Interface
OOB	Out of Band
OSP	Originating Service Provider
PASSporT	Personal Assertion Token

PBX	Private Branch Exchange
PKI	Public Key Infrastructure
RCD	Rich Call Data
RespOrg	Responsible Organization
SHAKEN	Signature-based Handling of Asserted information using toKENs
SIP	Session Initiation Protocol
SKS	Secure Key Store
SP	Service Provider
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
TA	Trust Authority
TFNA	Toll-Free Number Administrator
TN	Telephone Number
TNSP	TN Service Provider
TNR	TN Reseller
TSP	Terminating Service Provide
URI	Uniform Resource Identifier
UUID	Universally Unique Identifier
VC	Verifiable Credential
VoIP	Voice over Internet Protocol

4 Overview

This document presents a method for participants in the VoIP service provider network ecosystem — including TNSPs, TNRs, OSPs, enterprises and KYC vetters — to record identity information, KYC vetting status and TN assignment and delegation information on a distributed ledger. This service is described as the Enterprise Identity Distributed Ledger Network (EIDLN), which performs two key functional services:

- **Enterprise Identity and Asserted KYC Vetting:** A participating organization records its digital identity, called the Enterprise Identity, to be verified and authorized by a KYC vetter as described in clause 5 of this document. KYC vetting proves to all participants of the EIDLN, that the organization has been KYC vetted in accordance with the EIDL-GA policy.

- **TN Authorizations Right to Use:** Signed TN assignment and delegation information by an authoritative participant is recorded on the EIDLN, proving that an Enterprise Identity has the right to use a TN, which can be used to attest calls being placed. As described in clause 6 of this document, the TN authorization function is underpinned by a vetted Enterprise Identity, which allows the participant authorizing a TN to an Enterprise Identity to authenticate that the business representing the Enterprise Identity is legitimate.

Note: The EIDLN *Enterprise Identity and Asserted KYC* vetting service can be used to provide proof of enterprise organization identity and KYC vetting status to other business processes of the VoIP service provider ecosystem. This could be another TN authorization solution that is supporting SHAKEN attestation, as illustrated in figure 4.1 through the EIDLN APIs as described in Annex A of this document.

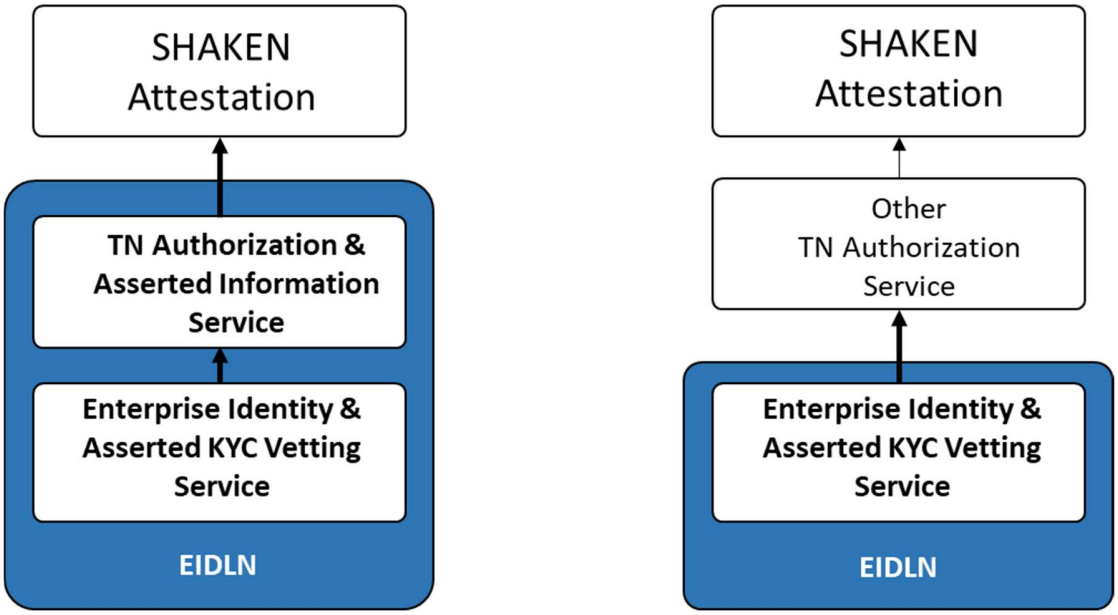

Figure 4.1 EIDLN Services Supporting SHAKEN Attestation

Enterprise Identity Management and KYC Vetting Service

A participating organization's Enterprise Identify and its proof of the KYC vetting authorization may be read from an instance of the ledger to determine authentication of the organization during customer onboarding, TN authorization and the SHAKEN attestation level to be assigned by the OSP for a call from an originating entity whose information is recorded on the ledger.

The method will illustrate how a participating organization will create and register an Enterprise Identity on the EIDLN. It also describes how the proof of KYC vetting of an organization's identity is registered and signed by an authorized vetter on the EIDLN. Allowing any participating organization to verify an organization's Enterprise Identify, the vetting status and proof it was performed by which vetting organization from the EIDLN.

The authority for KYC vetting of a participating organization's identity on the EIDLN depends on the organization's role (see clause 5 of this document for vetting trust hierarchy). Some organization roles on the EIDLN require the EIDLN TA to KYC verify and authorize their identity. These organizations' roles are TNSP's and KYC vetters. Other organization roles on the EIDLN have their vetting performed by the KYC vetter role of the EIDLN. The organizations vetted by the KYC vetter are TNR, brand enterprise and BPO enterprise.

If an organization's vetting status changes to unauthorized or revoked for any reason, as covered in clause 6.3, this is recorded on the EIDLN and is available to be read by all participants.

TN Authorizations Right to Use Service

The TN assignment and delegation information may be read from a ledger instance to determine the SHAKEN attestation level for a call from an originating entity whose information is recorded on the ledger. This 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. OSPs may use the references in the PASSporT 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. The TA will ensure all TNSP's stay in good standing and will have the ability to revoke any TNSP's credentials.

2. A TNSP assigns TNs to a TNR or a brand enterprise directly for the brand's own general calling purposes.

3. The TNR delegates TNs to a brand enterprise for the brand's own general calling purposes.

4. The Brand Enterprise may further delegate TNs to a BPO enterprise to conduct an outbound calling campaign on the brand's behalf.

5. The calling brand or BPO initiates calls using one or more OSPs. An OSP will authenticate the contents of a PASSporT that the calling brand/BPO populates using information from the EIDLN. The brand/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.

6. Based on the verification of the brand/BPO's PASSporT identity and TN right to use authorization information read from the EIDLN, an OSP determines that the calling brand/BPO's calls marked with the calling TN are entitled to Full Attestation marking designation in an outgoing SHAKEN PASSporT.

Enterprise Identity Distributed Ledger Network

Using DLT, the EIDLN provides control and scale for establishing an Enterprise Identity digital trust foundation or an identity trust fabric (ITF) to all participating organizations. As Figure 4.2 illustrates, the EIDLN enables any participating OSP in the call path to fully attest to a calling enterprise's SIP call with an authorized EIDLN identity using an authorized TN assignment or delegation. A full description of the EIDLN is explained in ATIS-I-0000076, *Enterprise Identity on Distributed Ledger for Authenticated Caller Use Cases.*

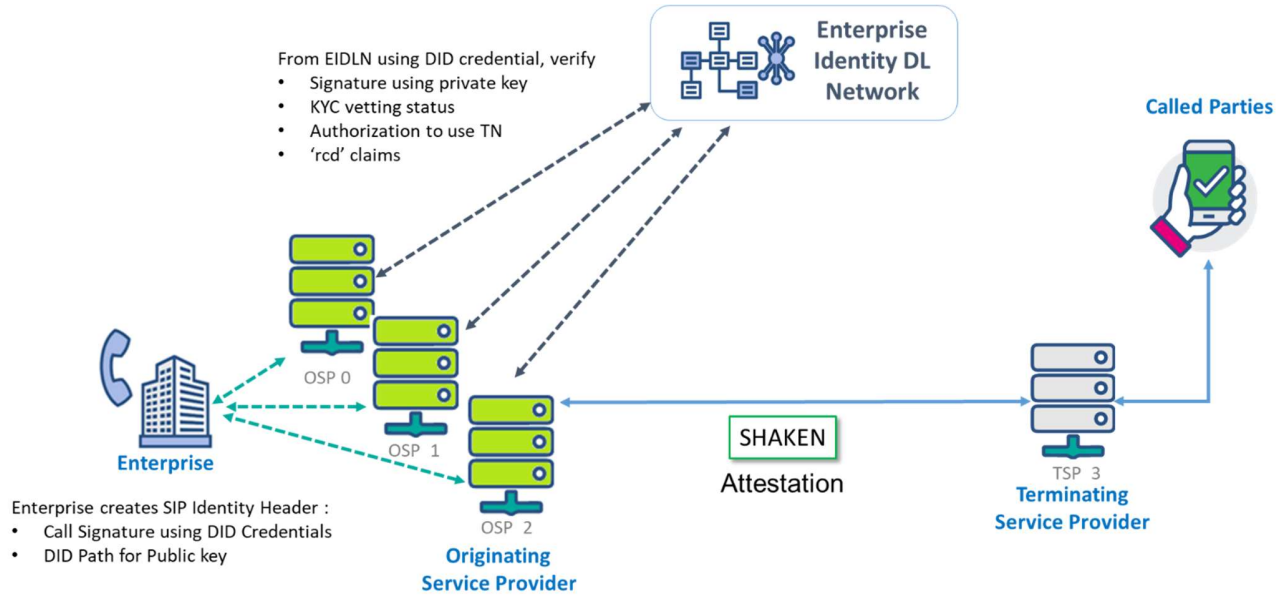

Figure 4.2 EIDLN to Attest a TN

The diagram in Figure 4.3 lists all organization actors of the EIDLN. These actors may have roles in the EIDLN's enterprise authentication and calling TN authorization process. For a full description of the EIDLN actor roles and functions, refer to ATIS-I-0000076, *Enterprise Identity on Distributed Ledger for Authenticated Caller Use Cases*.

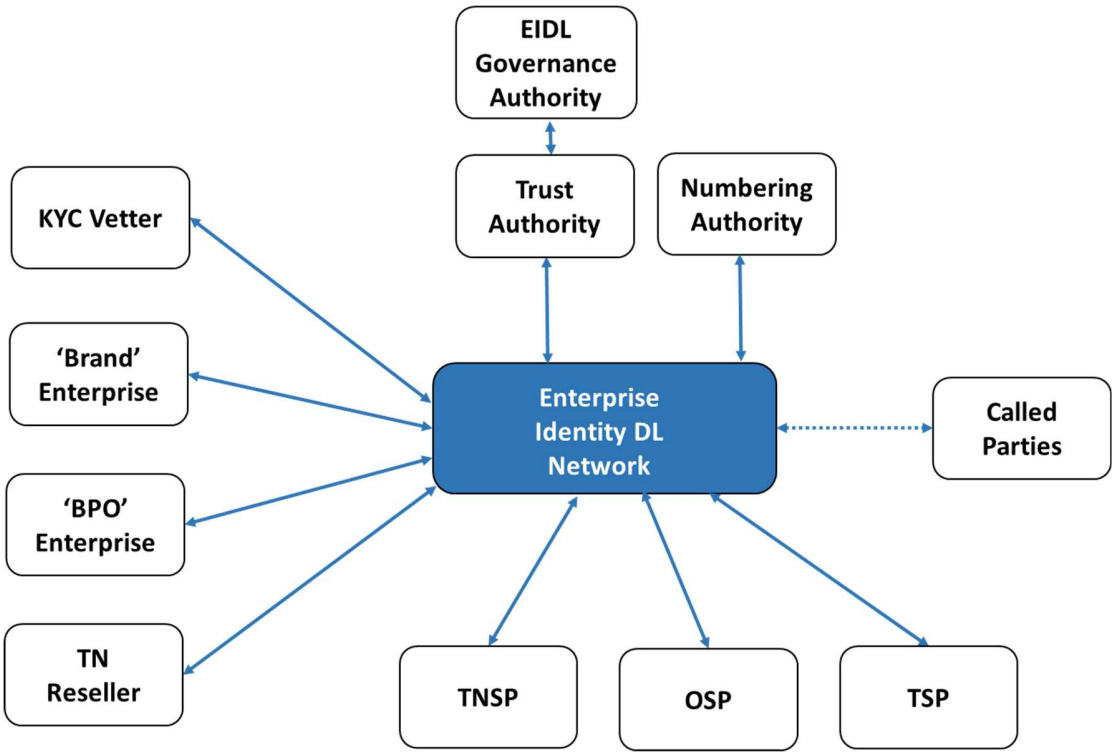

Figure 4.3 EIDLN Organization Actors

In addition to enterprise callers, all EIDLN organization actors must have an Enterprise Identity to transact on the network. An organization's Enterprise identity on the EIDLN is implemented using DIDs according to the [W3C Decentralized Identifiers].

The DID credential is generated with a public/private key pair by the organization actor. The private key is securely stored within the organization actor's secure key store (SKS). The public key is stored within the DID on the EIDLN.

The benefit of using DID for identities is that it enables the underlying DLT to be agnostic and allows interoperability across other identity networks as a federated identity service to support cross-border identity and TN authentication.

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

With a KYC vetted and approved DID, a TN can be assigned by a TNSP or delegated by a TNR to a brand to place calls or delegated by a brand to a KYC vetted and approved BPO to place calls on its behalf.

All TN allocations (assignments or delegation) on the EIDLN are represented and contained within VC claims according to [W3C Verifiable Credentials]. These VC claims are digitally signed assignments or TN delegation from the authorized TNSP to a TNR, then to a brand enterprise and finally to one or more BPOs (e.g., call centers). Each TN is encapsulated in a single VC. As the TN is assigned from the TNSP to TNR, TNR to brand and brand to BPO, each organization assignment or delegation is cryptographically signed within the VC to prove that claim. A single TN DID provides the full chain of custody for the TN by a TNSP, who assigned it, at what time and by whom. As these VC claims are on the distributed ledger, they are all true at any point in time as an authoritative source for all stakeholders against which they can verify.

Note: Throughout this document, when the term TNSP is used, it also refers to a RespOrg.

Using its DID, a KYC vetted enterprise will sign originating phone calls using the SIP identity header. This signature enables any OSP connected to the EIDLN to verify the calling enterprise DID and prove that the call is being placed by a trusted enterprise. Using this proof of identity, the OSP can then verify that the originating TN is being used by an enterprise with the authoritative right to use it. The OSP conducts that verification by checking the TN's VC on the EIDLN.. With this proof of identity and the number being used, the OSP can attest to the call using SHAKEN A-level attestation.

An Enterprise Identity implemented with DLT, using DID and VCs, will significantly enhance the capability for OSPs to validate and authenticate the calling party identity and the authority to place a call with an authorized TN. OSPs will be able to provide SHAKEN A-level attestation for enterprise calls in complex call scenarios described above.

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

4.1 Enterprise Identity Distributed Ledger Network Interworking with SHAKEN Architecture

Figure 4.4 illustrates the Enterprise Identity Distributed Ledger Network architecture interworking with the SHAKEN architecture to provide Enterprise Identity services to OSPs.

