vcon
Internet-Draft
Intended status: Standards Track
Expires: 22 January 2026

T. McCarthy-Howe
Strolid, Inc.
S. Lasker
Independent
21 July 2025

Voice Conversation (vCon) Consent Attachment
draft-howe-vcon-consent-00

Abstract

This document defines a consent attachment type for Voice Conversations (vCon), establishing standardized mechanisms for recording, verifying, and managing consent information within conversation containers as defined in the vCon core specification. The consent attachment addresses privacy compliance challenges through structured metadata including consenting parties, temporal validity periods, and cryptographic proof mechanisms.

The specification defines the mandatory and optional fields for consent attachments, including expiration timestamps, party references, dialog segments, and consent arrays. It supports granular consent management through purpose-based permissions and integrates with the AI Preferences vocabulary for automated processing systems. The attachment type incorporates SCITT (Supply Chain Integrity, Transparency, and Trust) for cryptographic transparency and provides integration patterns for consent ledger services.

Key features include automated consent detection during conversation processing, auditable consent records with cryptographic proofs, support for consent revocation through superseding statements, and integration with existing privacy regulations. The consent attachment enables organizations to maintain compliance while providing sufficient structure for automated processing and verification of consent throughout the vCon lifecycle.

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://github.com/vcon-dev/draft-howe-vcon-consent. Status information for this document may be found at https://datatracker.ietf.org/doc/draft-howe-vcon-consent/.

Discussion of this document takes place on the Virtualized Conversations Working Group mailing list (mailto:vcon@ietf.org), which is archived at https://mailarchive.ietf.org/arch/browse/vcon/. Subscribe at https://www.ietf.org/mailman/listinfo/vcon/.

Source for this draft and an issue tracker can be found at https://github.com/vcon-dev/draft-howe-vcon-consent.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 22 January 2026.

Copyright Notice

Table of Contents

1. Introduction 4
2. Terminology 5
  2.1. Core Terms 5
  2.2. vCon-Specific Terms 5
  2.3. Consent-Specific Terms 6
  2.4. Technical Terms 6
3. Regulatory Terms 7

4. Requirements 8
4.1. Consent Attachment Requirements Summary 8
4.2. Core Requirements 8
4.3. Temporal Management 8
4.4. Referential Integrity 9
4.5. Status Monitoring 9
4.6. 3.1.5. Terms of Service Integration 9
4.7. 3.1.6. Compliance and Audit 9
5. Consent Attachment Structure 9
6. Temporal Validity and Expiration 10
7. Terms of Service Integration 10
8. Party and Dialog References 10
9. Status Monitoring and Intervals 11
9.1. Recommended Interval Values 11
10. Consent Ledger Integration 12
10.1. SCITT Transparency Service Interface 12
10.2. Consent Statement Format 12
11. Consent Ledger Operations 13
11.1. Registration 13
11.1.1. Status Verification 13
11.1.2. Receipt Resolution 13
11.2. Consent Revocation 14
11.3. Implementation Requirements 14
11.4. Error Handling 15
11.5. Privacy Considerations 15
12. Cryptographic Proof Requirements 15
13. Granular Consent Management 16
13.1. AI Preferences Vocabulary Support 16
14. Consent Withdrawal and Revocation 17
15. Privacy and Data Minimization 18
16. Error Handling and Validation 18
17. Interoperability and Versioning 19
18. Security Considerations 19
19. Attachment Fields 20
19.1. Required Fields 20
19.2. Optional Fields 21
20. Consent Objects 21
21. Proof Objects 21
22. IANA Considerations 21
22.1. vCon Attachment Type Registration 21
23. How to Use the Consent Attachment 22
23.1. Creating a Consent Attachment 22
23.2. Example Usage 22
23.3. Processing Consent Attachments 23
24. Privacy Considerations 23
25. Security Considerations 24
25.1. Cryptographic Protection 24
25.2. Access Control 24

25.3. Network Security 24
25.4. Data Protection 24
25.5. Threat Mitigation 25
26. References 25
27. References 25
27.1. Normative References 25
27.2. Informative References 26
Acknowledgments 26
Authors' Addresses 26

1. Introduction

Voice conversations often contain sensitive information that requires proper consent management. This document defines a consent attachment type for Virtualized Conversations (vCon) that enables automated consent detection, structured consent recording, and proof mechanisms for compliance with privacy regulations.

