Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: 20 September 2026

D. Hardman
Provenant, Inc

19 March 2026

Verifiable Voice Protocol
draft-hardman-verifiable-voice-protocol-07

Abstract

Verifiable Voice Protocol (VVP) authenticates and authorizes organizations and individuals making and/or receiving telephone calls. This eliminates trust gaps that malicious parties exploit. Like related technologies such as SHAKEN, RCD, and BCID, VVP uses STIR to bind cryptographic evidence to a SIP INVITE, and verify this evidence downstream. VVP can also let evidence flow the other way, proving things about the callee. VVP builds from different technical and governance assumptions than alternatives, and uses richer, stronger evidence. This allows VVP to cross jurisdictional boundaries easily and robustly. It also makes VVP simpler, more decentralized, cheaper to deploy and maintain, more private, more scalable, and higher assurance. Because it is easier to adopt, VVP can plug gaps or build bridges between other approaches, functioning as glue in hybrid ecosystems. For example, it may justify an A attestation in SHAKEN, or an RCD passport for branded calling, when a call originates outside SHAKEN or RCD ecosystems. VVP also works well as a standalone mechanism, independent of other solutions. An extra benefit is that VVP enables two-way evidence sharing with verifiable text and chat (e.g., RCS and vCon), as well as with other industry verticals that need verifiability in non-telco contexts.

About This Document

This note is to be removed before publishing as an RFC.

The latest revision of this draft can be found at https://dhh1128.github.io/vvp/draft-hardman-verifiable-voice-protocol.html. Status information for this document may be found at https://datatracker.ietf.org/doc/draft-hardman-verifiable-voice-protocol/.

Source for this draft and an issue tracker can be found at https://github.com/dhh1128/vvp.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

When we get phone calls, we want to know who's calling, and why. Often, we want similar information when we _make_ calls as well, to confirm that we've truly reached who we intend. Strangers abuse expectations in either direction, far too often.

Regulators have mandated protections, and industry has responded. However, existing solutions have several drawbacks:

* Assurance of callers derives only from the signatures of originating service providers, with no independently verifiable proof of what they assert.

* Proving the identity of the callee is not supported.

* Each jurisdiction has its own governance and its own set of signers. Sharing information across boundaries is fraught with logistical and regulatory problems.

* Deployment and maintenance costs are high.

* Market complexities such as the presence of aggregators, wholesalers, and call centers that proxy a brand are difficult to model safely.

* What might work for enterprises offers few benefits and many drawbacks for individual callers.

VVP solves these problems by applying crucial innovations in evidence scope, evidence format, and vetting mechanisms. These innovations profoundly upgrade what is provable in an ecosystem, as well as what is cacheable and what must be centralized. However, they have only subtle effects on the content of a STIR PASSporT, so they are explored outside this spec.

2. Conventions and Definitions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Overview

Fundamentally, VVP requires identified parties (callers and/or callees) to curate a dossier ([TOIP-DOSSIER]) of stable evidence that proves things about them. This is done once or occasionally, in advance, as a configuration precondition. Then, for each call, participants decide whether to share this evidence. Callers share evidence by creating an ephemeral STIR-compatible VVP PASSporT ([RFC8225]) that cites (4) their preconfigured dossier. This passport travels along the delivery route as an Identity header in a SIP INVITE. Callees share evidence by adding an analogous passport to an attribute line in the SDP [RFC8866] body of their SIP response. This passes a signed citation to their dossier in the other direction. Verifiers anywhere along the route check the citation(s) and corresponding dossier(s), including realtime revocation status, to make decisions (5).

A VVP call may carry assurance in either or both directions. Compliant implementations may choose to support only assurance about the caller, only assurance about the callee, or both.

3.1. Roles

Understanding the workflow in VVP requires a careful definition of roles related to the protocol. The terms that follow have deep implications for the mental model, and their meaning in VVP may not match casual usage.

3.1.1. Callee

For a given phone call, a _callee_ receives the SIP INVITE. Typically one callee is targeted, but multiparty SIP flows allow INVITEs to multiple callees, either directly or via a conference server (see [RFC4353] and [RFC4575]). A callee can be an individual consumer or an organization. The direct service provider of the callee is the _terminating service provider_ (_TSP_). In many use cases for VVP, callers attempt to prove things to callees, and callees and their service providers use VVP primarily with a verifier mindset. However, enterprises or call centers that accept inbound calls from individuals may want assurance to flow the other direction; hence, VVP supports optional evidence about callees as well.