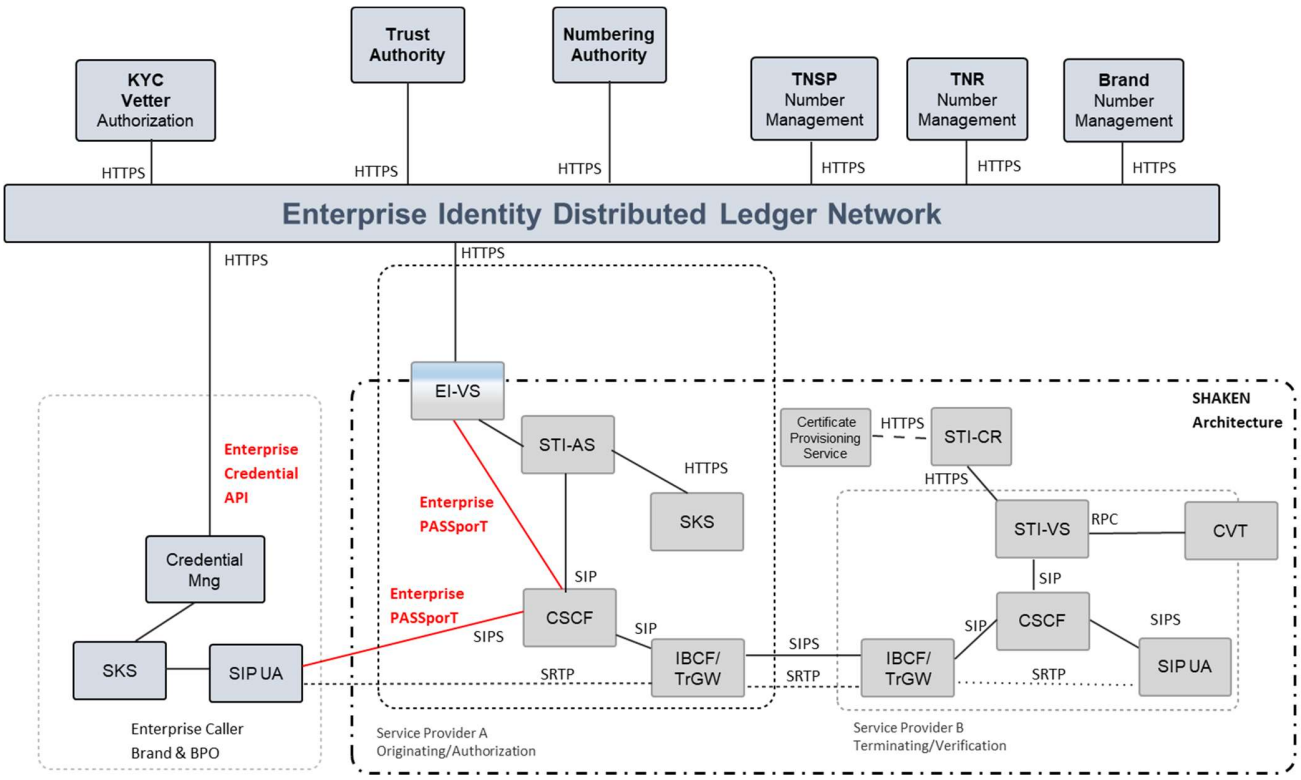

Figure 4.4 EIDLN Interworking with SHAKEN Architecture

4.2 Overview of Self-Sovereign Identity (SSI)

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

An EIDLN Actor SSI can create an account on the EIDLN to generate a DID credential with a public-private key pair. The DID and public key are published to the EIDLN for authentication when transacting with other actors on the EIDLN.

4.3 Overview of Decentralized Identifier

The concept of a globally unique DID 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 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 VCs cryptographically and immutably on the blockchain. The ITF is where supply-chain partners can verify the authenticity of an identity and related VCs. 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 DIDs and authenticated using proofs such as digital signatures and privacy-preserving biometric protocols. 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 ordinary 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 and centralized certificate authorities for key management, which is the standard in hierarchical PKI. Each entity can serve as its own root authority in cases where the DID registry is a distributed ledger.

DID Path format

DID format is similar to the URN format. The key difference is that we are specifying a method instead of a namespace in the DID format. The method refers to *W3C, Decentralized Identifiers (DIDs) v1.0*, which is the specification used to register and manage DIDs on a particular blockchain or DLT. Figure 4.5 compares the two formats.

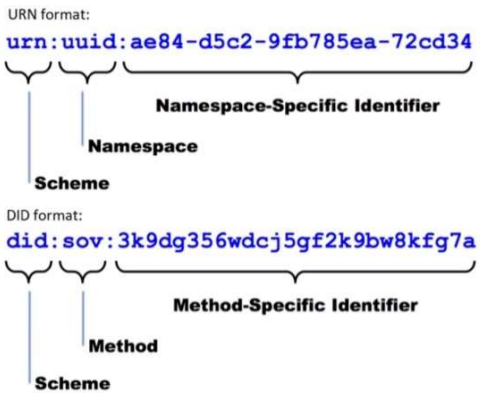

Figure 4.5 DID Format - Example Sovrin Network Method) Compared to URN Format

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-Linked Data (LD) file. 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 a TN of the DID's owner.
4. The DID owner 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 DID document's integrity.

The format of an enterprise identity DID document is:

```
{
"@context": ["https://www.w3.org/ns/did/v1", "https://w3id.org/security/v1"],
       "id": "did:example:<BRAND1>",
       " verificationMethod ": {
        "id": "did:example:<BRAND1>",
       "type": "Ed256VerificationKey2018",
       "controller": "did:example:<BRAND1>,",
        "publicKeyJwk": {
                  "kty":"EC",
                  "crv":"P-256",
                  "x":"f83OJ3D2xF1Bg8vub9tLe1gHMzV76e8Tus9uPHvRVEU",
                   "y":"x_FEzRu9m36HLN_tue659LNpXW6pCyStikYjKIWI5a0"}
}}
```

Note: In a self-sovereign identity, the controller DID is the subject of the DID.

4.4 Overview of Verifiable Credentials

A Verifiable Credential (VC) 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 the quality/qualities and property/properties of an entity that establish its existence and uniqueness. The use cases outlined here are provided 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 VCs: 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 VCs. A holder is usually, but not always, the subject of the VCs that it is holding. Holders store their credentials in credential repositories.

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

5 Enterprise Identity Management on Distributed Ledger

This clause describes the specific implementation of the W3C DIDs and VCs on the EIDLN to provide organization actor identification and proof of KYC vetting status for participants of the EIDLN and caller TN attestation for the OSP. For more details about the Enterprise identity use cases, architecture, actors and functional flows, see ATIS-I-0000076, *Enterprise Identity on Distributed Ledger for Authenticated Caller Use Cases*.

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

Each EIDLN actor must generate a DID together with a public/private key pair and publish their DID document with its public key onto the EILDN. EIDLN. Figure 5.1 illustrates this process.

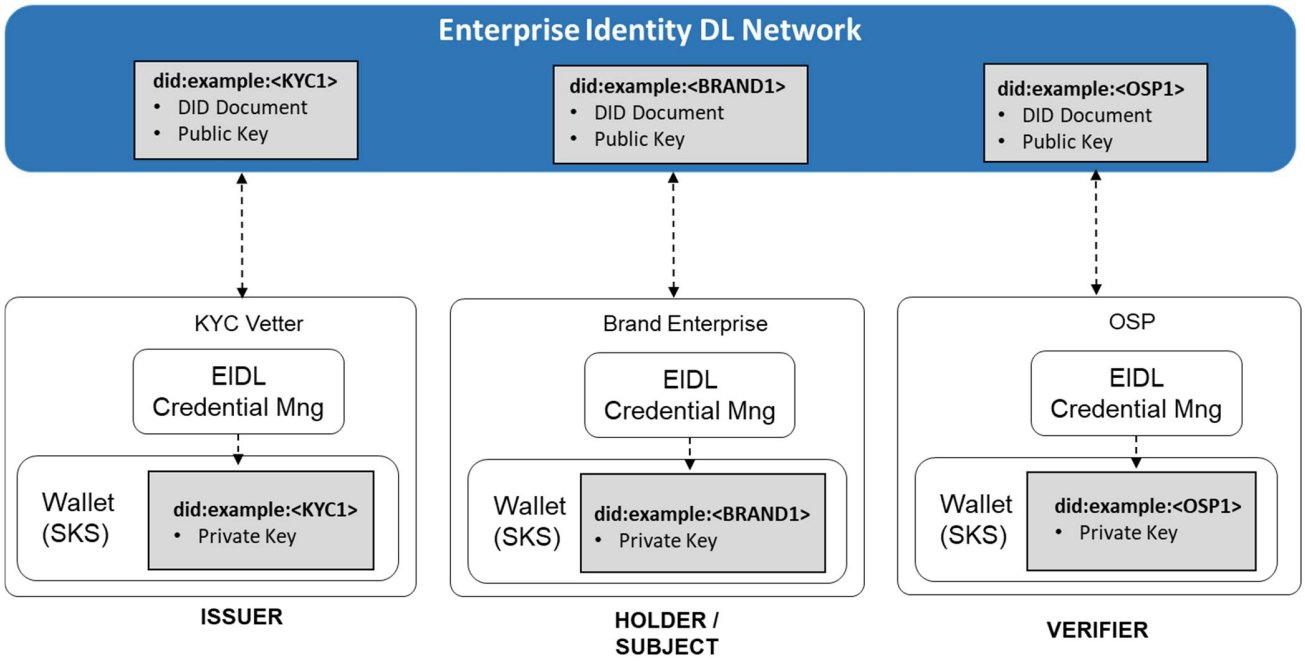

Figure 5.1 Creation of DID on EIDLN

5.1.1 Identity Creation API

Using the Enterprise Identity API definition described in Annex A of this document, an organization's client-side credential manager will integrate with the EIDLN to generate tits DID credentials and a corresponding key pair. Figure 5.2 is an example of the organization's client-side API generating a key pair and registering with the EIDLN.

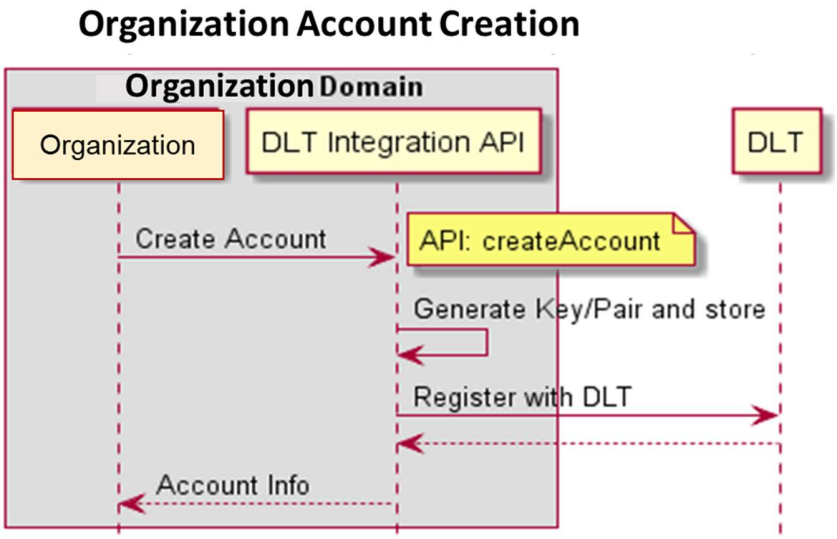

Figure 5.2 API Flow Creation of DID on EIDLN

5.2 Identity Vetting Trust Authority Hierarchy

Every actor of the EIDLN must have its DID authorized to participate in the Enterprise Identity service. Using the organization DID, the requesting organization actor will apply to the appropriate vetting authority to have its DID authorized to participate on the EIDLN.

The TA's role is to verify the organizations that represent the roles of TNSP and KYC vetter actors of the EIDLN. The TA vetting authorization process is according to the policy defined by the EIDLN Governance Authority. When the TA has authorized an organization, the TA will create a VC claim on the EIDLN for that organization's DID to indicate that it is authorized.

The KYC vetter's role is to verify the organizations that represent the roles of TNR, brand enterprise and BPO enterprise actors on the EILDN. The KYC vetting authorization process is according to the policy defined by the EIDLN Governance Authority. When the KYC vetter has authorized an organization, the KYC will create a VC claim on the EIDLN for that organization's DID to indicate that it is authorized.

The TNSP assigns TNs that are under its control to TNR and brand organizations that have been KYC vetted and authorized on the EIDLN. A TNSP's right to assign TNs must comply with the policy of the appropriate numbering authority for the given TN. When a TNSP assigns a TN to a TNR or brand organization, the TNSP will create a VC claim on the EIDLN for that organization's DID to indicate that it is authorized to use that TN.

Figure 5.3 summarizes the vetting trust hierarchy of the EIDLN.

Note: a KYC vetter must be approved by the TA before it can authorize other organizations.

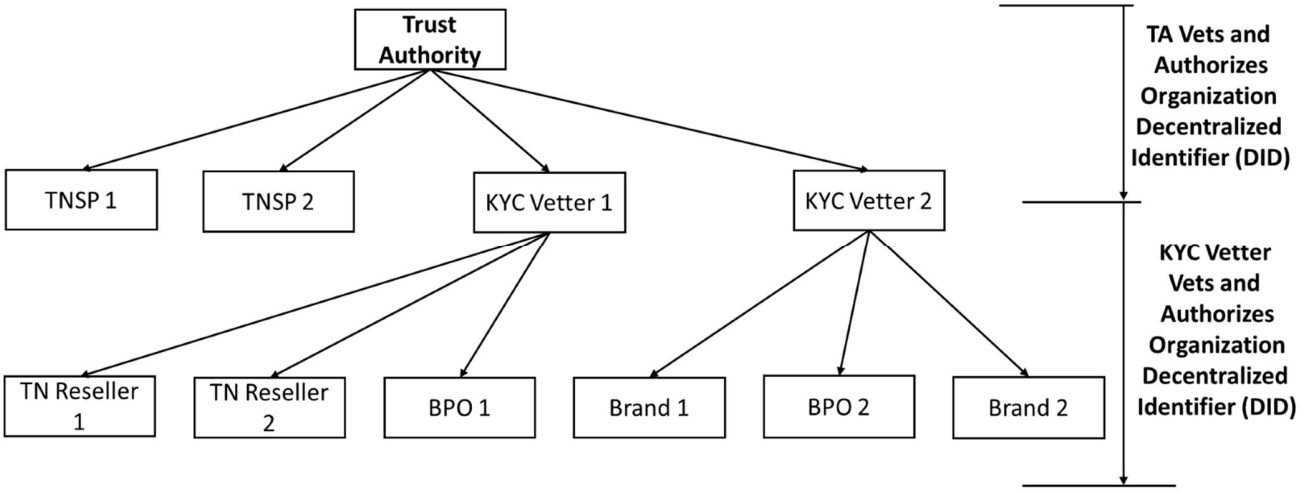

Figure 5.3 EIDLN Trust Hierarchy

The issuer (TA or KYC vetter) of the verifiable credential claim for the organization will contain data attributes that indicate the organization's business function:

- 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 Authorized, Revoked
- Vetter Name = ""Vetter Business Name"" – Assigned by TA

5.3 Identity Vetting Authorization Verifiable Credential

The Enterpriser Identity vetting status by authorized vetting actors on the EIDLN is contained within W3C VC claims. A VC is a tamper-evident credential with authorship that can be cryptographically verified as proof of the Enterprise Identity using the public key of the actors Enterprise Identity DID. When an authorized KYC vetter or TA on the EIDLN has vetted the Enterprise Identity, a VC claim is created to indicate the vetting status for that Enterprise Identity. It is added to the Enterprise Identity DID for proof.

Vetting Status Verifiable Credential

A vetting status VC contains the following claims:

Issuer: The actor enterprise identity DID that provides vetting of Enterprise Identities on the EIDLN and creates a verifiable claim of the subject's Enterprise Identity DID vetting status. This will be the Enterprise Identity DID of either the KYC vetter or TA actor.

Example of the issuer claim:

```
"issuer": "did:example:<KYC1>",
```

Subject: The subject Enterprise Identity DID that the vetting credential is for and the vetting status.

Example of the credentialSubject claim:

```
"credentialSubject": {
                        "id": "did:example:<BRAND1>,vettingstatus",
                        "organizationVettingStatus": "Authorized"
```

Vetting Authorization Process

As indicated in the trust hierarchy, the TA must authorize a TNSP, KYC vetter and OSP to perform their function on the EIDLN. Only an Authorized KYC vetter can then vet and authorize TNRs, brand enterprises and BPO enterprise actors to perform their function on the EIDLN.

1. Using an out-of-band (OOB) request (as indicated in option 1 below), the organization will request the vetter actor to perform vetting of its business for authorization to use the EIDLN services, according to the agreed policy of the EIDLN (out of scope for this document at this time).

2a. The vetting authority (TA or KYC vetter) will perform vetting of the requesting organization according to the agreed EIDLN policy. If authorized, the vetting authority will generate and sign a verifiable credential claim, indicating the organization DID and that its authorization status is now authorized. The vetter returns the status to the requesting organization and the VC claim identity to support this.

2b. Optionally, the requesting organization may have rich call data information (referred to as 'rcd') that is to be associated with its Enterprise Identity DID. The vetted authority (TA or KYC vetter) will perform vetting of 'rcd' information to be associated with the requesting organization. If authorized, the vetting authority will generate and sign a verifiable credential claim for the supporting 'rcd' information claims for the organization's DID. The vetter returns the status to the requesting organization together with the VC claim identity to support this.

In the example outlined in figure 5.4, the brand enterprise (BRAND1) has been authorized by the KYC vetter (KYC1). KYC vetter (KYC1) has generated a signed VC claim to indicate this on the EIDLN, and any other actor can verify on the EIDLN. As part of the authorization process, the KYC vetter will verify the enterprise brand name that will be used to identify them. If the enterprise passes the KYC vetting process, then the KYC vetter will generate a VC claim on the EIDLN, indicating that the DID of the enterprise is authorized, and sign the VC with its private key for proof.