A vCon (Virtualized Conversation) is a standardized container format for storing conversation data, including metadata, participants, and conversation content, as defined in [I-D.draft-ietf-vcon-core-00]. The vCon specification provides an overview of the technology and use cases [I-D.draft-ietf-vcon-overview-00], while the core document defines the data model and structure. vCons support extensible attachments that can carry additional structured data related to the conversation. These attachments enable the association of various types of information with the conversation, such as transcripts, analytics, or consent records.

The consent attachment type provides a standardized mechanism for recording and managing consent information within vCon containers. This attachment captures essential consent metadata including the consenting party, the specific dialog or conversation segment covered by the consent, temporal validity periods, and any associated proof mechanisms. By embedding consent information directly within the vCon structure, implementations can maintain the integrity of the consent record and ensure it remains associated with the relevant conversation data throughout the lifecycle of the vCon.

The consent attachment addresses key privacy and compliance challenges faced by organizations handling voice conversations. It enables automated consent detection during conversation processing, provides auditable consent records, and supports regulatory compliance through structured consent management. The attachment type is designed to be flexible enough to accommodate various consent models while providing sufficient structure to enable automated processing and verification.

2. Terminology

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 [RFC8174] when, and only when, they appear in all capitals, as shown here.

2.1. Core Terms

Consent: Any freely given, specific, informed, and unambiguous indication of the data subject’s wishes by which they, by a statement or by a clear affirmative action, signify agreement to the processing of personal data relating to them [GDPR].

Data Subject: The identified or identifiable natural person to whom personal data relates [GDPR]. Also referred to as "consumer" in some jurisdictions [CCPA].

Data Controller: The natural or legal person, public authority, agency, or other body which, alone or jointly with others, determines the purposes and means of the processing of personal data [GDPR].

Data Processor: A natural or legal person, public authority, agency, or other body which processes personal data on behalf of the controller [GDPR].

Personal Data: Any information relating to an identified or identifiable natural person [GDPR].

Processing: Any operation or set of operations performed on personal data [GDPR].

2.2. vCon-Specific Terms

vCon: A standardized container for conversational information, including metadata, participants, dialog content, analysis, and attachments [I-D.draft-ietf-vcon-core-00].

vCon Instance: A vCon populated with data for a specific conversation [I-D.draft-ietf-vcon-core-00].

vCon Instance Version: A single version of an instance of a conversation, which may be modified to redact or append additional information [I-D.draft-ietf-vcon-core-00].

Party: An observer or participant to the conversation, either passive or active [I-D.draft-ietf-vcon-core-00].

Dialog: The captured conversation in its original form (e.g., text, audio, or video) [I-D.draft-ietf-vcon-core-00].

Analysis: Analysis, transformations, summary, sentiment, or translation typically of the dialog data [I-D.draft-ietf-vcon-core-00].

Attachment: A data block either included or referenced in a vCon [I-D.draft-ietf-vcon-core-00].

2.3. Consent-Specific Terms

Consent Attachment: A vCon attachment of type "consent" that contains structured consent information for one or more parties to a conversation.

Consent Statement: A structured representation of consent that includes the consenting party, purposes, temporal validity, and proof mechanisms.

Consent Ledger: A SCITT Transparency Service that maintains authoritative consent state and provides cryptographic receipts for consent operations.

Consent Proof: Cryptographic or other evidence that validates the authenticity and integrity of a consent statement.

Consent Purpose: A specific use case or processing activity for which consent is granted or denied.

Consent Status: The current state of consent (granted, denied, revoked, expired) for a specific purpose.

Consent Expiration: The point in time when consent becomes invalid, either through explicit expiration or revocation.

Consent Revocation: The withdrawal of previously granted consent by the data subject.

Consent Verification: The process of validating that consent is current, valid, and applicable to the requested processing activity.

2.4. Technical Terms

SCITT: Supply Chain Integrity, Transparency, and Trust - a protocol for maintaining append-only transparency ledgers [I-D.draft-ietf-scitt-scrapi-05].

SCRAPI: SCITT Receipt API - the interface for interacting with SCITT Transparency Services [I-D.draft-ietf-scitt-scrapi-05].