3.1.2. Originating Party

An _originating party_ (_OP_) controls the first _session border controller_ (_SBC_) that processes an outbound call, and therefore builds the VVP passport that cites evidence about the caller.

It may be tempting to equate the OP with "the caller", and in some perspectives this could be true. However, this simple equivalence lacks nuance and doesn’t always hold. In a VVP context, it is more accurate to say that the OP creates a SIP INVITE [RFC3261] with explicit, provable authorization from the party accountable for calls on the originating phone number. The OP originates the VVP protocol, but not always the call on the handset.

It may also be tempting to associate the OP with an organizational identity like "Company X". While this is not wrong, the precise cryptographic identity of an OP should be narrower. It typically corresponds to a single service operated by an IT department within (or outsourced but operating at the behest of) Company X, rather than to Company X generically. This narrowness limits cybersecurity risk, because a single service operated by Company X needs far fewer privileges than the company as a whole. Failing to narrow identity appropriately creates vulnerabilities in some alternative approaches. The evidence securing VVP MUST therefore prove a valid relationship between the OP’s narrow identity and the broader legal entities that stakeholders more naturally assume and understand.

The service provider associated with an OP is called the _originating service provider_ (_OSP_). For a given phone call, there may be complexity between the hardware that begins a call and the SBC of the OP -- and there may also be many layers, boundaries, and transitions between OSP and TSP.

3.1.3. Accountable Party

For a given call, the _accountable party_ (_AP_) is the organization or individual that has the right to use the originating phone number, according to the regulator of that number. When a callee asks, "Who's calling?", they have little interest in the technicalities of the OP, and it is almost always the AP that they want to identify. The AP is accountable for the call, and thus "the caller", as far as the regulator and the callee are concerned.

APs can operate their own SBCs and therefore be their own OPs. However, APs can also use a UCaaS provider that makes the AP-OP relationship indirect. Going further, a business can hire a call center, and delegate to the call center the right to use its phone number. In such a case, the business is the AP, but the call center is the OP that makes calls on its behalf. None of these complexities alter the fact that, from the callee's perspective, the AP is "the caller". The callee chooses to answer or not, based on their desire to interact with the AP. If the callee's trust is abused, the regulator and the callee both want to hold the AP accountable.

In order to verify a caller, VVP requires an AP to prepare a dossier of evidence that documents a basis for imposing this accountability on them. Only the owner of a given dossier can prove they intend to initiate a VVP call that cites their dossier. Therefore, if a verifier confirms that a particular call properly matches its dossier, the verifier is justified in considering the owner of that dossier the AP for the call. Otherwise, someone is committing fraud. Accountability, and the basis for it, are both unambiguous.

3.1.4. Verified Party

A _verified party_ (_VP_) is a party that uses VVP to prove assertions about itself and its delegation decisions. When VVP provides assurance about callers, the AP is a VP. When VVP provides assurance about callees, the callee is a VP. Some characteristics of proxies, delegates, and service providers may be proved by a dossier, but these parties are not VPs. They don't create dossiers, and dossiers are not focused on them.

3.1.5. Verifier

A _verifier_ is a party that wants to know who's calling or being called, and maybe why -- and that evaluates the answers to these questions by examining formal evidence. Callees, callers, TSPs, OSPs, government regulators, law enforcement doing lawful intercept, auditors, and even APs or OPs can be verifiers. Each may need to see different views of the evidence about a particular phone call, and it

may be impossible to comply with various regulations unless these views are kept distinct -- yet each wants similar and compatible assurance.

In addition to checking the validity of cryptographic evidence, the verifier role in VVP MAY also consider how that evidence matches business rules and external conditions. For example, a verifier can begin its analysis by deciding whether Call Center Y has the right, in the abstract, to make or receive calls on behalf of Organization X using a given phone number. However, VVP evidence allows a verifier to go further: it can also consider whether Y is allowed to exercise this right at the particular time of day when a call occurs, or in a particular jurisdiction, given the business purpose asserted in a particular call.