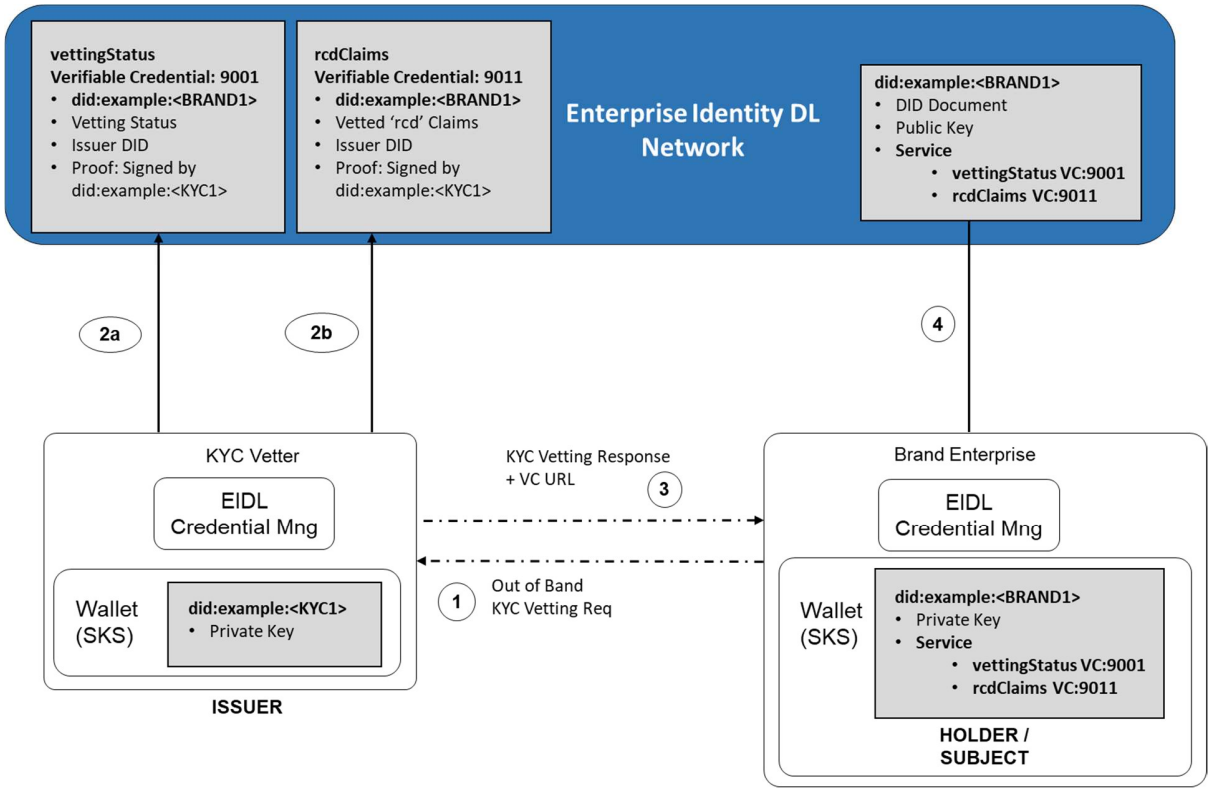

Figure 5.4 Request Org to be KYC Vetted and Create Vetted VC Claim

Organization Vetting Status VC Claim

Example of a W3C VC 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": "https://access.atis.org/credentials/9001",
       "type": ["VerifiableCredential", "VettingStatus"],
       "holder": "did:example:<BRAND1>",
       "issuer": "did:example:<KYC1>",
       "issuanceDate": "2020-01-01T19:73:24Z",
       "credentialSubject": {
              "id": "did:example:<BRAND1>, vettingstatus ",
              "organizationVettingStatus": "Authorized"
  },
  "proof": {signed by KYC vetter}
}
```

3. The KYC vetter will reply to the enterprise organization indicating the URL of the VC on the EIDLN that supports its KYC vetting approval. In this example, https://access.atis.org/credentials/9001, the issuer KYC1 has authorized the enterprise BRAND1.

4. The enterprise will update its DID document on the EIDLN with the service that indicates that it has been KYC vetted, using the URI of the VC supporting the vetting authorization.

```
{
 "@context": ["https://www.w3.org/ns/did/v1", "https://w3id.org/security/v1"],
 "id": "did:example:<BRAND1>",
 "publicKey": {
    "id": "did:example:123456789abcdefghi",
    "type": "Ed25519VerificationKey2018",
    "controller": "did:example:<BRAND1>",
    "publicKeyJwk": {
            "kty":"EC",
            "crv":"P-256",
            "x":"f83OJ3D2xF1Bg8vub9tLe1gHMzV76e8Tus9uPHvRVEU",
            "y":"x_FEzRu9m36HLN_tue659LNpXW6pCyStikYjKIWI5a0"
  }}
"service": [{
    "id":"did:example:<BRAND1>;vettingstatus",
    "type": "VerifiableCredentialService",
    "serviceEndpoint": "https://access.atis.org/credentials/9001"
  }]
}
```

5.3.1 Enterprise Identity Vetting Status

There are three possible identity vetting status options recorded in an organization's identity VC claim declared in the organization's VettingStatus claim.

Not Authorized:

- Where the organization's identity has conducted a vetting process but has not been authorized by the TA or a KYC vetter.
- Changed from authorized to not authorized by the TA or a KYC vetter, for example when vetting authorization has expired, and the TA or KYC vetter needs to perform a new vetting process on the origination identity. The EIDLN Governance Authority policy defines the valid vetting authorization period of an organization vetting status.

Authorized:

- Where the organization's identity has been vetted and is authorized by the TA or a KYC Vetter. The EIDLN Governance Authority policy defines the vetting authorization criteria of an organizations vetting status.

Revoked:

- Where the TA or a KYC vetter has revoked the organization's identity.
- The EIDLN Governance Authority policy defines the reason for revocation of an organizations vetting status.

Example of a W3C VC claim representing the revoked vetting status of an organization:

```
{
  "@context": ["https://www.w3.org/2018/credentials/v1",
       "https://access.atis.org/apps/group_public/dlt/v1/"],
       "id": "https://access.atis.org/credentials/9001",
       "type": ["VerifiableCredential", "VettingStatus"],
       "holder": "did:example:<BRAND1>",
       "issuer": "did:example:<KYC1>",
       "issuanceDate": "2020-01-01T19:73:24Z",
              "credentialSubject": {
              "id": "did:example:<BRAND1>",
              "organizationVettingStatus": "Revoked"
  },
  "proof": {signed by KYC vetter}
}
```

5.3.2 Organization 'rcd' Identifiable Information Verifiable Credential

During the vetting process, an organization will provide identifiable information about its business and function that can be used for a verified 'rcd'. Examples include the business' brand name, logo, business function and callback number. The vetted organization identifiable information can be included into a rcd VC, which is used for the verification of rcd enhanced calling party data presented in a modified rcd PASSporT described in clause 7, such as name, address, photos, logos and other information that may be extended in the future. The following VC claims are used to prove that the organization's identifiable information has been vetted.

"`organizationType`":

The vetted organization actor type on the EIDLN is TNSP, KYC vetter, TNR, brand or BPO.

"`organizationName`":

The name is the vetted organization brand name that will be used for enhanced calling party data. This claim can be assigned to a TN authorization VC as described in clause 6 to provide verification of the calling party "`nam`" key value presented in a modified rcd PASSporT covered in clause 7.

"`organizationLogo`":

The URL of the vetted organization brand identity logo will be used for enhanced calling party data. This claim can be assigned to a TN authorization VC as described in clause 6 to provide verification of the calling party "`jcd`" key value, including the URL for the logo in a modified rcd PASSporT covered in clause 7.

"`organizationJcard`":

The URL of the vetted JSONencoded jCard of the organization brand identity information will be used for enhanced calling party data. The JSON format or vCard defined in [RFC7095] is an extensible JSON object for the transport of personally identifiable types of information. This claim can be assigned to a TN authorization VC as described in clause 6 to provide verification of the calling party "`jcl`" key value presented in a modified rcd PASSporT covered in clause 7.

"`callingDataDigest`"

Contains a SHA256 digest of the "`rcd`" claims that will be used by the calling party, calculated by the vetter. The [draft-ietf-stir-passport-rcd] specifies how the claim of the "rcd" PASSporT is used to protect the integrity of the rcd from being maliciously modified. The "`rcdi`" claim contains a digest that is calculated across all of the rich call data, (i.e., the input to the digest calculation is the rcd claim contents, plus any resources referenced by the "`rcd`" claim contents, plus any resources referenced by the referenced resources and so on).

As described in clause 5.4, the requesting organization will request a KYC vetter to verify and authorize its associated rcd information to be associated to the actor's DID.

1. Using an OOB process, the organization will request the Vetter Actor to perform vetting of its business 'rcd' information for authorization to use the EIDLN services, according to the agreed policy of the EIDLN (out of scope for this document at this time).

2a. Verify the vetting status of the organization DID `VettingStatus` VC claim to ensure that the DID has been KYC vetted.

2b. The VA (TA or KYC vetter) will perform vetting of the requesting organizations rcd information according to the agreed EIDLN policy. If authorized, the vetting authority will generate and sign a VC claim, indicating the organization DID and their vetted rcd information. The vetter returns the status to the requesting organization together with the VC claim identity to support this.

Example of a W3C VC claim representing the vetted organization's rcd identification information:

```
{
  "@context": ["https://www.w3.org/2018/credentials/v1",
       "https://access.atis.org/apps/group_public/dlt/v1/"],
       "id": "https://access.atis.org/credentials/9011",
       "type": ["VerifiableCredential", "vettedrcd"],
       "holder": "did:example:<BRAND1>",
       "issuer": "did:example:<KYC1>",
       "issuanceDate": "2020-01-01T19:73:24Z",
       "credentialSubject": {
              "id": "did:example:<BRAND1>;rcdclaims>",
               "organizationType": "Brand",
              "organizationName": "Big Bank",
              "organizationLogo": "https://vetter.org/bigbankLogo.jpg",
              "organizationJcard": "https://vetter.org/bigbank.json",
              "callingDataDigest": "A9……"
       },
  "proof": { signed by KYC vetter }
}
```

3. The KYC vetter will reply to enterprise organization indicating the URL of the VC on the EIDLN that supports its rcd vetting approval. In this example, `https://access.atis.org/credentials/9011`, the issuer, KYC1, has authorized the enterprise BRAND1 rcd information.

4. The organization will update its DID document on the EIDLN with the service that indicates that it has vetted rcd information, using the URI of the VC supporting the VA.

```
{
 "@context": ["https://www.w3.org/ns/did/v1", "https://w3id.org/security/v1"],
 "id": "did:example:<BRAND1>",
 "publicKey": {
    "id": "did:example:<BRAND1>",
    "type": "Ed25519VerificationKey2018",
    "controller": "did:example:<BRAND1>",
    "publicKeyJwk": {
            "kty":"EC",
            "crv":"P-256",
            "x":"f83OJ3D2xF1Bg8vub9tLe1gHMzV76e8Tus9uPHvRVEU",
            "y":"x_FEzRu9m36HLN_tue659LNpXW6pCyStikYjKIWI5a0"
             }}
"service": [{
       "id":"did:example:<BRAND1>;vettingstatus",
       "type": "VerifiableCredentialService",
       "serviceEndpoint": "https://access.atis.org/credentials/9001"
  }, {
       "id":"did:example:<BRAND1>;rcdclaims",
       "type": "VerifiableCredentialService",
       "serviceEndpoint": "https://access.atis.org/credentials/9011"
 }]
}
```

5.3.3 Organization Vetting API

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

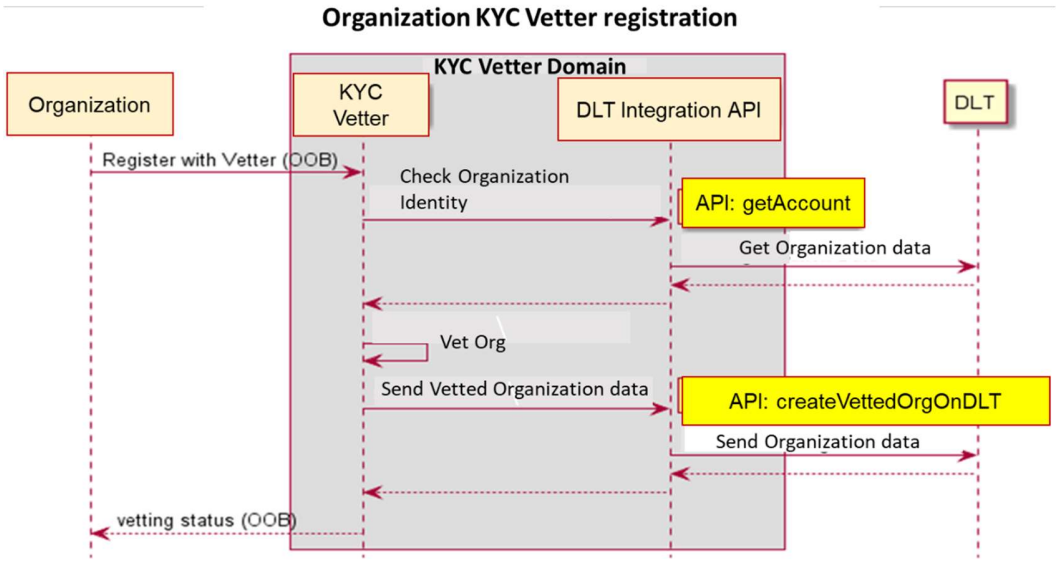

Figure 5.5 API Flow for OOB Request to be Vetted

5.3.4 Organization rcd Claims Vetting API

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

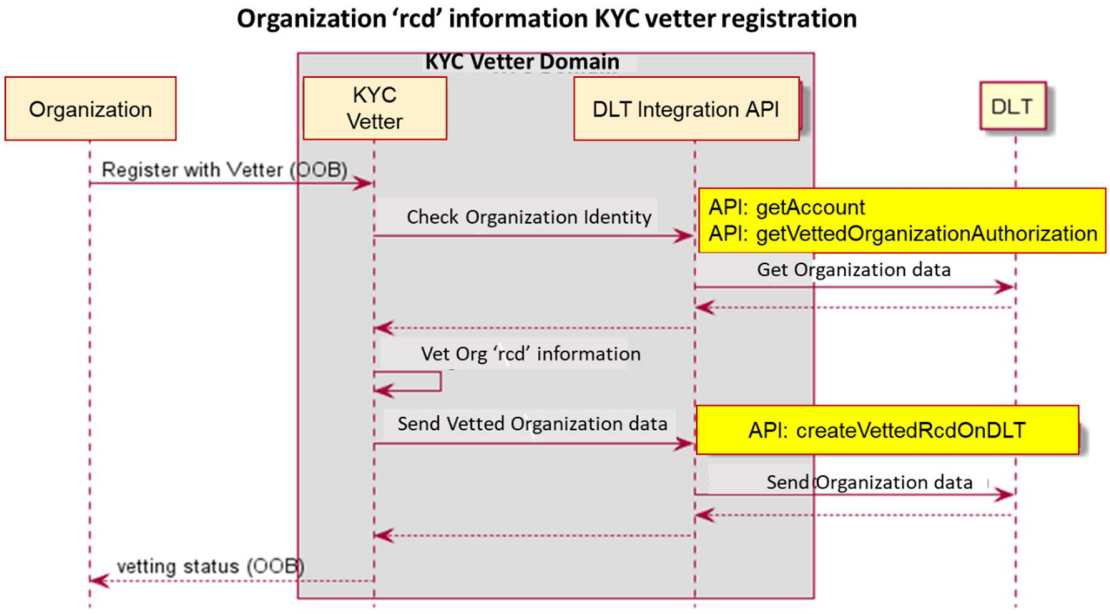

Figure 5.6 API Flow for OOB Request for rcd Claims to be Vetted

6 TN Authorizations

All TN authorizations by actors on the EIDLN are contained within W3C VC claims. A VC is a tamper-evident credential with authorship that can be a cryptographically verified proof of the TN authorization using the public key of the actor's Enterprise Identity DID. When a TN unique identity is created for the first time by a TNSP, a corresponding TN identity DID is created on the EIDLN to support the TN identity. The TN identity DID is used to manage the control of use and chain of custody for a given TN unique identity.

TN Authorization VC

A TN authorization VC contains the following claims:

Issuer: The actor enterprise identity DID that creates a claim and has the authority to assign/delegate the TN. This will be the Enterprise Identity DID of either the TNSP, TNR or brand actor.

Example of the issuer claim:

```
"issuer": "did:example:<TNSP1>",
```

Subject: The TN identity DID and information about the TN calling purpose as agreed for this TN use by the holder. The following claims are declared in the TN authorization VC:

The id specifies which DID of the TN authorization that this VC supports.

The TN authorization is verifiable subject claims that are included in the TN authorization verifiable credential:

TN	The TN authorized to be used by the holder identity.
Type	The type of TN authorization by the issuer, defined as allocation, assignment or delegation.
Purpose	Optionally, the VC can include a call purpose representing the use of the TN authorization. The call purpose is used for verification of the calling party "crn" key value presented in a modified 'rcd' PASSporT covered in clause 7.

Example of the credentialSubject claim TN authorization including call purpose:

```
"credentialSubject": {
  "id": "did:example:<TNIDENTITY>",
     "TN Authorization": {
           "TN": "12002001234",
           "Type": "Delegation",
           "Purpose": "Loan Application"
```

Holder: The actor enterprise Identity DID that is authorized to use the TN as a result of this assignment or delegation. This will be the Enterprise Identity DID of either the TNR, brand or BPO actor.

Example of the holder claim:

```
"holder":  "did:example:<BRAND1>",
```

TN Decentralized Identifier

For each unique TN identity being authenticated by the EIDLN, a unique TN identity DID exists to represent that unique TN identity. All TN authorizations for the TN identity are contained in this TN identity DID. It has the chain of custody for a TN and the history of its authorized assignment and delegation.

All TN identity DIDs for a TN identity are created on the EIDLN by TNSPs. When a TNSP is assigning a TN, the TNSP's action depends on whether there already exists a TN identity created by the TNSP for the TN. If no such TN identity exists, the TNSP will create a TN identity DID for the corresponding TN assigning it to the `"issuer"` DID, i.e., TNR or Brand. If such a TN identity exists, the TNSP will update the TN identity DID `"issuer"` DID to include the new assignment.

When removing assignment of a TN from an actor, the TNSP will update the TN identity DID `"issuer"` DID to indicate that the TN is no longer assigned. When a TN is no longer assigned by a TNSP to any actors, the TN identity DID is deleted from the EIDLN.

Note: A TN may be assigned by multiple TNSPs, with each TNSP creating its own TN identity DID for the TN. The TN identity DID, holds the public key details of the authorized actors Enterprise Identity DID~~,~~ used when signing calls for this TN.

The following clauses show how the TN identity DID is created at each stage of TN authorization.

The hierarchy for which TNs are authorized from the top-level level numbering authority to a TNSP, assigned to a TNR or brand enterprise, which can be further delegated to a Brand Enterprise and then further delegated to a BPO, is outlined in figure 6.1. An allocation, assignment or delegation can be made only to an actor with vetted status authorized.

Each TN authorization assignment or delegation from the TNSP to a TNR, brand or BPO is contained within individual TN verifiable credential claim and recorded to the EIDLN. This will prove that the authority for use of a given TN is assigned or delegated to the recipient organization identity holder**,** which is signed by the authorizing organization identity issuer that is assigning or delegating it to them.

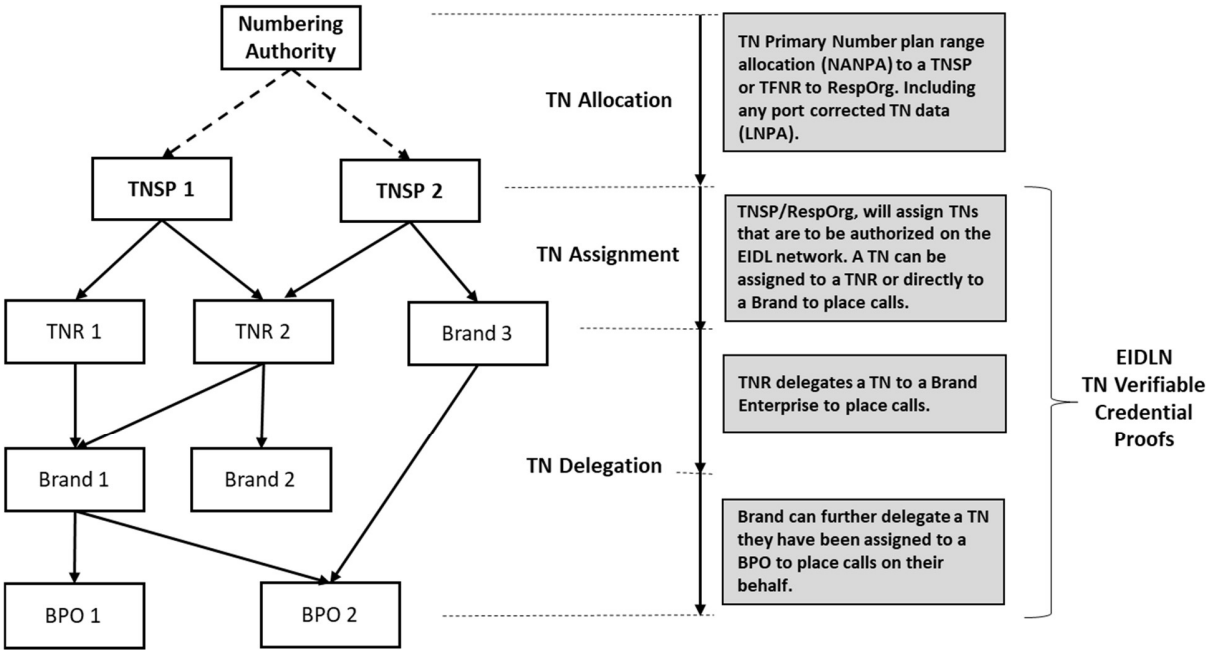

Figure 6.1 TN Authorization Hierarchy

When an authorized actor makes a TN authorization, it will generate a corresponding VC 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 TN's use. As the TN is further assigned and delegated by actors, the DID for the TN identity is appended with the authorized VC claim. The DID for a TN identity represents the current chain of authorization for the given TN. Figure 6.2 illustrates this process.

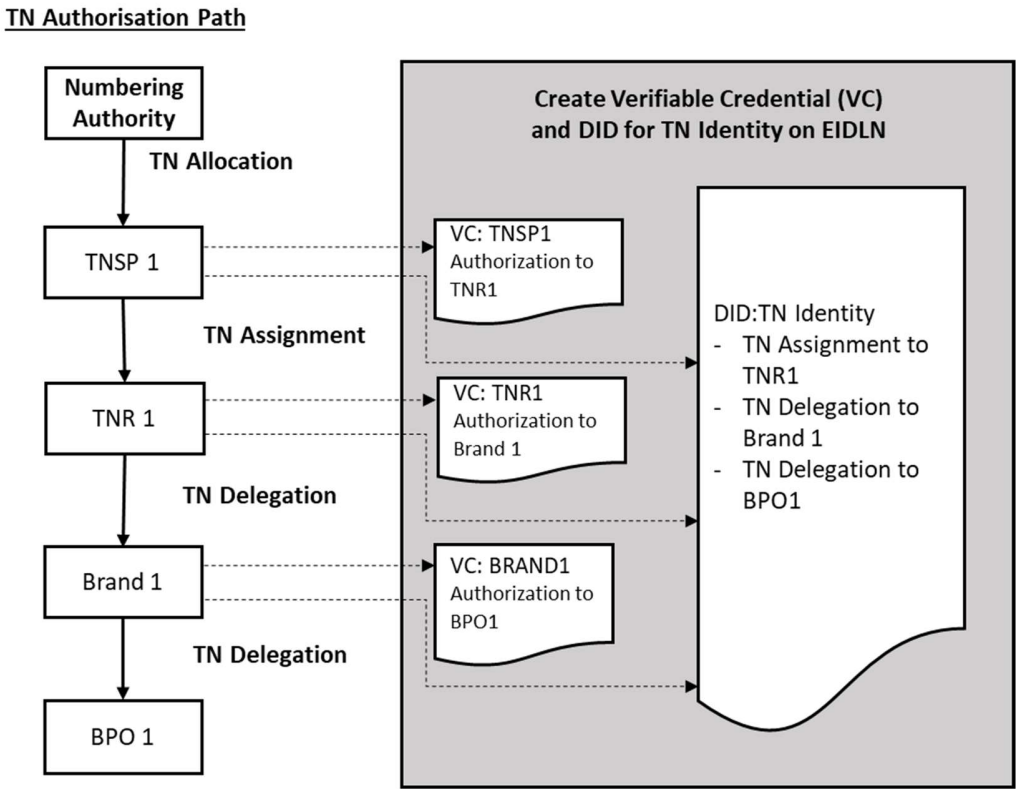

Figure 6.2 TN Authorization VC on EIDLN

Should the TN authorization be canceled or revoked, the VC 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. If 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 VC claims to support the new chain of authorization from the recipient TNSP to the assignee and delegee(s). Each authorizing actor in the chain will need to sign these verifiable claims to support the new chain of authorization. See clauses below on TN cancellation and revocation.

6.1 Assignment of a TN by TNSP to TNR or Brand

Assignment of a TN by a TNSP to a TNR or brand actor is implemented with the following steps. The following example, illustrated in figure 6.4, is for a TN assignment from a TNSP to a TNR.

1. A TN assignment request is made by the TNR to the TNSP as an OOB process, signing the request with its Enterprise Identity DID private key so that the TNSP can verify the request using the Enterprise Identity DID public key.
2. The TNSP will verify that the TNR requesting the TN assignment is currently of the vetting status "Authorized". This is done by checking the TNR Enterprise Identity DID vetting authorization VC claim before allocating the TN to the TNR.
3. If the TNR is vetting status "Authorized", the TNSP will create a VC claim for the specific TN assignment to the TNR, signed by the TNSP and recorded on the EIDLN as proof.
4. The TNSP will then create a TN identity DID with reference to the specific TN assignment VC to indicate that the TNR is now a holder of this TN and can sign calls with the corresponding Enterprise Identity DID private key.
5. If the TNR Status is not authorized/vetted, then the TN cannot be assigned to the TNR.
   The TNSP will respond to the TNR that the TN has been authorized with the URL of the VC that supports this.
6. The TNR will update its Enterprise Identity DID services to include the TN authorization VC URL on the EIDLN.

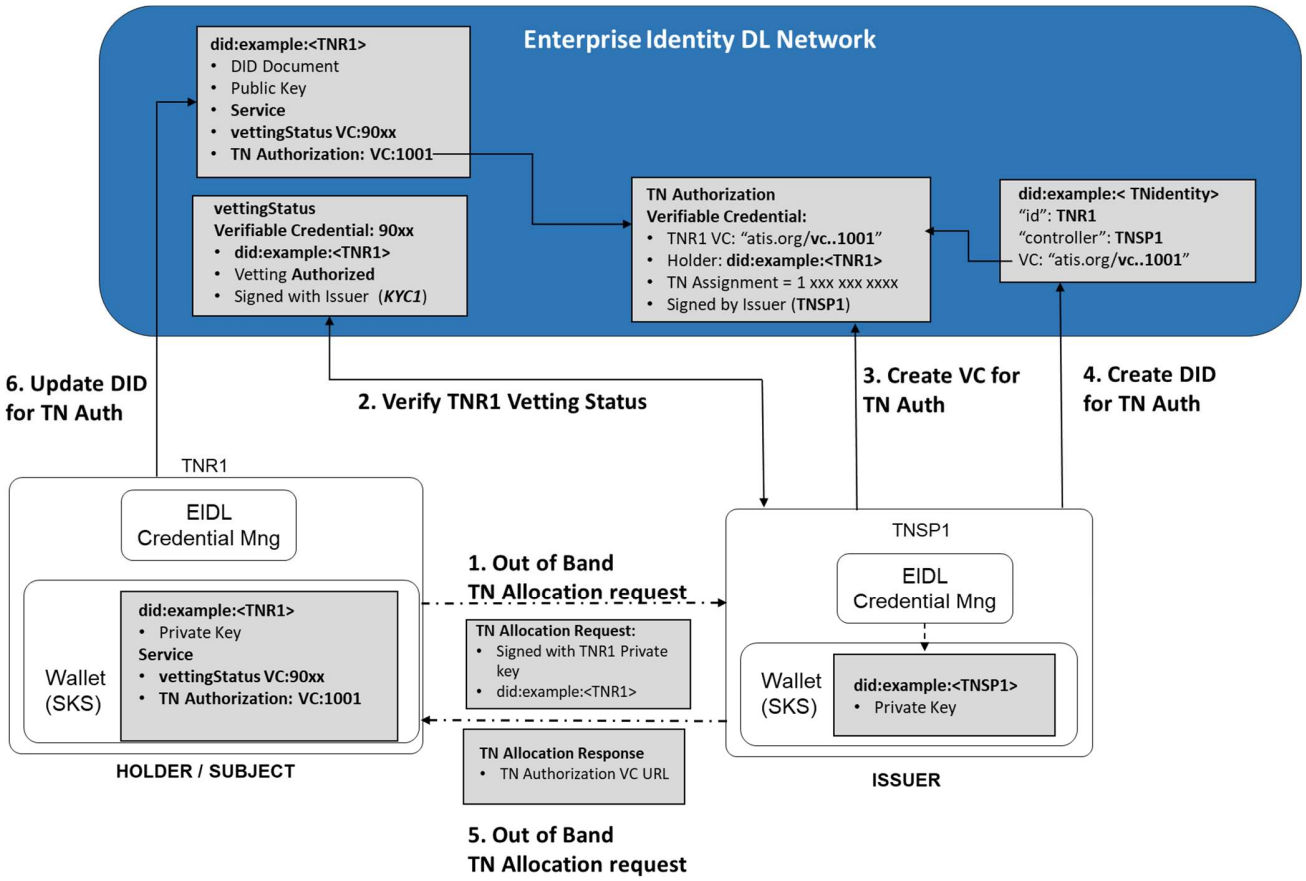

Figure 6.3 TNSP-to-TNR TN Assignment

TNSP TN Assignment VC Claim

Example of a W3C VC 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/1001",
  "type": ["VerifiableCredential", "TN Authorization"],
  "holder":  "did:example:<TNR1 >",
  "issuer": "did:example:<TNSP1>",
  "issuanceDate": "2020-01-01T19:73:24Z",
  "credentialSubject": {
    "id": "did:example:<TNIDENTITY>",
       "TN Authorization": {
             "TN": "12002001234",
              "Type": "Assignment",
             }},
"credentialStatus": {
    "id": "https:// access.atis.org /status/24,
    "type": "CredentialStatusList2017"
  },
  "proof": { signed by TNSP1 }
}
```

Adding the TN Assignment to the TN Identity DID

Example of a W3C DID Document used to prove the chain of custody for 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:example:<TNIDENTITY>",
  "authentication" :[{
       "id":"did:example:<TNR1 >",
       "controller":" did:example:<TNSP1>",
       "level":" TopRestricted",
       "publicKey": ""LMv6g……." }],
 "Service" :[{
       "id":"did:example:<TNR1;tnAuthorization>",
       "type": ""TN Authorization",
       "tn":"12002001234",
       "serviceEndpoint":"https://access.atis.org/credentials/1001"
             }],
   "proof": {...}
}
```

Adding the TN Assignment VC to the Enterprise Identity DID

Example of a W3C DID Document used to prove the chain of custody, and verify and authenticate the TN identity allocation to the TNR1 DID