COSE: CBOR Object Signing and Encryption - a standard for creating and processing signed, encrypted, and MACed objects [RFC8949].

COSE Sign1: A COSE structure containing a single signature [RFC8949].

AI Preferences Vocabulary: A standardized vocabulary for expressing preferences related to automated processing systems [I-D.draft-ietf-aipref-vocab-01].

Clock Skew: The difference in time between different systems, which must be accounted for when validating temporal constraints.

Referential Integrity: The consistency of references between consent attachments and the vCon elements they reference.

Data Minimization: The principle of limiting personal data collection to what is necessary for the specified purpose [GDPR].

Privacy by Design: The principle of embedding privacy considerations into the design and architecture of systems and processes [NIST-PRIVACY].

3. Regulatory Terms

GDPR: General Data Protection Regulation - the primary data protection law in the European Union [GDPR].

CCPA: California Consumer Privacy Act - a state statute intended to enhance privacy rights and consumer protection for residents of California [CCPA].

HIPAA: Health Insurance Portability and Accountability Act - a US law that provides data privacy and security provisions for safeguarding medical information [HIPAA].

Right to be Forgotten: The right of data subjects to have their personal data erased [GDPR].

Right of Access: The right of data subjects to obtain confirmation of whether their personal data is being processed and access to that data [GDPR].

Right of Rectification: The right of data subjects to have inaccurate personal data corrected [GDPR].

Right of Portability: The right of data subjects to receive their personal data in a structured, commonly used, machine-readable format [GDPR].

Lawful Basis: One of the six legal grounds for processing personal data under GDPR: consent, contract, legal obligation, vital interests, public task, or legitimate interests [GDPR].

4. Requirements

4.1. Consent Attachment Requirements Summary

The vCon consent specification establishes standardized requirements for managing consent information within voice conversation containers as defined in [I-D.draft-ietf-vcon-core-00]. The specification addresses privacy compliance challenges through structured consent attachments that capture essential metadata and proof mechanisms.

4.2. Core Requirements

Mandatory Fields: Consent attachments must include four essential fields:

* *expiration*: RFC3339 timestamp indicating consent validity period

* *party*: Zero-based index referencing the consenting party in the vCon parties array

* *dialog*: Zero-based index referencing the specific conversation segment

* *consents*: Array containing the actual consent records and permissions

Optional Fields: Additional fields may include terms of service references, status monitoring intervals, consent ledger URLs, and cryptographic proof mechanisms.

4.3. Temporal Management

Consent attachments must implement proper expiration handling with configurable clock skew tolerance (default maximum 5 minutes). Indefinite consent is supported through null expiration values but requires periodic revalidation mechanisms.

4.4. Referential Integrity

The specification enforces strict referential integrity between consent attachments and vCon elements. Party and dialog references must be validated against the vCon structure, and modifications to the underlying conversation data require corresponding updates to consent attachments.

4.5. Status Monitoring

Consent status must be verified at configurable intervals based on data sensitivity:

* High-sensitivity data: 24-hour verification cycles
* Standard business data: 7-day verification cycles
* Low-sensitivity data: 30-day verification cycles
* Public data: 90-day verification cycles

4.6. 3.1.5. Terms of Service Integration

Terms of service references must be immutable and support both URI references and embedded content. Implementations should maintain local caching with proper HTTP cache control compliance.

4.7. 3.1.6. Compliance and Audit

The specification enables automated consent detection, auditable consent records, and regulatory compliance through structured consent management. It supports various consent models while providing sufficient structure for automated processing and verification.

5. Consent Attachment Structure

The consent attachment MUST be a JSON object that is included in the vCon attachments array. The consent attachment MUST contain the following top-level fields: "expiration", "party", "dialog" and "consents". Implementations MUST reject consent attachments that are missing any of these required fields.

The consent attachment MAY contain the following top-level fields "terms_of_service", "status_interval", "consent_ledger", "proof".

The consent attachment SHOULD include appropriate metadata fields as defined in the vCon core specification [I-D.draft-ietf-vcon-core-00], including "type", "start", and "signature" fields when applicable.

6. Temporal Validity and Expiration

The "expiration" field MUST contain a timestamp in RFC3339 format indicating when the consent becomes invalid. Implementations MUST compare the current time against this expiration timestamp and SHALL reject expired consent attachments.