3.2. Lifecycle

VVP depends on three interrelated activities with evidence:

* Curating

* Citing

* Verifying

Chronologically, evidence must be curated before it can be cited or verified. In addition, some vulnerabilities in existing approaches occur because evidence requirements are too loose. Therefore, understanding the nature of backing evidence, and how that evidence is created and maintained, is a crucial consideration for VVP.

However, curating does not occur in realtime during phone calls, and is out of scope for a network protocol specification. Citing and verifying are the heart of VVP, and implementers will approach VVP from the standpoint of SIP flows [RFC3261], [RFC5626]. Therefore, we leave the question of curation to separate document (for example, [TOIP-DOSSIER]).

4. Citing

4.1. Citing the AP’s dossier

A VVP call that makes the caller verifiable begins when the OP (3.1.2) generates a new VVP passport [RFC8225] that complies with STIR [RFC8224] requirements. In its compact-serialized JWT [RFC7519] form, this passport is then passed as an Identity header in a SIP INVITE [RFC3261]. The passport _constitutes_ lightweight, direct, and ephemeral evidence; it _cites_ and therefore depends upon

comprehensive, indirect, and long-lived evidence (the AP's dossier). Safely and efficiently citing stronger evidence in a dossier is one way that VVP differs from alternatives.

4.1.1. Questions answered by an AP's passport

The passport directly answers at least the following questions:

* What is the cryptographic identity of the OP?
* How can a verifier determine the OP's key state at the time the passport was created?
* How can a verifier identify and fetch more evidence that connects the OP to the asserted AP?
* What brand attributes are asserted to accompany the call?

The first two answers come from the kid header. The third answer is communicated in the required evd claim. The fourth answer is communicated in the optional card and goal claims.

More evidence can then be fetched to indirectly answer the following additional questions:

* What is the legal identity of the AP?
* Does the AP have the right to use the originating phone number?
* Does the AP intend the OP to sign passports on its behalf?
* Does the AP have the right to use the brand attributes asserted for the call?

Dossiers can be further expanded to answer even more questions; such dynamic expansion of the scope of proof is compatible with but not specified by VVP.

4.1.2. Sample passport

An example will help. In its JSON-serialized form, a typical VVP passport for an AP (with some long CESR-encoded hashes shortened by ellipsis for readability) might look like this:

```
{
  "header": {
    "alg": "EdDSA",
    "typ": "passport",
    "ppt": "vvp",
    "kid": "https://agentsrus.net/oobi/EMC.../agent/EAx..."
  },
  "payload": {
    "orig": {"tn": ["+33612345678"]},
    "dest": {"tn": ["+33765432109"]},
    "card": ["NICKNAME:Monde d'Exemples",
      "CHATBOT:https://example.com/chatwithus",
      "LOGO;HASH=EK2...;VALUE=URI:https://example.com/ico64x48.png"],
    "goal": "negotiate.schedule",
    "call-reason": "planifier le prochain rendez-vous",
    "evd": "https://fr.example.com/dossiers/E0F....cesr",
    "origId": "e0ac7b44-1fc3-4794-8edd-34b83c018fe9",
    "iat": 1699840000,
    "exp": 1699840030,
    "jti": "70664125-c88d-49d6-b66f-0510c20fc3a6"
  }
}
```

The semantics of the fields are:

* alg _(required)_ MUST be either "EdDSA" ([RFC8032]), or (for post-quantum) "FN-DSA-512" ([FN-DSA]). Standardizing on best-in-class schemes prevents weaker cryptography from degrading the security guarantees of the ecosystem. The RSA, HMAC, and ES256 algorithms MUST NOT be used. (EdDSA is motivated by compatibility with the vLEI and its associated ACDC ecosystem, which currently uses the Montgomery-to-Edwards transformation.)

* typ _(required)_ Per [RFC8225], MUST be "passport".

* ppt _(required)_ Per [RFC8225], MUST identify the specific PASSporT type -- in this case, "vvp".

* kid _(required)_ MUST be the OOBI of an AID ([TOIP-KERI]) controlled by the OP (3.1.2). An OOBI is a special URL that facilitates ACDC's viral discoverability goals. It returns IANA content-type application/json+cesr, which provides some important security guarantees. The content for this particular OOBI MUST be a KEL ([TOIP-KERI]). Typically the AID in question does not identify the OP as a legal entity, but rather software running on or invoked by the SBC operated by the OP. (The AID that identifies the OP as a legal entity may be controlled by a multisig scheme and thus require multiple humans to create a

signature. The AID for kid MUST be single-sig and, in the common case where it is not the legal entity AID, MUST have a delegate relationship with the legal entity AID that’s proved through formal evidence.)