```
{
"@context": ["https://www.w3.org/ns/did/v1", "https://w3id.org/security/v1"],
 "id": "did:example:<TNR1>",
  "publicKey": {
    "id": "did:example:<TNR1>",
    "type": "Ed25519VerificationKey2018",
    "controller": "did:example:<TNR1>",
    "publicKeyJwk": {
            "kty":"EC",
            "crv":"P-256",
            "x":"f83OJ3D2xF1Bg8vub9tLe1gHMzV76e8Tus9uPHvRVEU",
            "y":"x_FEzRu9m36HLN_tue659LNpXW6pCyStikYjKIWI5a0"
  }}
"service": [{
       "id":"did:example:<TNR1>;vettingstatus",
       "type": "VerifiableCredentialService",
       "serviceEndpoint": "https://access.atis.org/credentials/90xx"
  }, {
       "id":"did:example:<TNIDENTITY>;tnAuthorization",
       "type": "VerifiableCredentialService",
       "serviceEndpoint": "https://access.atis.org/credentials/1001"
 }]
}
```

The same process as above is used for a TN assignment from a TNSP to a brand directly, where the brand Enterprise Identity DID is assigned for the holder within the VC claim, and the brand Enterprise Identity DID public key is assigned to the TN identity DID.

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

Delegation of a TN by a TNR to a brand actor or a brand to a BPO actor is implemented with the following steps. The following example, illustrated in figure 6.5, is from a TNR delegation of a TN to a brand.

1. A TN assignment request is made by the brand to the TNR as an OOB process, signing the request with their Enterprise Identity DID private key, so that the TNR can verify the request using the Enterprise Identity DID public key.
   Note: The TNR can only assign a TN that is under its authorized control from a TNSP.
2. The TNR will verify that the brand requesting the TN assignment currently has the vetting status of "`Authorized`". This is done by checking the brand Enterprise Identity DID vetting authorization VC claim before allocating the TN to the brand.
3. If the brand is vetting status "`Authorized`", the TNR will create a VC claim for the specific TN delegation to the brand, signed by the TNR and recorded on the EIDLN as proof.
4. The TNR will create a TN identity DID update with a B reference to the specific TN Authorization VC, which indicates that the brand is now a holder of this TN and can sign calls with its corresponding Enterprise Identity private key.
   If the brand status is not authorized/vetted, then the TN cannot be assigned to the brand.
5. The TNR will respond to the brand that the TN has been authorized, with the URL of the VC that supports this.
6. The brand will update its Enterprise Identity DID services to include the TN Authorization VC URL on the EIDLN.

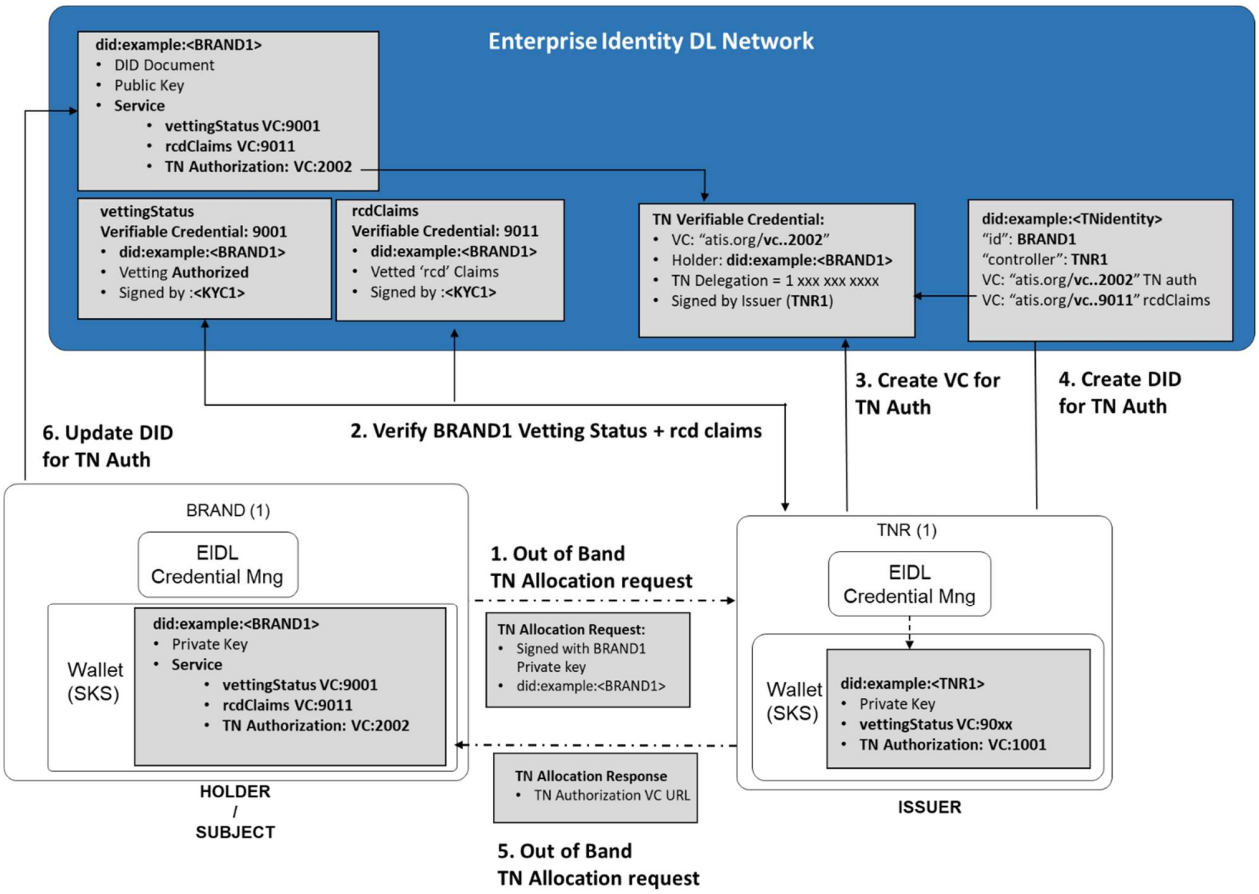

Figure 6.4 TNR-to-Brand TN Delegation – VC

TNR TN Delegation to a Brand Actor VC Claim

The following is an example of a W3C VC 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 for the intended purpose of making calls for "Loan Application". The "Purpose" claim from the VC can be verified against the 'rcd PASSporT "crn" key value claim. .

```
{
  "@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:example:<BRAND1>",
  "issuer": "did:example:<TNR1>",
  "issuanceDate": "2020-01-01T19:73:24Z",
  "credentialSubject": {
    "id": "did:example:<TNIDENTITY>",
       "TN Authorization": {
             "TN": "12002001234",
             "Type": "Delegation",
             "Purpose": "Loan Application"}},
"credentialStatus": {
    "id": "https:// access.atis.org /status/24",
    "type": "CredentialStatusList2017"},
"proof": { signed by TNR1 }
}
```

Adding the TN Delegation from the TNR to the Brand in the TN Identity DID

Below is an example of a W3C DID Document used to verify and authenticate the TN identity allocation to the TNR1 DID and verify and authenticate the TN identity allocation to the BRAND1 DID.

```
{
  "@context": ["https://www.w3.org/2018/credentials/v1",
       "https://access.atis.org/apps/group_public/dlt/v1/"],
"id": "did:example:<TNIDENTITY>",
  "authentication" :[{
       "id":"did:example:<TNR1 >",
       "controller":" did:example:<TNSP1>",
       "level":" TopRestricted",
       "publicKeyJwk": " ……."
       },{
       "id":"did:example:<BRAND1 >",
       "controller":"did:example:<TNR1>",
       "level":"TopRestricted",
       "publicKeyJkw": "……."
       }],
 "Service" :[{
       "id":"did:example:<TNR1;tnAuthoirization >",
       "type": ""TN Authorization",
       "tn":"12002001234",
       "serviceEndpoint":" https://access.atis.org/credentials/1001"
       },{
       "id":"did:example:<BRAND1;tnAuthoirization>",
       "type": "TN Authorization",
       "tn":"12002001234",
       "serviceEndpoint":"https://access.atis.org/credentials/2002"
       }],
   "proof": {...}
}
```

Adding the TN Assignment VC to the Enterprise Identity DID

The following is an example of a W3C DID Document used by BRAND1 to prove the allocation of the TN identity.

```
{
"@context": ["https://www.w3.org/ns/did/v1", "https://w3id.org/security/v1"],
 "id": "did:example:<BRAND1>",
 "publicKey": {
    "id": "did:example:<BRAND1>",
    "type": "Ed25519VerificationKey2018",
    "controller": "did:example:<BRAND1>",
    "publicKeyJwk": {
            "kty":"EC",
            "crv":"P-256",
            "x":"f83OJ3D2xF1Bg8vub9tLe1gHMzV76e8Tus9uPHvRVEU",
             "y":"x_FEzRu9m36HLN_tue659LNpXW6pCyStikYjKIWI5a0"
  }}
"service": [{
       "id":"did:example:<BRAND1>;vettingstatus",
       "type": "VerifiableCredentialService",
       "serviceEndpoint": "https://access.atis.org/credentials/9001"
       },{
       "id":"did:example:<BRAND1>;rcdclaims",
       "type": "VerifiableCredentialService",
       "serviceEndpoint": "https://access.atis.org/credentials/9011"
       },{
       "id":"did:example:<TNIDENTITY>;tnAuthorization",
       "type": "VerifiableCredentialService",
       "serviceEndpoint": "https://access.atis.org/credentials/2002"
        }]
}
```

6.2.1 Delegation of a TN by Brand to BPO

The same process as described in figure 6.5 above is used for a TN delegation from a brand to a BPO directly. As figure 6.6 illustrates, the BPO Enterprise Identity DID is assigned for the "Holder" within the VC claim, and the BPO Enterprise Identity DID public key assigned to the TN identity DID.

Specifically, when a brand delegates a TN authorization of use to a BPO, where the call is being placed by the BPO on behalf of the brand, the BPO will send enhanced call data with the signed call to indicate that the call is from the brand. The VC for the TN delegation by the brand to the BPO contains a "TermsOfUse" reference to allow the brand rcd claim information to be delegated to the BPO. This allows the rcd claims for the brand to be authenticated when the calls are placed by the BPO and signed by the BPO for this TN.

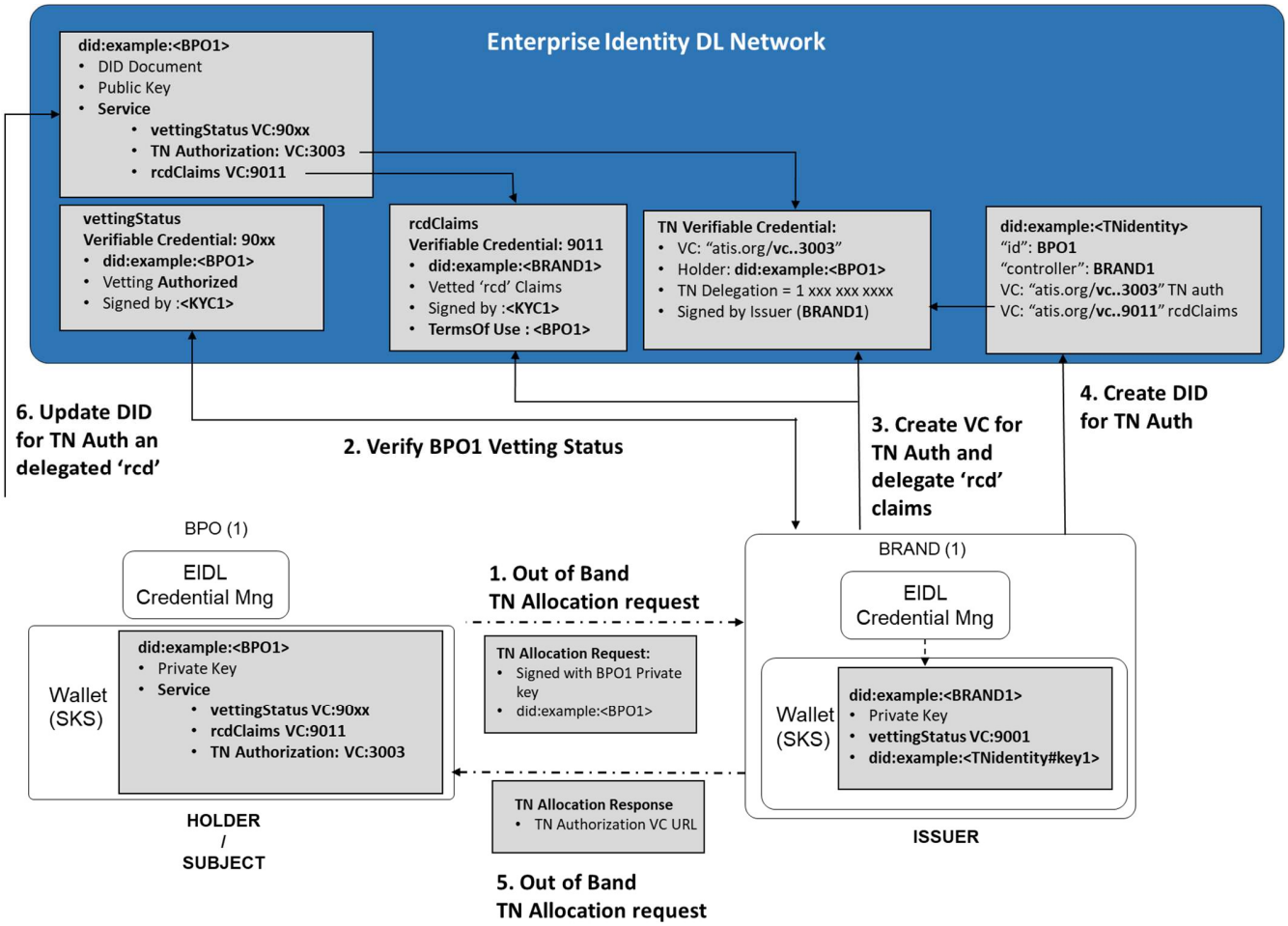

Figure 6.5 Brand-to-BPO TN Delegation

TNR TN Delegation to a Brand Actor VC Claim

The following is an example of a W3C VC claim representing the TN delegation by BRAND1 to a BPO1. The claims by BRAND1 to the BPO1 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 by BRAND1 for the intended purpose of the BPO1 making calls for "Loan Application" on behalf of BRAND1. The Purpose claims can be verified against the rcd PASSporT "crn" key value claim

```
{
  "@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:example:<BPO1>",
  "issuer": "did:example:<BRAND1>",
  "issuanceDate": "2020-01-01T19:73:24Z",
  "credentialSubject": {
    "id": "did:example:<TNIDENTITY>",
       "TN Authorization": {
             "TN": "12002001234",
             "Type": "Delegation",
             "Purpose": "Loan Application"
             }},
"credentialStatus": {
    "id": "https:// access.atis.org /status/24",
    "type": "CredentialStatusList2017"},
"proof": { signed by BRAND1 }
}
```

Adding the TN Delegation to the TN Identity DID

Below is an example of a W3C DID Document used to verify and authenticate the TN allocation to BRAND1.

```
{
  "@context": ["https://www.w3.org/2018/credentials/v1",
       "https://access.atis.org/apps/group_public/dlt/v1/"],
"id": "did:example:<TNIDENTITY>",
  "authentication" :[{
       "id":"did:example:<TNSP >",
       "controller":" did:example:<TNR1>",
       "level":" TopRestricted",
       "publicKeyJkw": "……."
       },{
       "id":"did:example:< TNR>",
       "controller":"did:example:<BRAND1>",
       "level":"TopRestricted",
       "publicKeyJkw": "……."
       },{
       "id":"did:example:<BRAND1>",
       "controller":"did:example:<BPO1>",
       "level":"TopRestricted",
       "publicKey
       }],
 "Service" :[{
       "id":"did:example:< TNR1;tnAuthoirization >",
       "type": ""TN Authorization",
       "tn":"12002001234",
       "serviceEndpoint":"https://access.atis.org/credentials/1001"
       },{
       "id":"did:example:< BRAND1;tnAuthoirization >",
       "type": "TN Authorization",
       "tn":"12002001234",
       "serviceEndpoint":"https://access.atis.org/credentials/2002"
       },{
```

```
        "id":"did:example:< BPO1;tnAuthoirization >",
        "type": "TN Authorization",
        "tn":"12002001234",
        "serviceEndpoint":"https://access.atis.org/credentials/3003"
        }],
    "proof": {...}
}
```

Adding the Delegated Use of the Brand rcd VC Information to the BPO

The brand can add a "TermsOfUse" statement in its "vettedrcd" VC that allows the rcd information that has been vetted for the brand to be delegated to the BPO. The terms of use will allow the "assigner"=BRAND1 to the "assignee"= BPO1 the use of the rcd information strictly for the TN authorization VC, as indicated in the "action" claim.

In the following example, a W3C VC is used to assign delegated terms of use for BRAND1's rcd information to the BPO1.

```
{
  "@context": ["https://www.w3.org/2018/credentials/v1",
     "https://access.atis.org/apps/group_public/dlt/v1/"],
      "id": "https://access.atis.org/credentials/9011",
      "type": ["VerifiableCredential", "vettedrcd"],
      "holder":  "did:example:<BRAND1>",
      "issuer": "did:example:<KYC1>",
      "issuanceDate": "2020-01-01T19:73:24Z",
      "credentialSubject": {
            "id": "did:example:<BRAND1>;rcdclaims>",
            "organizationType": "Brand",
      "organizationName": "Big Bank",
      "organizationLogo": "https://vetter.org/bigbankLogo.jpg",
      "organizationJcard": "https://vetter.org/bigbank.json",
      "callingDataDigest": "A9……"
      },
      "proof": { signed by KYC vetter }
}],
"termsOfUse": [{
    "type": "IssuerPolicy",
    "id": "https://access.atis.org/credentials/9033",
    "prohibition": [{
      "assigner": "did:example:<BRAND1>",
      "assignee": "did:example:<BPO1>";
      "target": "http://example.edu/credentials/9011", ( VC of 'rcd' Claims )
      "action": "https://access.atis.org/credentials/3003" ( VC of TN Delegation)
  }]},
  "proof": [ ... ]}
```

Adding the TN Assignment and Delegated rcd VCs to the BPO DID

The following is an example of a W3C DID document used to verify and authenticate the TN allocation to the brand DID.