The expiration timestamp MAY be set to null to indicate indefinite consent duration, as defined by local policy or applicable regulations. When the expiration field is null, the consent is considered valid indefinitely until explicitly revoked. However, implementations SHOULD provide mechanisms for periodic consent revalidation even when indefinite consent is granted.

When processing consent attachments, implementations MUST account for clock skew and SHOULD allow for reasonable time differences between systems. The acceptable clock skew tolerance SHOULD be configurable but MUST NOT exceed 5 minutes by default.

7. Terms of Service Integration

The "terms_of_service" field MUST contain either a URI reference to the applicable terms of service document or an embedded terms object. When using URI references, implementations MUST support HTTPS URLs and SHOULD support content integrity verification through cryptographic hashes.

The terms of service reference MUST be immutable for the lifetime of the consent attachment. If terms of service change, a new consent attachment MUST be created rather than modifying the existing reference.

Implementations SHOULD maintain a local cache of terms of service documents to ensure availability during consent verification. The cache SHOULD respect HTTP cache control headers when fetching terms documents via URI.

8. Party and Dialog References

The "party" field MUST contain a zero-based integer index referencing an entry in the vCon parties array. Implementations MUST validate that the referenced party index exists in the vCon and SHOULD verify that the party has authority to grant consent for the specified dialog.

The "dialog" field MUST contain a zero-based integer index referencing an entry in the vCon dialog array. Multiple consent attachments MAY reference the same dialog entry to represent consent from different parties or for different purposes.

Implementations MUST maintain referential integrity between consent attachments and the referenced vCon elements. When a vCon is modified through redaction or amendment, consent attachment references MUST be updated accordingly or the consent MUST be invalidated.

9. Status Monitoring and Intervals

The "status_interval" field MUST specify the maximum time interval, in seconds, between consent status verifications. Implementations MUST perform consent status checks at least as frequently as specified by this interval.

The status interval MUST be a positive integer value. A status interval of zero indicates that consent status MUST be verified on every access to the associated dialog content.

9.1. Recommended Interval Values

The following intervals are RECOMMENDED based on privacy regulation requirements and operational considerations:

* *High-sensitivity data*: 86400 seconds (24 hours) - For medical, financial, or legal conversations
* *Standard business data*: 604800 seconds (7 days) - For typical business communications
* *Low-sensitivity data*: 2592000 seconds (30 days) - For general customer service interactions
* *Public/transparent data*: 7776000 seconds (90 days) - For non-private communications that do not contain personally identifiable information not in the public domain.

Implementations SHOULD consider the following factors when selecting intervals:

* Data sensitivity and regulatory requirements (GDPR, CCPA, HIPAA, etc.)
* Risk tolerance and compliance obligations

* Operational overhead of frequent verification

* User experience impact of verification delays

* Consent revocation patterns and requirements

When the status interval expires, implementations MUST either verify consent status through the consent ledger mechanism or treat the consent as potentially invalid until verification is completed. Implementations MAY continue to honor consent during brief verification delays but MUST NOT exceed twice the specified status interval without successful verification.

10. Consent Ledger Integration

The "consent_ledger" field SHOULD contain a URL referencing a SCITT Transparency Service that maintains the authoritative consent state. Implementations MAY use this URL to verify current consent status and detect consent revocations.

10.1. SCITT Transparency Service Interface

Consent ledger services MUST implement the SCRAPI interface as specified in [I-D.draft-ietf-scitt-scrapi-05]. The consent ledger acts as a SCITT Transparency Service that:

1. *Registers Consent Statements*: Consent attachments are registered as Signed Statements using the /entries endpoint

2. *Issues Receipts*: Provides cryptographic receipts proving consent registration

3. *Enables Verification*: Allows verification of consent status through receipt validation

4. *Supports Revocation*: Handles consent revocation through statement updates

10.2. Consent Statement Format

Consent statements MUST be formatted as COSE Sign1 objects containing:

```
{
  "protected": {
    "alg": "ES256",
    "kid": "consent-issuer-key-id",
    "cty": "application/vcon-consent+json"
  },
  "unprotected": {
    "consent_id": "urn:uuid:consent-identifier",
    "vcon_uuid": "urn:uuid:vcon-identifier",
    "party_index": 0,
    "dialog_index": 0
  },
  "payload": {
    "consents": [
      {
        "purpose": "recording",
        "status": "granted",
        "timestamp": "2025-01-02T12:15:30Z"
      }
    ],
    "expiration": "2026-01-02T12:00:00Z",
    "terms_of_service": "https://example.com/terms"
  }
}
```