* orig _(required)_ Although VVP does not depend on SHAKEN, the format of this field MUST conform to SHAKEN requirements ([RFC8588]), for interoperability reasons. It MUST also satisfy one additional constraint, which is that only one phone number is allowed. Despite the fact that a containing SIP INVITE may allow multiple originating phone numbers, only one can be tied to evidence evaluated by verifiers.

* dest _(required)_ For interoperability reasons, MUST conform to SHAKEN requirements.

* card _(optional)_ Contains one or more brand attributes. These are analogous to [RFC9796] or [CTIA-BCID] data, but differ in that they MUST be justified by evidence in the dossier. Because of this strong foundation that interconnects with formal legal identity, they can be used to derive other brand evidence (e.g., an RCD passport) as needed. Individual attributes MUST conform to the VCard standard [RFC6350].

* goal _(optional)_ A machine-readable, localizable goal code, as described informally by [ARIES-RFC-0519]. If present, the dossier MUST prove that the OP is authorized by the AP to initiate calls with this particular goal.

* call-reason _(optional)_ A human-readable, arbitrary phrase that describes the self-asserted intent of the caller. This claim is largely redundant with goal; most calls will either omit both, or choose one or the other. Since call-reason cannot be analyzed or verified in any way, and since it may communicate in a human language that is not meaningful to the callee, use of this field is discouraged. However it is not formally deprecated. It is included in VVP to facilitate the construction of derivative RCD passports which have the property.

* evd _(required)_ MUST be the OOBI of a bespoke ACDC (the dossier, [TOIP-ACDC]) that constitutes a verifiable data graph of all evidence justifying belief in the identity and authorization of the AP, the OP, and any relevant delegations. This URL can be hosted on any convenient web server, and is somewhat analogous to the x5u header in X509 contexts. See below for details.

* origId _(optional)_ Follows SHAKEN semantics.

* iat _(required)_ Follows standard JWT semantics (see [RFC7519]).

* exp _(required)_ Follows standard JWT semantics. As this sets a window for potential replay attacks between the same two phone numbers, a recommended expiration SHOULD be 15 seconds (just long enough for an INVITE to be routed and trigger ringing on a handset), and MUST NOT exceed 60 seconds.

* jti _(optional)_ Follows standard JWT semantics.

4.2. Citing a callee's dossier

Optionally, evidence in VVP can also flow from callee to caller. For privacy reasons, individuals who receive phone calls may choose not to use VVP in this way. However, enterprises and call centers may find it useful as a reassurance to their customers about who they've reached.

In such cases, the callee must have curated a dossier. The format of the callee dossier is identical in schema to that used by a caller. It may therefore introduce evidence of the callee's legal identity, right to use a brand, right to use a TN, delegated authority to a call center proxy or an AI, and so forth. (A callee's dossier might differ in one minor way that doesn't affect the schema: it could prove the right to use a TN that has a DNO flag.)

A reference to the callee's dossier is conveyed by adding a special a=callee-passport:X attribute line to the SDP [RFC8866] body of the callee's 200 OK response. (Optionally, the lines MAY also be added to a 180 Ringing response, to make the callee verifiable earlier, but it MUST appear on the 200 OK response.) The value of this line is a JWT in compact form, with the ;type=vvp suffix. This is exactly compliant with the format used by callers to convey VVP passports in Identity headers. However, Identity headers are not used for callees because existing SIP tooling does not expect or preserve Identity headers on responses. Furthermore, the identity of a callee is primarily of interest to the caller, who is willing to parse the SDP body; it does not need the same full-route auditability as the identity of a caller.