```
{
"@context": ["https://www.w3.org/ns/did/v1", "https://w3id.org/security/v1"],
 "id": "did:example:<BPO1>",
 "publicKey": [{
    "id": "did:example:<BRAND1>",
    "type": "Ed25519VerificationKey2018",
    "controller": "did:example:<BPO1>",
    "publicKeyJwk": {
            "kty":"EC",
            "crv":"P-256",
            "x":"f83OJ3D2xF1Bg8vub9tLe1gHMzV76e8Tus9uPHvRVEU",
            "y":"x_FEzRu9m36HLN_tue659LNpXW6pCyStikYjKIWI5a0"
  }}
"service": [{
       "id":"did:example:<BPO1>;vettingstatus",
       "type": "VerifiableCredentialService",
       "serviceEndpoint": "https://access.atis.org/credentials/90xx"
       },{
       "id":"did:example:<TNIDENTITY>;tnAuthorization",
       "type": "VerifiableCredentialService",
       "serviceEndpoint": "https://access.atis.org/credentials/3003"
       },{
       "id":"did:example:<BRAND1>;rcdclaims", ( delegated by BRAND for TN )
       "type": "VerifiableCredentialService",
       "serviceEndpoint": "https://access.atis.org/credentials/9011"
       }]
}
```

6.2.2 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 to generate the VC claim to support the TN allocation/assignment/delegation on the EIDLN. The authorized TN assigner is the TNSP, TNR or brand actor.

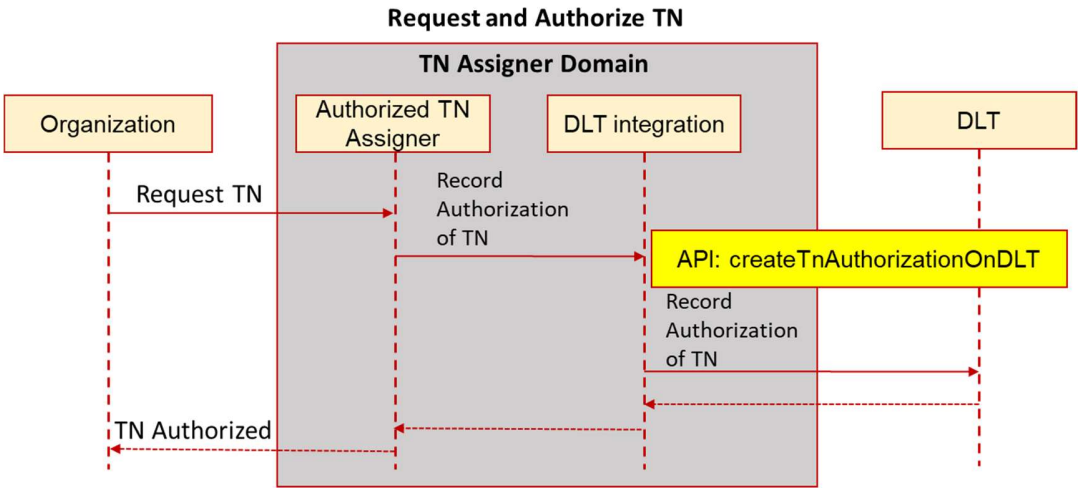

Figure 6.6 API Flow for TN Authorization

6.3 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 "currentStatus" property for the discovery of information about a VC.

The following example of the VC claim status file is for a TN being ported from a TNSP1 and a canceled number from being revoked for use 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:example:<TNSP1>",
    "issued": "2020-02-04T14:27:42Z",
    "proof": { ... }
  },{
    "claim": {
      "id": "https://access.atis.org/credentials/2002>",
      "currentStatus": "Revoked",
      "statusReason": "Canceled"
    },
    "issuer": " did:example:<TNR1>",
    "issued": "2020-02-04T14:27:42Z",
    "proof": { ... }
  }]
```

7 OSP Attestation of a call using EIDLN

This clause describes how the EIDLN 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 prove that it has the authority to place calls with the calling TN.

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, the OSP can use the information included in the enterprise PASSporT token to qualify the caller identity and authorization to place this call from the calling TN.

1) The calling enterprise will use the private key associated with its KYC-vetted Enterprise Identity DID to generate the enterprise PASSporT for the call being placed. The enterprise PASSporT header will include the URI of the callers Enterprise Identity DID on the EIDLN, and a signed claim within the enterprise PASSporT payload of the TN identity DID indicating the authorization for the calling Enterprise Identity to use the TN for placing calls.
2) The enterprise PASSporT token is included in the SIP INVITE sent to the OSP.
3) The OSP EIDLN verification service will verify the enterprise PASSporT token from the EIDLN to authenticate the calling entity enterprise identity and check the authorized use of the calling TN.
    a. Using the URI of the Enterprise Identity DID from the enterprise PASSporT header, retrieve the public key from the EIDLN to verify the token.
    b. Check that the Enterprise Identity has been KYC vetted from the vetting VC and that the current status is authorized.
    c. Check that the Enterprise Identity DID contains a VC authoring the use of the TN by this DID for the TN indicated in the SIP calling number and the "orig" claim in the enterprise PASSporT payload.
4) Based on the verification of the enterprise PASSporT for the signed call.
    a. If the enterprise PASSporT verification passes, the OSP authentication service may, based on local policy, interpret this verification result as establishing that the entity populating the PASSporT has a known authenticated identity and an association with the calling TN. Armed with this attestation criteria information, the OSP shall perform the SHAKEN authentication procedures defined in ATIS-1000074 and may assert an attestation level of full or "A" attestation. The OSP shall sign the SHAKEN PASSporT with SHAKEN certificate credentials tied to its SPC.
    b. If the enterprise PASSporT verification fails, the OSP authentication service should ignore this as input to determine the attestation level of a generated SHAKEN PASSporT and follow the standard procedures as described in ATIS-1000074 for determining attestation level based on local policy.
5) The OSP will forward the SIP INVITE including the SHAKEN PASSporT to the TNSP.
6) The TSP will verify the SHAKEN PASSporT according to the SHAKEN Specification [ATIS-1000074].

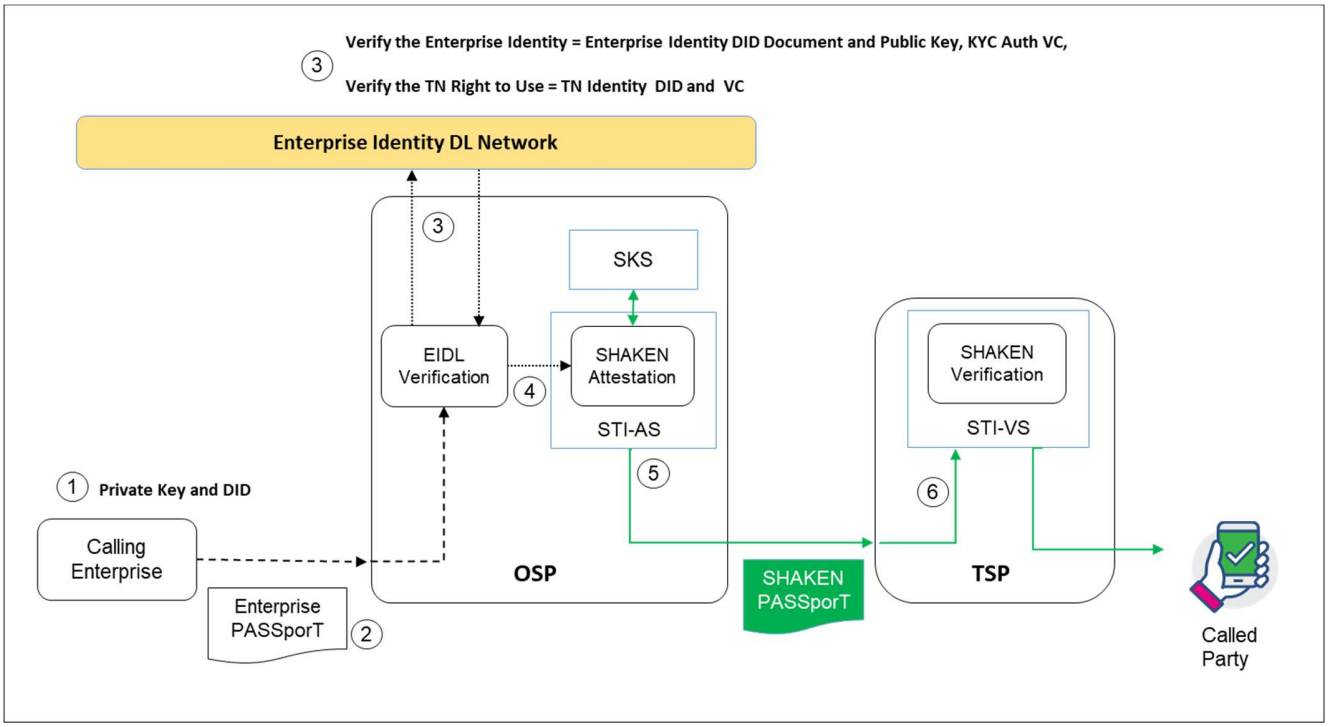

Figure 7.1 Enterprise Identity from EIDLN to Provide OSP A-Level Attestation

7.1 Enterprise PASSporT Encoding

The enterprise PASSporT token will be encoded using either a modified base PASSporT, as shown below, or by using a modified rcd PASSporT with the same modifications. The proposed modification to the base PASSporT used the reference type for the JWT authentication credentials ("jku" versus "x5u") to provide the URI of the callers enterprise identity DID on the EIDLN.

7.1.1 Use of a 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:example:098765432abcdefghi"
}
```

Payload

```
{
      "dest":{"tn":"12125551213"},
      "iat":1443208345,
      "orig":"tn":"12155551212"},
}
```

Description of the header fields and payload claims:

The "jku" key value claim indicates the URI of the DID of the of the caller identity. Use of the "jku" URI format as defined in RFC 7515 sec. 4.1.2.

Note: It is suggested that RFC 8225 be updated to include the option to 4.3 to indicate that the "jku" header parameter may reference the Enterprise Identity DID URI for the public key used to digitally sign the JWS of the PASSporT.

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

The "iat" should be set to the date and time of issuance.

7.1.2 Use of a Modified "rcd" PASSporT Encoding for Enterprise Identity

The rcd PASSporT is defined in ATIS-10000094 *Signature-based Handling of Asserted information using toKENs (SHAKEN): Calling Name and Rich Call Data Handling Procedures*. The proposed modification to the rcd PASSport used the reference type for the JWT authentication credentials ("jku" versus "x5u") to provide the URI of the enterprise identity DID on the EIDLN.

This is an example of an rcd extension PASSporT with modifications to resolve the Enterprise Identity and TN authorization, with rcd claims for enhanced calling data and call reason:

Protected Header

```
{
      "alg":"ES256",
      "typ":"passport",
      "ppt":"rcd",
      "jku":"did:example:098765432abcdefghi"
}
```

Payload