11. Consent Ledger Operations

11.1. Registration

* *Endpoint*: POST /entries
* *Content-Type*: application/cose
* *Body*: COSE Sign1 consent statement
* *Response*: 201 with receipt or 303 with location for async processing

11.1.1. Status Verification

* *Endpoint*: GET /entries/{entry-id}
* *Response*: 200 with receipt or 302/404 for pending/not found

11.1.2. Receipt Resolution

* *Endpoint*: GET /entries/{entry-id}

* *Response*: 200 with current receipt or 404 if not found

11.2. Consent Revocation

Consent revocation is handled by registering a new Signed Statement that supersedes the original consent:

```
{
  "protected": {
    "alg": "ES256",
    "kid": "consent-issuer-key-id",
    "cty": "application/vcon-consent+json"
  },
  "unprotected": {
    "consent_id": "urn:uuid:consent-identifier",
    "vcon_uuid": "urn:uuid:vcon-identifier",
    "supersedes": "urn:uuid:original-consent-id",
    "revocation": true
  },
  "payload": {
    "consents": [
      {
        "purpose": "recording",
        "status": "revoked",
        "timestamp": "2025-01-03T10:30:00Z"
      }
    ],
    "revocation_reason": "user_request",
    "revocation_timestamp": "2025-01-03T10:30:00Z"
  }
}
```

11.3. Implementation Requirements

When a consent ledger URL is provided, implementations MUST:

1. *Support SCRAPI*: Implement all mandatory SCRAPI endpoints

2. *Use HTTPS*: All communications MUST use TLS 1.2 or higher

3. *Authenticate*: Implement appropriate authentication as specified by the ledger service

4. *Handle Failures*: Gracefully handle service unavailability

5. *Verify Receipts*: Validate all receipts using SCITT verification procedures

6. *Cache Strategically*: Cache consent state within status_interval limits

11.4. Error Handling

Consent ledger services MUST return appropriate HTTP status codes and Concise Problem Details objects as specified in SCRAPI:

* *400*: Malformed consent statement
* *401*: Authentication required
* *403*: Registration policy violation
* *404*: Consent not found
* *429*: Rate limiting exceeded
* *503*: Service temporarily unavailable

11.5. Privacy Considerations

Consent ledger services SHOULD implement privacy-preserving mechanisms:

* *Minimal Data*: Only store consent metadata, not full vCon content
* *Access Control*: Implement appropriate access controls for consent queries
* *Audit Logging*: Maintain audit trails for compliance purposes
* *Data Retention*: Implement appropriate data retention policies

12. Cryptographic Proof Requirements

The "proof" field MUST contain an array of proof objects that provide cryptographic evidence of valid consent. Each proof object MUST include a "type" field specifying the proof mechanism and a "value" field containing the proof data.

Proof objects MAY include various types of evidence:

* *Dialog-based proofs*: Consent statements or confirmations given verbally or textually within the conversation dialog itself

* *External document proofs*: Signed consent forms, terms of service agreements, or other legal documents referenced by URL or embedded content

* *External site proofs*: Consent granted through web interfaces, mobile applications, or other external systems that provide cryptographic attestation

Implementations MUST support digital signature proofs using algorithms specified in the IANA COSE Algorithms registry [COSE-ALG]. The proof SHOULD include a timestamp indicating when the consent was granted and MUST include a reference to the consenting party's cryptographic identity.

When multiple proofs are present, implementations MUST verify all proofs and SHALL reject the consent attachment if any proof fails validation. The proof verification process MUST include validation of the signing key authority and certificate chain when applicable.

Proof objects MAY reference external proof data to minimize consent attachment size. When external references are used, implementations MUST verify the integrity of externally referenced proof data using cryptographic hashes included in the proof object.

13. Granular Consent Management

The "consents" field MUST contain an array of consent objects, each specifying a particular permission granted by the consenting party. Each consent object MUST include a "purpose" field identifying the specific use case and a "status" field indicating whether consent is granted or denied.