Although dossiers are identical in either direction, the callee JWT has a slightly different schema than a caller's VVP passport. The headers of the JWT match, but kid contains the OOBI of the callee, not of the OP. Two new claims are added to the JWT payload: call-id and cseq. These MUST contain the values of the Call-ID and CSeq values on the preceding SIP INVITE. The iat claim MUST also be present and MUST contain a value from the system clock of the callee. The exp field MAY also be present and use a value chosen by the

callee; if it is missing, this communicates the callee's intention to impose no new timeout logic on the call. The evd field MUST also be present, and MUST contain the OOBI of the callee's dossier. The card and goal claims are also allowed. Other claims MAY be present, but MUST be ignored by compliant implementations that do not understand them. (Because the callee references the specific SIP dialog via call-id and cseq, there is no point in repeating fields that describe the dialog, like orig, dest, and so forth.)

5. Verifying

5.1. Verifying the caller

5.1.1. Algorithm

When a verifier encounters a VVP passport, they SHOULD verify by using an algorithm similar to the following. Optimizations may combine or reorder operations, but MUST achieve all of the same guarantees, in order to be compliant implementations.

1. Analyze the iat and exp claims to evaluate timing. Confirm that exp is greater than iat and also greater than the reference time for analysis (e.g., _now_), and that iat is close enough to the reference time to satisfy the verifier's tolerance for replays. (A replay attack would have to call from the same orig to the same dest with the same iat, within whatever window the verifier accepts. Thirty seconds is a recommended default value.)

2. Confirm that the orig, dest, and iat claims match contextual observations and other SIP metadata. That is, the passport appears aligned with what is known about the call from external sources.

3. Extract the kid header.

4. Fetch the key state for the OP at the reference time from the OOBI in kid. Caches may be used to optimize this, as long as they meet the freshness requirements of the verifier.

5. Use the public key of the OP to verify that the signature on the passport is valid for that key state. On success, the verifier knows that the OP is at least making an assertion about the identity and authorizations of the AP. (When reference time is now, this is approximately the level of assurance provided by existing alternatives to VVP.)

6. Extract the evd field, which references the dossier that constitutes backing evidence.

7. Use the SAID ([TOIP-CESR]) of the dossier as a lookup key to see whether the dossier has already been fully validated. Since dossiers are highly stable, caching dossier validations is recommended.

8. If the dossier requires full validation, perform it. Validation includes checking the signature on each ACDC in the dossier's data graph against the key state of its respective issuer at the time the issuance occurred. Key state is proved by the KEL ([TOIP-KERI]), and checked against independent witnesses.

   Issuance is recorded explicitly in the KEL's overall event sequence, so this check does not require guesses about how to map issuance timestamps to key state events. Subsequent key rotations do not invalidate this analysis.

   Validation also includes comparing data structure and values against the declared schema, plus a full traversal of all chained cryptographically verifiable evidence, back to the root of trust for each artifact. The verifier MUST accept the root of trust as a valid authority on the vital question answered by each credential that depends upon it. The correct relationships among evidence artifacts MUST also be checked (e.g., proving that the issuer of one piece is the issuee of another piece).

9. Check to see whether the revocation status of the dossier and each item it depends on has been tested recently enough, at the reference time, to satisfy the verifier's freshness requirements. If no, check for revocations anywhere in the data graph of the dossier. Revocations are not the same as key rotations. They can be checked much more quickly than doing a full validation. Revocation checks can also be cached, possibly with a different freshness threshold than the main evidence.

10. Assuming that the dossier is valid and has no breakages due to revocation, confirm that the OP is authorized to sign the passport. If there is no delegation evidence, the AP and the OP MUST be identical, and the OP MUST be the issuee of the identity credential; otherwise, the OP MUST be the issuee of a delegated signing credential for which the issuer is the AP.

11. Extract the orig field and compare it to the TNAlloc credential cited in the dossier to confirm that the AP (3.1.3) -- or, if OP is not equal to AP and OP is using their own number, the OP (3.1.2) -- has the right to originate calls with this number.

12. If the passport includes non-null values for the optional card claim, extract that information and check that the brand attributes claimed for the call are justified by a brand credential in the dossier.

13. Check any business logic. For example, if the passport includes a non-null value for the optional goal claim, confirm that the verifier is willing to accept a call with that goal. Or, if the delegated signer credential says that the OP can only call on behalf of the AP during certain hours, or in certain geos, check those attributes of the call.

5.2. Verifying the callee

The callee is verified with an algorithm that MAY be optimized but MUST achieve the same security guarantees as this:

1. Confirm that the call-id and cseq claims match the values of Call-ID and CSeq from the preceding SIP INVITE.

2. Confirm that the iat claim matches contextual observations and other SIP metadata. That is, the timing described by the callee appears aligned with what is known about the call from external sources.