```
{
      "dest":{"tn":"12155551213"}
      "iat":1443208345,
      "orig":{"tn":"12155551212"},
      "rcd":{"nam":"Big Bank","jcd":["vcard",[["logo",{},"uri",
                "https://vetter.org/bigbankLogo.jpg"]]]}}
      "rcdi":<computed per draft-ietf-stir-passport-rcd>
      "crn":"Loan Application"
}
```

Description of the header fields and payload claims:

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

Note: It is suggested RFC 8225 be updated to include the option to 4.3 to indicate that "jku" header parameter to reference the Enterprise Identity DID URI for the public key used to digitally sign the JWS of the PASSporT.

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

The "iat" should be set to the date and time of issuance.

Note: [draft-ietf-stir-passport-rcd] defines three rcd PASSporT claims: "rcd", "crn" and "rcdi".

There are two main key values possible as part of the "jku" claim used for the enterprise PASSporT:

- The "nam" claim", which is a minimally required key value as part of the "rcd" claim value JSON object
- The "crn" claim, which contains a call reason phrase that describes the intent of the call. It is optional but recommended for enhancing usefulness to call recipients.

The "rcd" claim containing the JSON object of the enhanced calling data can contain the following key values:

- A "nam" key with a value that identifies the display name of the originating entity. If the originating entity does not have a display name, the "nam" key value shall be the empty string.

- A "jcd" key value for an "rcd" claim should be constructed with the value being equal to a jCard string. Note that additional objects are optional but may be ignored or disregarded by the receiving entity depending on the rendering capabilities of the device and/or network local policy.

- A "jcl" key value for an "rcd" claim should be constructed with the value being equal to an HTTPS URL of a file hosted on an HTTPS server containing a jCard string. Note that additional objects are optional but may be ignored or disregarded by the receiving entity depending on the rendering capabilities of the device and/or network local policy.

The "rcdi" claim protects the contents of resources referenced by the "rcd" claim from being inadvertently or maliciously modified to unauthorized values.

The "crn" claim to convey the reason for the call. The contents of the "rcd" claim have no bearing on the inclusion or value of the "crn" claim.

7.2 Enterprise PASSporT Verification Procedure

7.2.1 OSP Verification using a modified base PASSporT

When the brand/BPO originates a SIP call to an OSP (directly or indirectly), it encodes the enterprise PASSporT using the modified base PASSporT format included in the SIP identity header as defined in clause 7 of this document. The OSP decision to process the Enterprise PASSporT received in originating INVITE requests is based entirely on local policy (i.e., the OSP can apply a policy to perform these functions always, selectively based on some criteria or never).

If local policy dictates that the OSP accepts the received enterprise PASSporT, then it shall verify the enterprise PASSporT from the EIDLN as shown in figure 7.2.

1. From the "jku" claim in the header, retrieve the Enterprise Identity DID public key verify the PASSporT token.
2. Verify the Enterprise Identity DID have vetting status authorized from the vetting status VC.
3. Verify that the Enterprise Identity DID has the authorized right to use this TN from the TN Authorization VC.

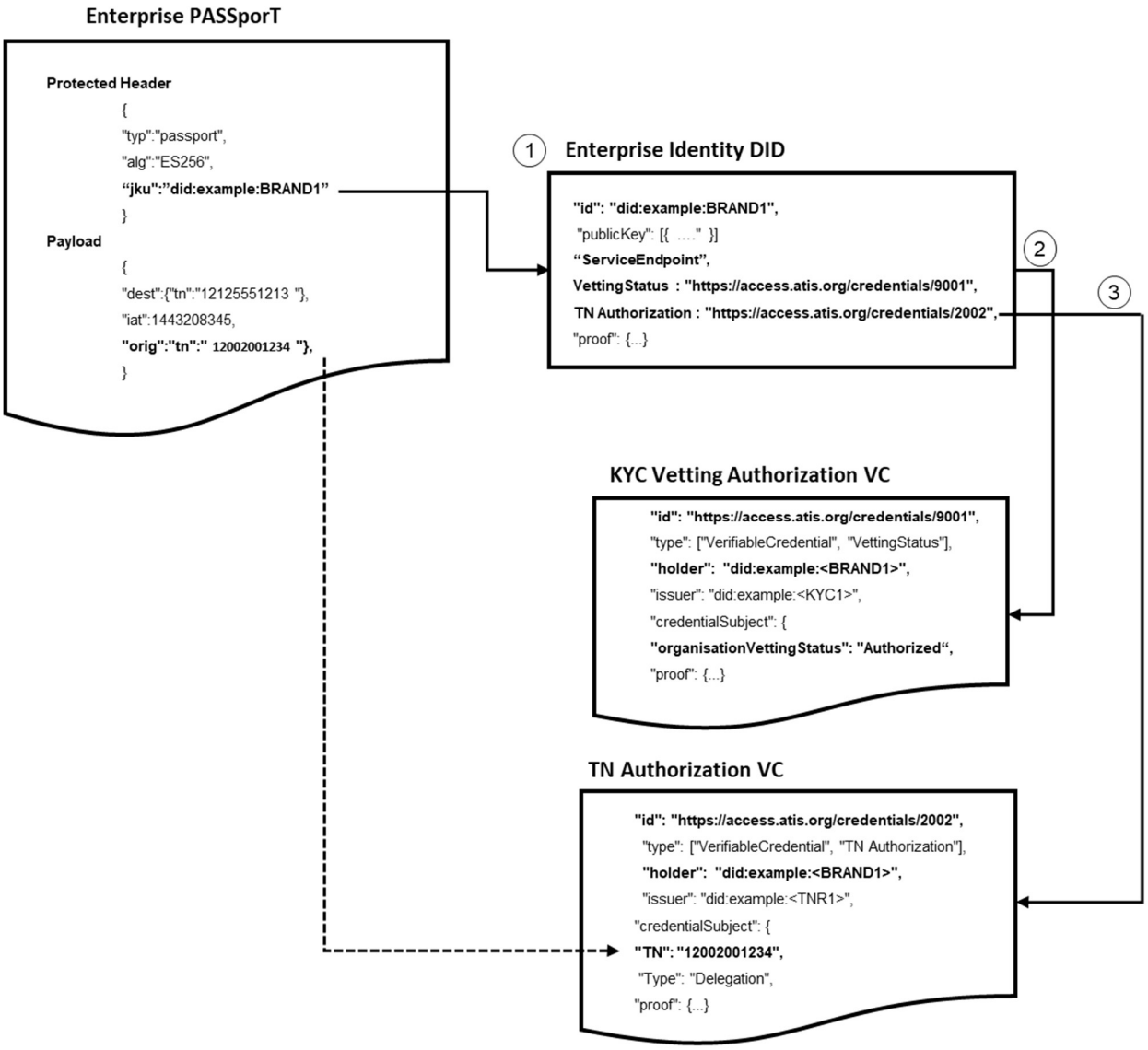

Figure 7.2 OSP Verification of an Originating Caller Using a Modified Base PASSporT

7.2.2 OSP Caller Verification API

Using the Enterprise Identity API definition described in Annex A of this document, an OSP organization can request the organization identity data to verify a modified base PASSporT, verify the organization vetting status and verify the authorization to use the TN.

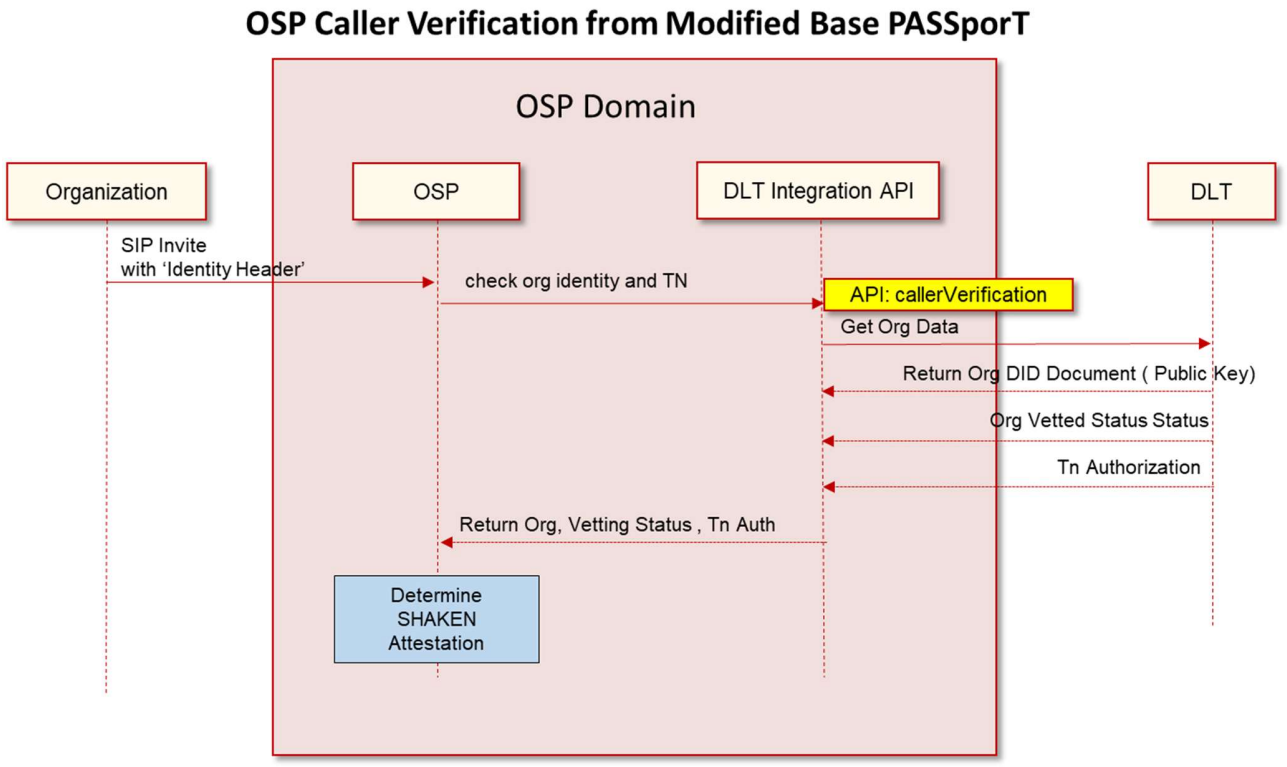

Figure 7.3 API Flow for OSP Caller Verification from Modified Base PASSporT

7.2.3 OSP Verification Using a modified “rcd” PASSporT

When the brand/BPO originates a SIP call to an OSP (directly or indirectly), it encodes the enterprise PASSporT using the modified rcd PASSporT format included in the SIP identity header as defined in clause 7 of this document. The OSP decision to process the Enterprise PASSporT received in originating INVITE requests is based entirely on local policy (i.e., the OSP can apply a policy to perform these functions always, selectively based on some criteria, or never).

If local policy dictates that the OSP accepts the received enterprise PASSporT, then it shall verify the enterprise PASSporT from the EIDLN as shown in figure 7.3.

1. From the "jku" claim in the header retrieve the Enterprise Identity DID Public Key and verify the PASSporT token.
2. Verify the Enterprise Identity DID have a vetting status authorized from the vetting status VC.
3. Verify that the Enterprise Identity DID has the authorized right to use this TN from the TN Authorization VC.
4. Where the Enterprise PASSporT is using a modified rcd PASSporT format, the enterprise PASSporT may include "rcd" claims that can be used to verify the caller’s name and purpose of the call against the corresponding TN identity VC claim used in point 4 of this claim.

The OSP can verify and compare the presented "rcd" claims in the modified rcd PASSporT with the EIDLN values from the "vettedrcd" VC claims:

- The "vettedrcd" VC claim "organizationName" = the "nam" value for rcd “rcd”
- The "vettedrcd" VC claim "organizationLogo": = the "jcl" value for rcd
- The "vettedrcd" VC claim "organizationJcard" = the "jcd" value for rcd
- The "vettedrcd" VC claim "callingDataDigest" = the "rcdi" value for rcd
- The "TN Authorization" VC claim "purpose" = the "crn" value for rcd

Based on local policy, the OSP will add these "rcd" claims to a SHAKEN PASSporT. The EIDLN verification will convey to the SHAKEN authentication service to populate the base SHAKEN claims as specified in [ATIS-1000074]. The SHAKEN authentication service shall add rcd claims to a SHAKEN PASSporT only if the criteria for A attestation are met.

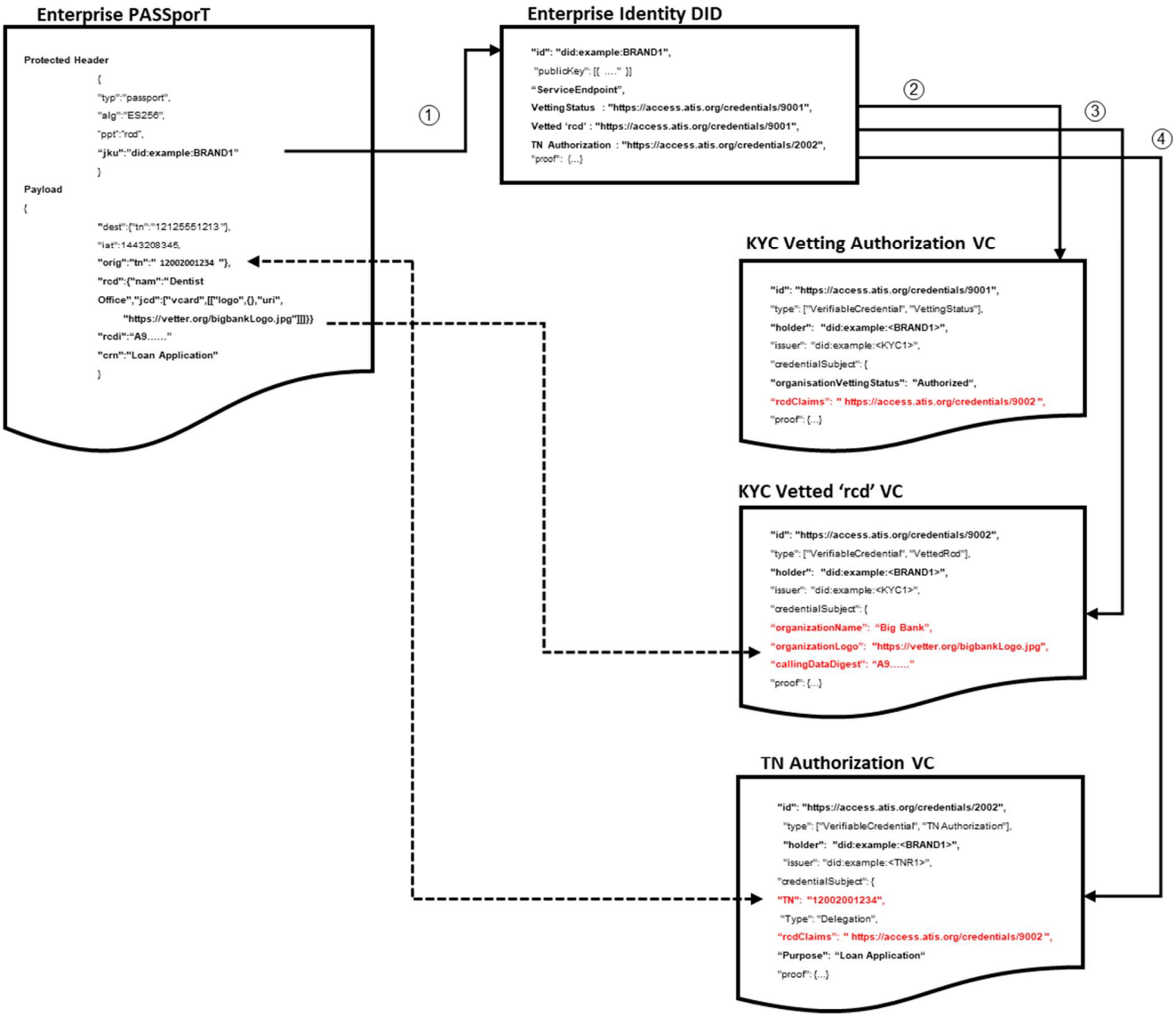

Figure 7.4 OSP Verification of an Originating Caller Using a Modified rcd PASSporT

7.2.4 OSP Caller Verification API

Using the Enterprise Identity API definition described in Annex A of this document, an OSP organization can request the organization identity data to verify a modified rcd PASSporT, verify the organization vetting status and authorization to use the TN, including verification of the rcd digest from the EIDLN.

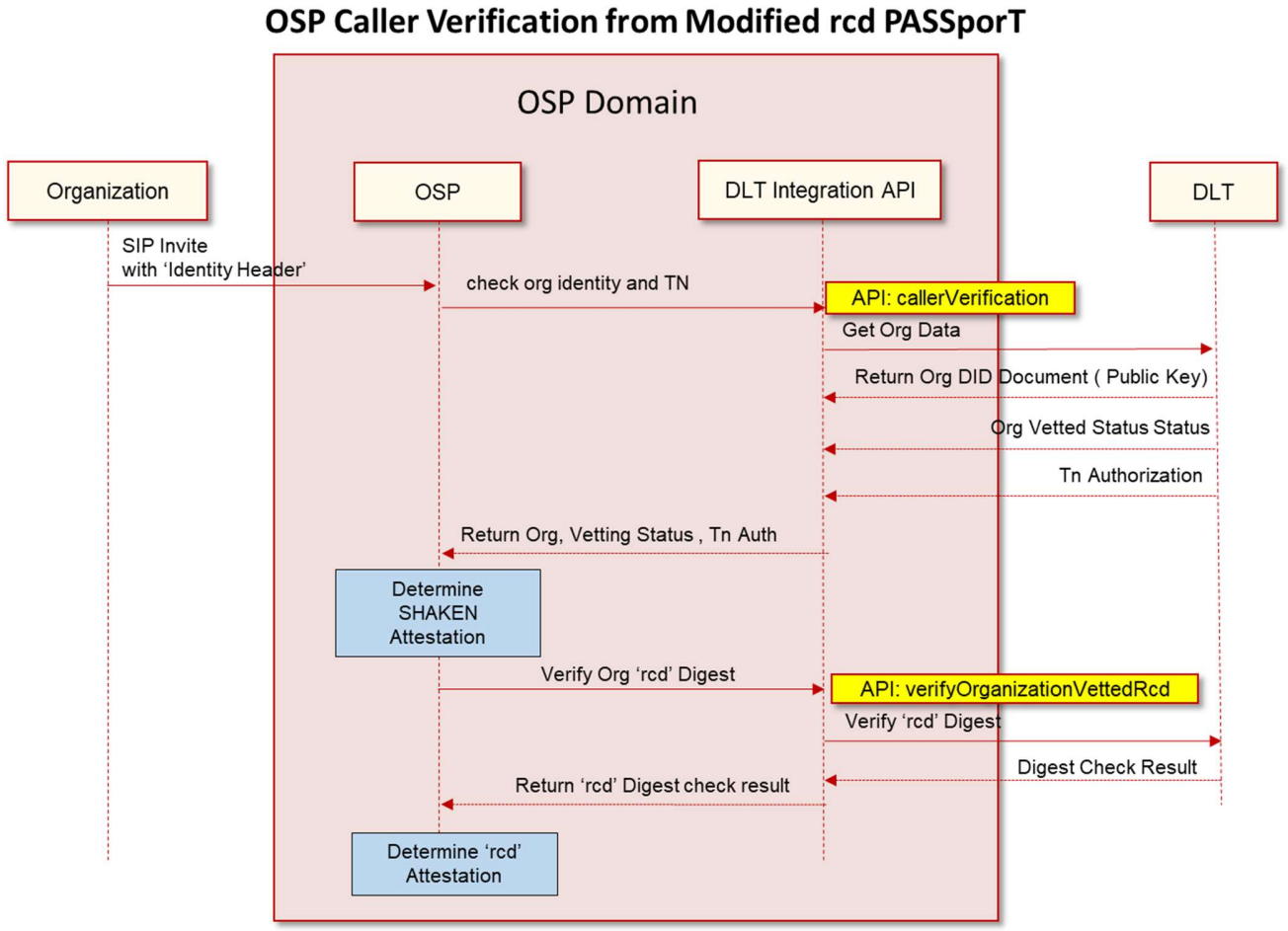

Figure 7.5 API Flow for OSP Caller Verification from Modified rcd PASSporT

8 Traceback using Enterprise Identity

The U.S. Congress-approved TRACED Act of 2019 directs the FCC to facilitate industry's effort to trace back the origin of unlawful robocalls by establishing a consortium that would coordinate these private-led traceback efforts. These efforts are intended to uncover the true origin of unlawful robocalls, the source of which often is masked due to spoofed caller ID information.

Traceback is the process of determining the origin of a call, typically by starting with the receiving party and TSP and tracing backward through the path of the intermediate providers and, ultimately, to the originating service provider and the origin of the call. Traceback can be used to find the source of robocalls and, thus, the entities responsible for those calls.

Two key principals as it relates to traceback that can use the Enterprise Identity DLT Network are:

- **Principle #5**. **Confirm the Identity of Commercial Customers.** Confirm the identity of new commercial VoIP customers by collecting information such as physical business location, contact person(s), state or country of incorporation, federal tax ID and the nature of the customer's business.
- **Principle #6. Require Traceback Cooperation in Contracts.** For all new and renegotiated contracts governing the transport of voice calls, use best efforts to require cooperation in traceback investigations by identifying the upstream provider from which the suspected illegal robocall entered its network or by identifying its own customer if the call originated in its network.

8.1 Principle #5. Confirm the Identity of Commercial Customers.

A business entity can have its commercial identity data verified and contained within a VC claim from the KYC vetting process that it can use with other businesses on the EIDLN.

The KYC vetting process of an Enterprise Identity will authenticate a business entity's identity on the EIDLN for a TNR, brand enterprise or BPO enterprise. As part of the KYC compliance process, the business entity will provide commercial identity information to the KYC vetter to confirm that the business representing the Enterprise Identity is a legitimate business entity and whom it claims to be.

The KYC Vetter compliance process will perform vetting of the enterprise representing the Enterprise Identity checking key commercial identity information. Examples include business trading name, business brand name, contact name, contact details, location, incorporation and federal tax ID. Upon successful vetting of the enterprise's commercial identity information, the KYC vetter can create a signed, verifiable credential claim containing the key commercial identity information verified.

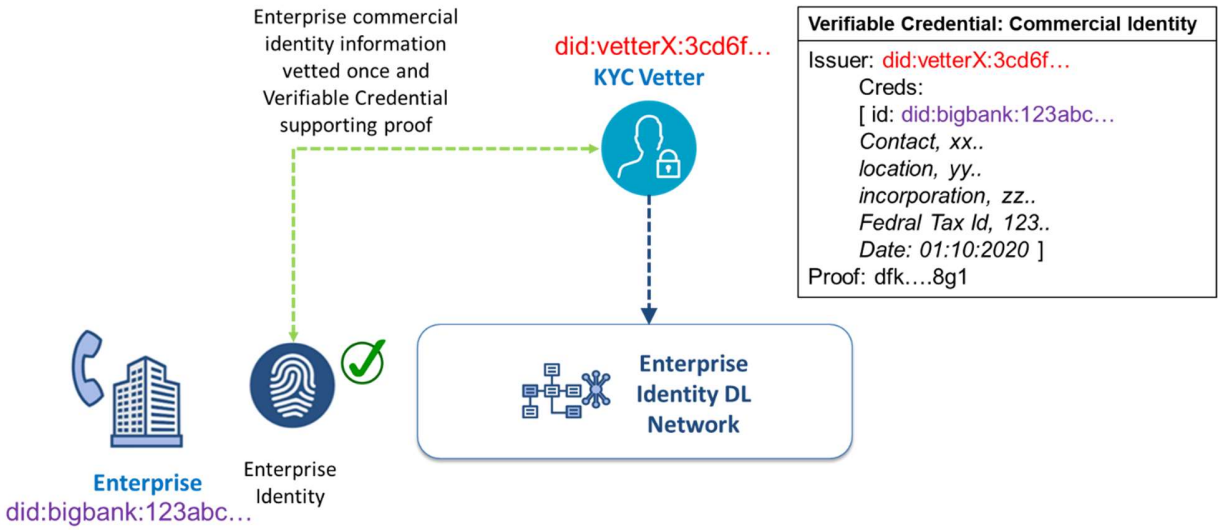

Figure 8.1 KYC Vetter Creation of Commercial Identity Credential Claim

This VC claim containing the commercial identity information can now be used by the enterprise to share and prove that the commercial identity information is vetted and can be used by other actors on the EIDLN with their consent (i.e., a TNSP, TN reseller or an OSP hosting the TN for the enterprise for the first time and required commercial information to satisfy its KYC compliance process.)

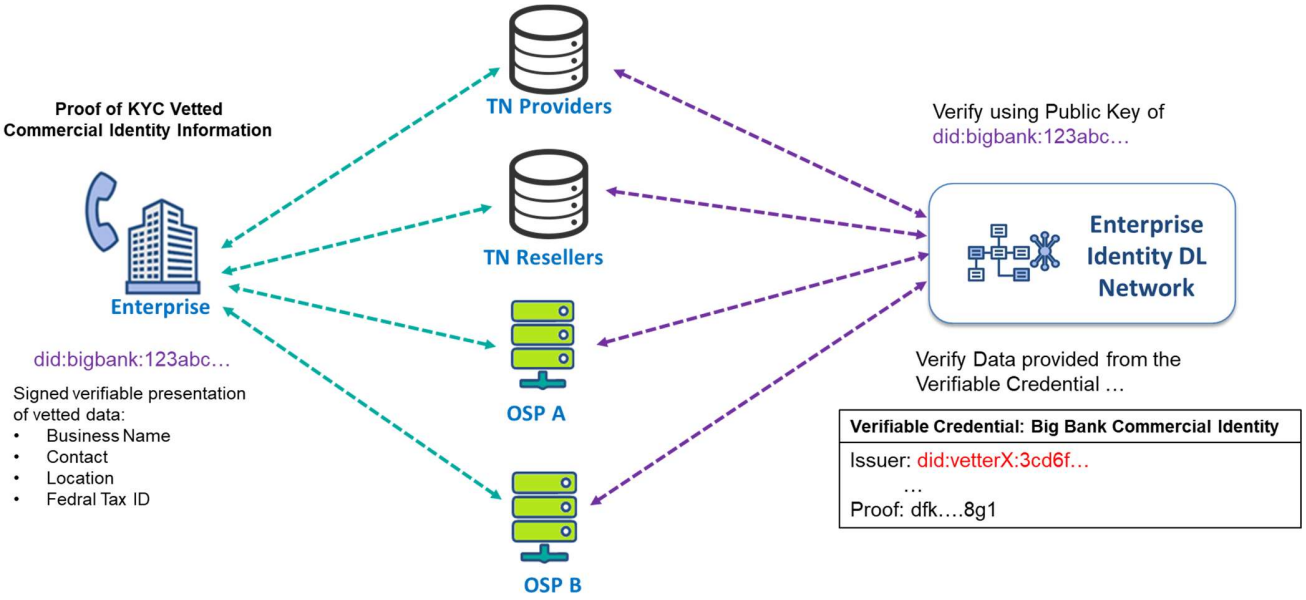

Figure 8.2 Using the VC Commercial Identity to Prove with Multiple Actors

When an enterprise is requested to provide commercial data to another EIDLN actor, it can present verifiable data contained within the VC claim that is proofed by the KYC vetter. The EIDLN actor can then verify that the data provided for this enterprise has been vetted and can be trusted.

8.2 Principle #6. Require Traceback Cooperation in Contracts

Traceback is a cooperative effort by telecommunications providers that starts with a TSP possessing evidence of suspicious traffic. A traceback provider will trace and identify the source of illegal robocalls to provide for the robust protection of voice networks and users of such services from fraudulent, abusive, and/or unlawful robocalls.