Standard consent purposes MUST include "recording", "transcription", "analysis", "storage", and "sharing". Implementations MAY define additional purpose categories but SHOULD use standardized purpose taxonomies when available.

13.1. AI Preferences Vocabulary Support

Implementations SHOULD support the AI Preferences vocabulary as defined in [I-D.draft-ietf-aipref-vocab-01] for expressing granular consent related to automated processing systems. When using the AI Preferences vocabulary, consent objects MAY include the following standardized categories:

* *tdm*: Text and Data Mining - Automated analytical techniques for analyzing text and data

* *ai*: AI Training - Training machine learning models or artificial intelligence

* *genai*: Generative AI Training - Training AI models that generate synthetic content

* *search*: Search - Building search indexes or providing search results

* *inference*: AI Inference - Using assets as input to trained AI/ML models

When the AI Preferences vocabulary is used, consent objects SHOULD include a "vocabulary" field set to "ai-pref" to indicate the use of this standardized vocabulary. The "purpose" field SHOULD use the corresponding label from the AI Preferences vocabulary (e.g., "ai", "genai", "tdm").

Example consent object using AI Preferences vocabulary:

```
{
  "purpose": "genai",
  "status": "denied",
  "vocabulary": "ai-pref",
  "timestamp": "2025-01-02T12:15:30Z"
}
```

Implementations that support the AI Preferences vocabulary MUST follow the hierarchical relationship rules defined in [I-D.draft-ietf-aipref-vocab-01], where more specific categories override general categories unless explicitly stated otherwise.

Each consent object SHOULD include additional metadata such as the timestamp when consent was granted, any restrictions or conditions on the consent, and references to applicable legal regulations.

Implementations MUST support fine-grained consent evaluation, allowing different consent decisions for different purposes. When multiple consent objects apply to the same purpose, the most restrictive consent MUST take precedence.

14. Consent Withdrawal and Revocation

Implementations MUST support consent withdrawal mechanisms that allow data subjects to revoke previously granted consent. When consent is revoked, the consent status in the ledger service MUST be updated immediately and all processing of the associated dialog content MUST cease.

The consent withdrawal process MUST provide confirmation to the data subject and SHOULD include a timestamp of when the revocation became effective. Implementations MUST honor consent revocations even if they cannot immediately delete or modify existing vCon instances.

When consent is revoked, implementations SHOULD notify all parties that have received copies of the vCon containing the revoked consent. The notification mechanism MAY use the SCITT transparency service integration described in the vCon overview [I-D.draft-ietf-vcon-overview-00] and related vCon lifecycle specifications.

15. Privacy and Data Minimization

Consent attachments MUST implement data minimization principles by including only information necessary for consent verification and audit purposes. Personal information SHOULD NOT be duplicated in consent attachments when it is already available in the referenced vCon elements.

The consent attachment design MUST support privacy-preserving verification mechanisms that allow consent validation without exposing sensitive personal information to all parties in a consent verification workflow.

Implementations SHOULD support consent attachment redaction techniques that allow sensitive consent details to be removed while maintaining the ability to verify that valid consent was originally present.

16. Error Handling and Validation

Implementations MUST validate consent attachment syntax according to the JSON schema defined in this specification. Malformed consent attachments MUST be rejected with appropriate error messages indicating the specific validation failures.

When consent validation fails, implementations MUST NOT process the associated dialog content and SHOULD log the failure for audit purposes. The error handling process MUST distinguish between temporary failures (such as network timeouts during ledger verification) and permanent failures (such as invalid cryptographic proofs).

Implementations SHOULD provide detailed error reporting to assist with troubleshooting consent validation issues while avoiding exposure of sensitive information in error messages.

17. Interoperability and Versioning

Consent attachments MUST include version information to support evolution of the consent attachment specification. Implementations MUST handle version mismatches gracefully and SHOULD support multiple consent attachment versions during transition periods.

The consent attachment format MUST be designed to support extensions while maintaining backward compatibility with existing implementations. New fields MAY be added to consent attachments but MUST NOT break existing validation logic.

Implementations SHOULD support consent attachment format negotiation when multiple parties exchange vCons with different consent attachment version support.

18. Security Considerations

All consent attachments MUST be integrity protected using the vCon signing mechanisms as defined in [I-D.draft-ietf-vcon-core-00]. Consent attachments containing sensitive information SHOULD be encrypted when the vCon is transmitted outside secure environments.

