Architecture Overview

Suggest Edits

Verite’s identity topology is attestation-based. People and institutions make claims about their identities, and others (such as financial institutions and service providers) attest to the validity of those claims by providing cryptographic proofs about them. Once such a cryptographic attestation exists, people can choose to share them with other entities and services that can then verify the attestation as proof of the claim.

This article describes the architectural topology and key components of this model.

Identity Hierarchy

In this architecture, individuals and institutions have one or more identities, and each of those identities in turn may have multiple claims associated with it. This digital hierarchy maps to the physical world in which entities such as individuals and institutions have several identity contexts: A person may have a financial identity, a healthcare identity, a university identity, a social media identity, etc. Each of those unique identity contexts have separate claims associated with them.

Verite focuses on the identity claims and how to prove them to others without sharing more data than is needed (data minimization) and without relying on others to control the claims or control the act of sharing them.

Identity Protocol

Decentralized identity protocols define the way attestations are issued, stored, requested, verified, and revoked.

The protocols enable individuals and institutions to request and receive cryptographic attestations about identity claims from an "issuer". A financial institution issuing attestations about KYC or accredited investor status to its customers is an an example of an issuer.

Individuals and institutions can custody and manage these attestations in their crypto/identity wallets. The attestations take the form of "Verifiable Credentials".

The individuals and institutions who receive Verifiable Credentials about their identities are often referred to as "subjects" and the identity responsible for custodying the Verifiable Credentials is referred to as a "holder." The subject and holder are the same when self-custodying, but they are not the same if the subject uses a hosted wallet service. One example of a subject is a customer who receives a KYC Verifiable Credential from her financial institution. If she custodies this Verifiable Credential in her own wallet, she is also the holder; if she chooses to custody the credential in a hosted wallet service, then the wallet service is the holder.

Individuals and institutions decide when and with whom they want to share their Verifiable Credentials. Those who request and receive Verifiable Credentials are referred to as "verifiers" or more generally as "relying parties". A smart contract requiring proof of KYC is one example of a relying party.

Foundational Standards

Verite endorses the W3C Verifiable Credentials (VCs) Data Model. A credential is a claim made by an issuer about an individual (subject), and a Verifiable Credential is a cryptographically secure wrapper around the credential. Verite also leverages Verifiable Presentations to wrap Verifiable Credentials for transmission and prevent replay attacks. Relying parties can verify a credential without contacting its issuer by employing the extensible mechanism defined by the Verifiable Credentials specification. The verification process is designed to be privacy-respecting and cryptographically secure.

Foundational standards that Verite draws upon include:

- W3C Verifiable Credentials (“VCs”):
- Verifiable Credentials define a standard flexible, tamper-evident way for an issuer to make a claim about a subject in a way that is independently verifiable and privacy-preserving. VCs are the data model used for Verite claims attestations.
- Verifiable Presentations ("VPs"):
- Also defined in the VC spec, Verifiable Presentations provide a way to securely package a set of VCs for transmission to a relying party in a way that allows the subject to prove control over the credentials.
- Identifiers, Decentralized Identifiers:
- An "identifier" refers to both a subject and an issuer of a VC. The identifier data type is a URI, and may be a W3C Decentralized Identifier (DID), which is used in Verite reference implementations. Verite does not require the use of DIDs, but they are one standard way to implement identifiers that minimize correlability and enable proof of control over credentials.
- Verifiable Data Registry:
- This is a general concept that enables decoupling of verification from the issuer. It's typically used for storing revocation registries, trusted issuer lists, or similar data repositories to be accessed during the verification process. In

VC implementations, these are often implemented with a distributed ledger (note that in the Verite approach, the credentials themselves are not stored on-chain).

- Credential Manifest:
- This standard from the Decentralized Identity Foundation (DIF) defines the mechanism for requesting and receiving a credential. A Credential Manifest allows an issuer to describe (in a machine-readable way) what types (schemas) of credentials they issue and what their requirements are. It also describes the format for a subject/holder to submit an application for a credential that conforms to those requirements.
- Presentation Exchange:
- This DIF standard enables a verifier to describe what types of credentials they require from a subject/holder, and how the subject/holder can send a submission.
- Wallet and Credential Interactions:
- Verite's Wallet interaction protocols use a lightweight flow loosely based on the work-in-progress DIF Wallet and Credential Interactions specification.

More About Standards

A Verite primer on VCs, VPs, and DIDs is available forreference here .

Specification References

The complete list of specifications used in Verite's implementation is available forreference here . Updated3 months ago * Table of Contents * * Identity Hierarchy * * Identity Protocol * * Foundational Standards * * * More About Standards * * * Specification References