When a complainant has reported a suspicious robocall, which has been verified using the EIDLN, to the TSP, it can provide the enterprise identity of the calling party as provided within the Enterprise PASSporT for that call to the traceback provider.

The traceback provider can directly identify the calling party by retrieving the commercial identity data for the enterprise identity from the EIDLN without requesting each upstream service provider's information to identify the calling party.

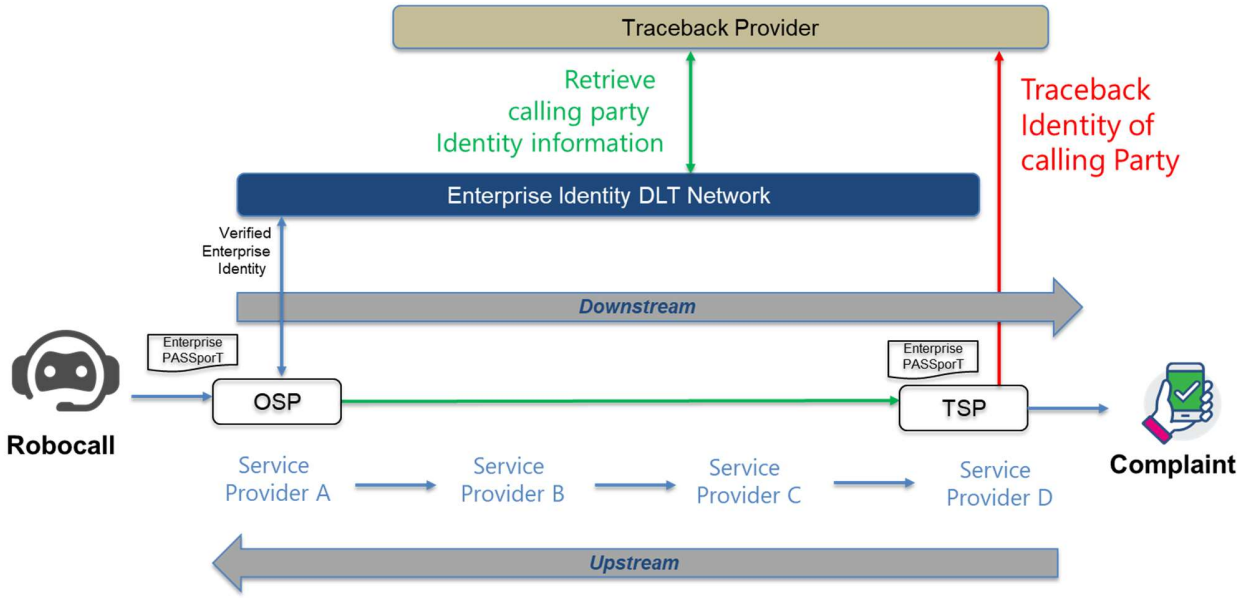

Figure 8.3 Traceback of Robocaller Identity Using EIDLN

9 ANNEX A: Example of Open API for EIDL

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

Note: The EIDLN OpenAPI definition format file 'EDLT_POC_API_Swagger' is available for download from ATIS at https://access.atis.org/

9.1 Organization Accounts

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

9.1.1 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 EIDLN 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 EIDLN 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: did, name, type did: *string* The DID of the account. This is assigned during the creation (POST) process by the EIDLN integration API 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 }

Responses

Status: 201 - Expected response to a valid request

Schema

{

An account to represent a given organization

did: *string*

DID of the account. This is a URI composed of three parts:

- the scheme **'did:'** ,
- a method identifier **'example:'**,
- and a unique, '**method-specific identifier'**, which 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

privateKey: string

The private key generated for this account by the EIDLN integration API.

didDocument: *string*

The DID document of this account in JSON format, including the public key in JWK format.

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

}

9.1.2 getAccount

Retrieve account information from the EIDLN using the DID 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 of the organization* Required

Responses

Status: 200 - Expected response to a valid request

- Schema

{

The account information to represent a given organization DID

did: *string*

DID of the account. This is the DID assigned during the account creation process and is used 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

didDocument: *string*

The DID document of this account in JSON format, including the public key in JWK format.

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

}

9.2 Vetter

9.2.1 createVettedOrganizationOnDLT

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

/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 of the subject organization vetted status change* Required
vetterIdParam*	String *The DID 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 - Revoked* 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 – Expected response to a valid request

Schema

{

Required: organization, status, vetter

organization: {

The DID of the subject organization vetted status change

did: string

DID of the account. This is the DID assigned during the account creation process and is used to identify the organization.

name: string

type: integer (int32)

Type of organization. 2 - Brand, 3 – BPO

privateKey: string

didDocument: string

The DID document of this account in JSON format, including the public key in JWK format.

createdTS: integer (int64)

Time that this account was created in UTC milliseconds from EPOCH

updatedTS: integer(int64)

}

vetter: {

An account to represent a given organization

did: string

DID of the vetter organization, either a TA or KYC vetter account. This is the organization assigned in the 'vetterIdParam*'

name: string

type: integer (int32)

Type of organization. 0 - KYC Vetter, 6 – TA

privateKey: string

didDocument: string

The DID document of this account in JSON format, including the public key in JWK format.

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 - Revoked

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

}

9.2.2 getVettedOrganizationAuthorization

Retrieve the vetted authorization status and history for a given organization

/vetterAuthorization/{orgIdParam}/history/{historyStatus}

Usage and SDK Samples

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

Parameters

Path parameters

Name	Description
orgIdParam*	String *The DID of the subject organization* Required
historyStatus*	Integer(int32) *Type of status response . 0 - Current, 1 - History* Required

Responses

Status: 201 – Returns Vetted Organization Response

Schema

{

Required: organization, vetterAuthorizationList

organization: {

The DID of the subject organization vetted status

did: string

DID of the subject organization.

name: string

type: integer (int32)

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

privateKey: string

didDocument: string

The DID document of this account in JSON format, including the public key in JWK format.

createdTS: string (date-time)

Time that this account was created

updatedTS: string (date-time)

Time that this account was last updated

}

all of:[:List current authorization record if Status =0, return all record history if Status = 1(History)

vetterAuthorizationList:[

vetter: {

An account to represent a given organization

did: string

DID of the TA/ KYC vetter account.

name: string

type: integer (int32)

Type of organization. 0 - KYC Vetter, 6 – TA

privateKey: string

didDocument: string

The DID document of this account in JSON format, including the public key in JWK format.

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

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 - Revoked

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

}

9.2.3 createOrganizationVettedRcdOnDLT

Allows a KYC vetter to update the organization vetted rcd information to the EIDLN

Note: this API call was not implemented for the EIDLN PoC.

9.3 TNAuthorization

9.3.1 createTnAuthrizationOnDLT

Allows an authorized organization to assign or delegate a TN to another organization on the EIDLN.

/ tnAuthorization

Usage and SDK Samples

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

Parameters

Body parameters

Name	Description
body	*{* *tnAuthorization: [* *{* *Required: issuerDID, subjectDID,* authorizationEndDate*, authorizationStartDate, intendedUseofAuthorization, authorizationType* tnId: *string* *Identifier of the telephone number.* issuerDID: *string* *DID of the Organization that is allocating the TN.* subjectDID: *string* *DID of the Organization that the TN is being authorized to.* intendedUseOfAuthorization: *string* *Text field that allows you to specify the intended purpose the TNs in the authorization 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 flexibility to the solution to allow different types of identifiers for TNs in the future.* authorizationType*: string*

	The type of TN authorization, either "Allocation", "Assignment" or "Delegation". *authorizationStartDate: string (date-time)* *Time that this TN Authorization becomes valid. Must be a future date.* authorizationEndDate*: string (date-time)* *Time that this TN Authorization 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 TNAuthorization

- Schema

{

tnAuthorizationResponse:[

{

verifiableCredential:

{

@context:[

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

id: string

URI if the TN Verifiable credential for this authorization

type: string

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

issuer: string

Entity DID that issued the credential (Authorized to use the TN)

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:

tnId: *string*

Identifier of the telephone number.

issuerDID: *string*

DID of the Organization that is allocating the TN.

subjectDID: *string*

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

intendedUseOfAuthorization: *string*

Text field that allows you to specify the intended purpose the TNs in the authorization 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 flexibility to the solution to allow different types of identifiers for TNs in the future.

authorizationType*:* *string*

The type of TN authorization, either 'Allocation', "Assignment" or "Delegation".

authorizationStartDate: string (date-time)

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

authorizationEndDate*:* *string (date-time)*

Time that this TN Authorization *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

}

9.4 OSP Verification

9.4.1 callerVerification

Verification of the caller organization identity and authorization to use a specific TN identity from the EIDLN

/callerVerification/{orgIdParam}/tn/{tnldparam}"

Usage and SDK Samples

curl -X GET "http://localhost/v1/callerVerification/{orgIdParam}/tn/{tnldparam}"

Parameters

Path parameters

Name	Description
orgIdParam*	String *The DID of the subject organization being verified.* Required
tnIdParam*	String Identifier of Telephone Number Required

Responses

Status: 201 - Returns Verification Response

Schema

{

Required: organization, status, tnAuthorizationResponse

organization: {

The DID of the subject organization vetted status change

did: string

DID of the organization being verified

privateKey: string

didDocument: string

The DID document of this account in JSON format, including the public key in JWK format.

createdTS: string (date-time)

Time that this account was created

updatedTS: string (date-time)

Time that this account was last updated

}

vetter: {

Vetting Status of the subject organization DID

did: string

DID of the KYC vetter account.

type: integer (int32)

Type of organization. 0 - KYC Vetter

privateKey: string

didDocument: string

The DID document of this account in JSON format, including the public key in JWK format.

createdTS: string (date-time)

Time that this account was created

updatedTS: string (date-time)

Time that this account was last updated

}

result: integer (int32)

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

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.

tnAuthorizationResponse: {

verifiableCredential:

{

@context:[

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

type: string

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

issuerDID: string

Entity DID that issued the credential (Authorized to use the TN)

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

}

]

}

Status: default - unexpected error

Schema

{

Required: code,message

code: integer (int32)

message: string

}

9.4.2 verifyOrganizationVettedRcd

Verify the vetted ‘rcd’ information for a given organization.

Note: this API call was not implemented for the EIDLN PoC

Contributors to this report

- Charter Communication
- First Orion
- IBM
- Inteliquent
- Metaswitch
- Numeracle
- TNSI
- T-Mobile
- Somos