Implementations MUST protect consent ledger communications using TLS and SHOULD implement additional authentication mechanisms to prevent unauthorized consent status queries or modifications.

The consent attachment design MUST prevent replay attacks and consent forgery through appropriate use of timestamps, nonces, and cryptographic binding to the specific vCon instance and dialog content.

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.

```
{
  "expiration": "2026-01-02T12:00:00Z",
  "terms_of_service": "https://example.com/terms",
  "party": 0,
  "dialog": 0,
  "status_interval": "30d",
  "consent_ledger": "https://ledger.example.com/consent/123",
  "proof": [
    {
      "timestamp": "2025-01-02T12:15:30Z",
      "type": "verbal_confirmation"
    },
    {
      "url": "https://example.com/consent-form.pdf",
      "type": "external_asset"
    }
  ],
  "consents": [
    {
      "value": true,
      "consent": "recording"
    },
    {
      "value": true,
      "consent": "transcription"
    },
    {
      "value": true,
      "consent": "analysis"
    }
  ]
}
```


19. Attachment Fields

19.1. Required Fields

* *expiration*: ISO 8601 timestamp indicating when consent expires
* *party*: Index of the party in the vCon parties array
* *dialog*: Index of the dialog in the vCon dialog array
* *consents*: Array of consent objects with value and consent type

19.2. Optional Fields

* *terms_of_service*: URL to the terms of service document
* *status_interval*: Duration string (e.g., "30d") for status check intervals
* *consent_ledger*: URL to external consent ledger for audit purposes
* *proof*: Array of proof objects supporting the consent

20. Consent Objects

Each consent object MUST contain:

* *value*: Boolean indicating consent status (true for granted, false for denied)
* *consent*: String identifying the consent type (e.g., "recording", "transcription", "analysis")

21. Proof Objects

Proof objects provide evidence of consent and MAY contain:

* *timestamp*: ISO 8601 timestamp when consent was given
* *type*: Proof type identifier (e.g., "verbal_confirmation", "external_asset")
* *url*: URL to external proof document (for external_asset type)

22. IANA Considerations

22.1. vCon Attachment Type Registration

This document requests IANA to register the following vCon attachment type:

* *Type*: consent
* *Description*: Consent information for voice conversation participants
* *Reference*: This document

23. How to Use the Consent Attachment

23.1. Creating a Consent Attachment

To create a consent attachment for a vCon:

1. *Identify the consenting party*: Reference the party index in the vCon parties array

2. *Specify the dialog segment*: Reference the dialog index in the vCon dialog array

3. *Define consent permissions*: Create consent objects for each purpose

4. *Set expiration*: Define when the consent expires

5. *Add proof mechanisms*: Include cryptographic or other proof of consent

6. *Include in vCon*: Add the attachment to the vCon attachments array

23.2. Example Usage

```
{
  "type": "consent",
  "start": "2025-01-02T12:15:30Z",
  "expiration": "2026-01-02T12:00:00Z",
  "party": 0,
  "dialog": 0,
  "consents": [
    {
      "purpose": "recording",
      "status": "granted",
      "timestamp": "2025-01-02T12:15:30Z"
    }
  ],
  "proof": [
    {
      "type": "verbal_confirmation",
      "timestamp": "2025-01-02T12:15:30Z"
    }
  ]
}
```

23.3. Processing Consent Attachments

When processing a vCon with consent attachments:

1. *Validate structure*: Ensure all required fields are present
2. *Check expiration*: Verify the consent hasn't expired
3. *Verify references*: Ensure party and dialog indices are valid
4. *Validate proofs*: Verify cryptographic or other proof mechanisms
5. *Check ledger status*: If a consent ledger is specified, verify current status
6. *Apply permissions*: Use consent status to determine allowed operations

24. Privacy Considerations

This document describes mechanisms for managing consent information within vCon containers as defined in [I-D.draft-ietf-vcon-core-00]. Implementers MUST ensure compliance with applicable privacy regulations, including but not limited to GDPR [GDPR], CCPA [CCPA], and HIPAA [HIPAA].