3. If the exp claim is present, analyze the iat and exp claims to evaluate timeout.

4. Extract the kid header.

5. Fetch the key state for the callee at the reference time from the OOBI in kid. Caches may be used to optimize this, as long as they meet the freshness requirements of the verifier.

6. Use the public key of the callee to verify that the signature on the passport is valid for that key state.

7. Extract the evd field, which references the dossier that constitutes backing evidence.

8. Use the SAID ([TOIP-CESR]) of the dossier as a lookup key to see whether the dossier has already been fully validated. Since dossiers are highly stable, caching dossier validations is recommended.

9. Confirm that the dossier was signed (issued) by the same AID that appears in the kid header.

10. If the dossier requires full validation, perform it.

11. Check to see whether the revocation status of the dossier and each item it depends on has been tested recently enough, at the reference time, to satisfy the verifier’s freshness requirements.

12. Compare the callee’s TN to the TNAlloc credential cited in the dossier to confirm that the callee has the right to accept calls at this number.

13. If the passport includes non-null values for the optional card claim, extract that information and check that the brand attributes claimed for the call are justified by a brand credential in the dossier.

14. Check any business logic. For example, if the passport includes a non-null value for the optional goal claim, and the preceding INVITE included a VVP passport that also declared a goal, confirm that the callee’s and caller’s goals overlap (one must be a subset of the other). Or, if the delegated signer credential says that a call center or an AI can accept calls during certain hours, or in certain geos, check those attributes of the call.

5.3. Planning for efficiency

A complete verification of either caller or callee passport, from scratch, is quite rigorous. With no caches, it may take several seconds, much like a thorough validation of a certificate chain. However, much VVP evidence is stable for long periods of time and lends itself to caching, subject to the proviso that revocation freshness must be managed wisely. Since the same dossier is used to add assurance to many calls -- perhaps thousands or millions of calls, for busy call centers -- and many dossiers will reference the same issuers and issuees and their associated key states and KELs ([TOIP-KERI]), caching will produce huge benefits.

Furthermore, because SAIDs and their associated data (including links to other nodes in a data graph) have a tamper-evident relationship, any party can perform validation and compile their results, then share the data with verifiers that want to do less work. Validators like this are not oracles, because consumers of such data need not trust shared results blindly. They can always directly recompute some or all of it from a passport, to catch deception. However, they can do this lazily or occasionally, per their preferred balance of risk/effort.

In toto, these characteristics mean that no centralized registry is required in any given ecosystem. Data can be fetched directly from its source, across jurisdictional boundaries. Because it is fetched from its source, it comes with consent. Privacy can be tuned. Simple opportunistic, uncoordinated reuse (e.g., in or across the datacenters of TSPs) will arise spontaneously and will dramatically improve the scale and efficiency of the system.

5.4. Historical analysis

Normally, a verification algorithm determines whether the passport verifies _now_. (This is the only evaluation that's valid for most JWTs, because they depend on ephemeral key state fetched just in time from x5u). However, a VVP passport can do more. Its kid header references a KEL for the signer's AID ([TOIP-KERI]), and its evd header references a dossier issued by either the AID of the AP or the AID of the callee. Thence it connects to a KEL ([TOIP-KERI]). These data structures provide key state transitions that are timestamped -- both by the controllers of the AIDs, and by their independent witnesses. Although the timestamps are not guaranteed to be perfectly synchronized, they can be compared to establish rough transition times and to detect duplicity.

Using this historical information, it becomes possible to ask whether a VVP passport would have verified at an arbitrary moment in the past. In such framings, the reference time from the verification algorithm is _then_, not _now_. In the normal case where _then_ falls outside a fuzzy range, answers about key state are clear to all observers. In the rare cases where _then_ falls inside a fuzzy range, a state transition was underway but not yet universally known, and a verifier can compute the key state (and thence, the outcome of the verification algorithm) according to their preferred interpretation.

6. Security Considerations

Complying with a specification may forestall certain easy-to-anticipate attacks. However, _it does not mean that vulnerabilities don't exist, or that they won't be exploited_. The overall assurance of VVP requires reasonable vigilance. Given that a major objective of VVP is to ensure security, implementers are strongly counseled to understand the underlying principles, the assumptions, and the ways that choices by their own or other implementations could introduce risk.

Like most cryptographic mechanisms, VVP depends on the foundational assumption that human stakeholders will manage cryptographic keys carefully. VVP enforces this assumption more thoroughly than many existing solutions:

* Parties that issue credentials MUST be identified with AIDs ([TOIP-KERI]) that use witnesses. This guarantees a non-repudiable, publicly accessible audit log of how their key state evolves, and it makes key rotation easy. It also offers compromise and duplicity detection. Via prerotation, it enables recovery from key compromise. AIDs can be upgraded to use quantum-proof signing algorithms without changing the identifier.

* Parties that issue credentials MUST do so using ACDCs ([TOIP-ACDC]) signed by their AID rather than a raw key. This makes evidence revocable. It also makes it stable across key rotation, and prevents retrograde attacks by allowing verifiers to map an issuance or revocation event to an unambiguous key state in the KEL ([TOIP-KERI]).

* Parties that issue credentials SHOULD employ threshold-based multi-signature schemes. This enhances security by distributing signing authority across multiple key holders, reducing the risk of single-point compromise. Threshold-based signatures ensure that no single key compromise undermines the system’s integrity while enabling controlled key recovery and rotation without disrupting credential validity.

Nonetheless, it is still possible to make choices that weaken the security posture of the ecosystem, including at least the following:

* Sharing keys or controlling access to them carelessly

* Issuing credentials with a flimsy basis for trust

* Delegating authority to untrustworthy parties

* Delegating authority without adequate constraints

* Failing to fully verify evidence

Generally understood best practices in cybersecurity will avoid many of these problems. In addition, the following policies that are specific to VVP are strongly recommended:

1. Passports SHOULD have an aggressive timeout (e.g., 30 seconds). Signatures on passports are not anchored in a KEL, and must therefore be evaluated for age with respect to the time they were

received. Overly old passports could be a replay attack (a purported second call with the same orig and dest numbers, using the same backing evidence, soon after the first.)

2. Witnesses (which MUST be used) SHOULD be used in such a way that high availability is guaranteed, and in such a way that duplicity by the controller of an AID is detected. (Verifiers will be able to see the witness policy of each AID controller, and SHOULD decide for themselves whether the party is reliable, depending on what they observe.)

3. Revocations SHOULD be timely, and the timeliness guarantees of issuers SHOULD be published.

4. Watchers SHOULD propagate events to local caches with a low latency, and MUST provide information that allows verifiers to decide whether that latency meets their freshness requirements.

7. IANA Considerations

This document defines a new SDP [RFC8866] session-level attribute:

Attribute name: callee-passport Long-form description: Contains a STIR-compatible passport that references a dossier of evidence about the callee’s identity, brand, and related attributes. Used in 200 OK and/or 180 Ringing responses. Type of attribute: session-level Subject to charset: No Reference: This document

This specification also depends on OOBIs ([TOIP-KERI]) being served as web resources with IANA content type application/cesr.

8. References

8.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/rfc/rfc2119>.

[RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, DOI 10.17487/RFC3261, June 2002, <https://www.rfc-editor.org/rfc/rfc3261>.

[RFC4575] Rosenberg, J., Schulzrinne, H., and O. Levin, Ed., "A Session Initiation Protocol (SIP) Event Package for Conference State", RFC 4575, DOI 10.17487/RFC4575, August 2006, <https://www.rfc-editor.org/rfc/rfc4575>.

[RFC5626] Jennings, C., Ed., Mahy, R., Ed., and F. Audet, Ed., "Managing Client-Initiated Connections in the Session Initiation Protocol (SIP)", RFC 5626, DOI 10.17487/RFC5626, October 2009, <https://www.rfc-editor.org/rfc/rfc5626>.

[RFC8032] Josefsson, S. and I. Liusvaara, "Edwards-Curve Digital Signature Algorithm (EdDSA)", RFC 8032, DOI 10.17487/RFC8032, January 2017, <https://www.rfc-editor.org/rfc/rfc8032>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/rfc/rfc8174>.