The consent attachment provides structured mechanisms for recording and verifying consent, but implementers are responsible for ensuring that their implementations properly respect and enforce data subject rights, including:

* *Right of Access*: Data subjects must be able to access their consent records
* *Right of Rectification*: Data subjects must be able to correct inaccurate consent information
* *Right to be Forgotten*: Data subjects must be able to revoke consent and request deletion
* *Right of Portability*: Data subjects must be able to export their consent data
* *Consent Withdrawal*: Data subjects must be able to withdraw consent at any time

Implementers SHOULD follow privacy by design principles [NIST-PRIVACY] and implement appropriate technical and organizational measures to protect personal data throughout the consent lifecycle.

25. Security Considerations

Consent attachments contain sensitive information that requires appropriate security measures:

25.1. Cryptographic Protection

* All consent attachments MUST be integrity protected using vCon signing mechanisms
* Consent attachments containing sensitive information SHOULD be encrypted when transmitted outside secure environments
* Implementations MUST use strong cryptographic algorithms as specified in [COSE-ALG]

25.2. Access Control

* Implementations MUST implement appropriate access controls for consent data
* Consent verification SHOULD be performed with minimal privilege
* Audit logging MUST be implemented for all consent operations

25.3. Network Security

* All communications with consent ledger services MUST use TLS 1.2 or higher
* Implementations MUST validate certificate chains and hostnames
* Implementations SHOULD implement certificate pinning for critical services

25.4. Data Protection

* Consent data SHOULD be encrypted at rest
* Implementations MUST implement secure key management
* Implementations SHOULD support hardware security modules for key storage

25.5. Threat Mitigation

* Implementations MUST prevent replay attacks through proper use of timestamps and nonces

* Implementations MUST validate all cryptographic proofs to prevent forgery

* Implementations SHOULD implement rate limiting to prevent abuse

26. References

{backmatter}

27. References

27.1. Normative References

[I-D.draft-ietf-aipref-vocab-01]
Keller, P. and M. Thomson, "A Vocabulary For Expressing AI Usage Preferences", Work in Progress, Internet-Draft, draft-ietf-aipref-vocab-01, June 2025, <I-D.draft-ietf-aipref-vocab-01>.

[I-D.draft-ietf-scitt-scrapi-05]
Birkholz, H. and J. Geater, "SCITT Reference APIs", Work in Progress, Internet-Draft, draft-ietf-scitt-scrapi-05, July 2025, <I-D.draft-ietf-scitt-scrapi-05>.

[I-D.draft-ietf-vcon-core-00]
Petrie, D., "Voice Conversation (vCon) Core Data Model", Work in Progress, Internet-Draft, draft-ietf-vcon-core-00, March 2025, <I-D.draft-ietf-vcon-core-00>.

[I-D.draft-ietf-vcon-overview-00]
McCarthy-Howe, T., "Voice Conversation (vCon) Overview", Work in Progress, Internet-Draft, draft-ietf-vcon-overview-00, March 2025, <I-D.draft-ietf-vcon-overview-00>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", RFC 2119, BCP 14, March 1997, <RFC2119>.

[RFC3339] Klyne, G. and C. Newman, "Date and Time on the Internet: Timestamps", RFC 3339, July 2002, <RFC3339>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", RFC 8174, BCP 14, May 2017, <RFC8174>.

[RFC8949] Bormann, C. and P. Hoffman, "Concise Binary Object Representation (CBOR)", RFC 8949, STD 94, December 2020, <RFC8949>.

27.2. Informative References

[CCPA] State of California, "California Consumer Privacy Act", 2018, <https://oag.ca.gov/privacy/ccpa>.

[COSE-ALG] IANA, "COSE Algorithms", July 2025, <https://www.iana.org/assignments/cose/cose.xhtml>.

[GDPR] European Union, "General Data Protection Regulation", 2018, <https://gdpr.eu/>.

[HIPAA] U.S. Department of Health and Human Services, "Health Insurance Portability and Accountability Act", 1996, <https://www.hhs.gov/hipaa/index.html>.

[NIST-PRIVACY] National Institute of Standards and Technology, "NIST Privacy Framework", 2020, <https://www.nist.gov/privacy-framework>.

Acknowledgments

* Thank you to Diana James and Cody Launius for their collaboration and imagination.

Authors' Addresses

Thomas McCarthy-Howe
Strolid, Inc.
Email: thomas.howe@strolid.com

S. Lasker
Independent
Email: stevenlasker@hotmail.com