[RFC8224] Peterson, J., Jennings, C., Rescorla, E., and C. Wendt, "Authenticated Identity Management in the Session Initiation Protocol (SIP)", RFC 8224, DOI 10.17487/RFC8224, February 2018, <https://www.rfc-editor.org/rfc/rfc8224>.

[RFC8225] Wendt, C. and J. Peterson, "PASSporT: Personal Assertion Token", RFC 8225, DOI 10.17487/RFC8225, February 2018, <https://www.rfc-editor.org/rfc/rfc8225>.

[RFC8588] Wendt, C. and M. Barnes, "Personal Assertion Token (PaSSporT) Extension for Signature-based Handling of Asserted information using toKENs (SHAKEN)", RFC 8588, DOI 10.17487/RFC8588, May 2019, <https://www.rfc-editor.org/rfc/rfc8588>.

[RFC8866] Begen, A., Kyzivat, P., Perkins, C., and M. Handley, "SDP: Session Description Protocol", RFC 8866, DOI 10.17487/RFC8866, January 2021, <https://www.rfc-editor.org/rfc/rfc8866>.

[TOIP-ACDC] Smith, S., Feairheller, P., Griffin, K., Ed., and Trust Over IP Foundation, "Authentic Chained Data Containers (ACDC)", November 2023, <https://trustoverip.github.io/tswg-acdc-specification/>.

[TOIP-CESR]
Smith, S., Griffin, K., Ed., and Trust Over IP Foundation, "Composable Event Streaming Representation (CESR)", November 2023, <https://trustoverip.github.io/tswg-cesr-specification/>.

[TOIP-DOSSIER]
Hardman, D., "Verifiable Dossiers", September 2025, <https://trustoverip.github.io/kswg-dossier-specification/>.

[TOIP-KERI]
Smith, S., Griffin, K., Ed., and Trust Over IP Foundation, "Key Event Receipt Infrastructure (KERI)", January 2024, <https://trustoverip.github.io/tswg-keri-specification/>.

8.2. Informative References

[ARIES-RFC-0519]
Hardman, D., "Aries RFC 0519: Goal Codes", April 2021, <https://github.com/hyperledger/aries-rfcs/blob/main/concepts/0519-goal-codes/README.md>.

[CTIA-BCID]
CTIA, "Branded Calling ID Best Practices", November 2022, <https://api.ctia.org/wp-content/uploads/2022/11/Branded-Calling-Best-Practices.pdf>.

[FN-DSA] NIST, "FIPS 206: FN-DSA (Falcon)", September 2025, <https://csrc.nist.gov/presentations/2025/fips-206-fn-dsa-falcon>.

[RFC4353] Rosenberg, J., "A Framework for Conferencing with the Session Initiation Protocol (SIP)", RFC 4353, DOI 10.17487/RFC4353, February 2006, <https://www.rfc-editor.org/rfc/rfc4353>.

[RFC6350] Perreault, S., "vCard Format Specification", RFC 6350, DOI 10.17487/RFC6350, August 2011, <https://www.rfc-editor.org/rfc/rfc6350>.

[RFC7519] Jones, M., Bradley, J., and N. Sakimura, "JSON Web Token (JWT)", RFC 7519, DOI 10.17487/RFC7519, May 2015, <https://www.rfc-editor.org/rfc/rfc7519>.

[RFC9796] Wendt, C. and J. Peterson, "SIP Call-Info Parameters for Rich Call Data", RFC 9796, DOI 10.17487/RFC9796, July 2025, <https://www.rfc-editor.org/rfc/rfc9796>.

Acknowledgments

Much of the cybersecurity infrastructure used by VVP depends on KERI, which was invented by Sam Smith, and first implemented by Sam plus Phil Fairheller, Kevin Griffin, and other technical staff at GLEIF. Thanks to logistical support from Trust Over IP and the Linux Foundation, and to a diverse community of technical experts in those communities and in the Web of Trust group.

Techniques that apply KERI to telco use cases were developed by Daniel Hardman, Randy Warshaw, and Ruth Choueka, with additional contributions from Dmitrii Tychinin, Yaroslav Lazarev, Arshdeep Singh, and many other staff members at Provenant, Inc. Thanks as well to Ed Eykholt for multiple editorial improvements.

Author’s Address

Daniel Hardman
Provenant, Inc
Email: daniel.hardman@gmail.com