Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: 30 September 2026

J. van de Meent
R. AI
Humotica

29 March 2026

JIS: JTel Identity Standard - Identity and Trust Establishment for Autonomous Agents

draft-vandemeent-jis-identity-01

Abstract

This document defines JIS (JTel Identity Standard), a protocol for establishing identity, negotiating trust, and binding intent declarations to actor interactions. JIS provides three core mechanisms: a dual-keypair identity model separating human-device binding (HID) from device authentication (DID), a trust establishment handshake (FIR/A) that negotiates capabilities and records intent, and a human-readable context layer (Humotica) that captures the sense, context, intent, and explanation for every interaction.

JIS is transport-agnostic and operates as a semantic layer above existing protocols. It integrates with TIBET [TIBET] for provenance tracking, and is consumed by UPIP [UPIP], RVP [RVP], and AINS [AINS] for process integrity, continuous verification, and agent discovery respectively.

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 30 September 2026.

Copyright Notice

Table of Contents

1. Introduction 3
1.1. Problem Statement 4
1.2. Design Principles 4
1.3. Scope 5
2. Terminology 5
3. Identity Model 6
3.1. HID (Human Identity Key) 6
3.2. DID (Device Identity Key) 7
3.3. HID-DID Binding 7
3.4. Actor Identifier Format 7
3.5. Key Lifecycle 8
4. Trust Establishment (FIR/A) 8
4.1. Overview 8
4.2. Phase 1: INITIATE 8
4.3. Phase 2: CAPABILITIES 9
4.4. Phase 3: CONFIRM 10
4.5. Phase 4: EXECUTE 10
4.6. Trust Score Model 11
4.7. Failure Conditions 11
5. Context Layer (Humotica) 12
5.1. Structure 12
5.2. Minimum Required Context 12
5.3. Semantic Validation 12
5.4. Mapping to TIBET Components 13
6. Intent Validation 13
6.1. Validation as Evidence 13
6.2. Blocked Intent Patterns 14
6.3. Risk Scoring (BALANS) 14
6.4. Dialogue Resolution (NIR) 15
7. TIBET Integration 15
7.1. Identity as TIBET Precondition 15
7.2. FIR/A Events as TIBET Tokens 15
7.3. Continuity Chain Delegation 16
8. Transport Considerations 16
8.1. Baseline: JSON over HTTPS 16
8.2. Alternative Bindings 16
9. Privacy Considerations 16

9.1. HID Protection 17
9.2. Pseudonymous Operation 17
9.3. Data Minimization 17
10. Security Considerations 18
10.1. Identity Protection 18
10.2. Semantic Validation Limitations 18
10.3. FIR/A Handshake Attacks 18
10.4. Key Compromise and Rotation 19
10.5. Replay Protection 19
10.6. Trust Score Manipulation 19
11. IANA Considerations 20
11.1. Media Type Registration 20
12. References 20
12.1. Normative References 20
12.2. Informative References 21
Appendix A. Complete Flow Example 21
A.1. Bank Fraud Verification 22
Appendix B. Conformance Levels 23
B.1. JIS Basic 23
B.2. JIS Extended 23
B.3. JIS Full 23
Appendix C. Changes from -00 24
Acknowledgements 25
Authors' Addresses 25

1. Introduction

AI agents, IoT devices, automated services, and human operators increasingly interact across trust boundaries -- across organizations, protocols, and jurisdictions. These interactions require answers to three questions that existing protocols address incompletely:

1. WHO is acting? (Identity)

2. WHY are they acting? (Intent)

3. SHOULD they be trusted? (Trust)

Existing identity protocols (OAuth 2.0, SAML, FIDO2) solve authentication -- proving WHO. But they do not capture WHY an action is requested, nor do they provide a mechanism for bilateral trust negotiation where both parties declare intent and agree on capabilities before proceeding.

JIS addresses this gap by defining:

* A dual-keypair identity model where human identity (HID) never leaves the device and device identity (DID) handles all external authentication.

* A trust establishment handshake (FIR/A) where both parties declare intent, negotiate capabilities, and establish a trust relationship with a cryptographic genesis record.

* A human-readable context layer (Humotica) that makes every interaction auditable by non-technical reviewers.

1.1. Problem Statement

Three structural problems motivate JIS:

PROTOCOL FRAGMENTATION: When N systems communicate, they require up to N*(N-1)/2 pairwise integrations. JIS reduces this to N adapters by providing a protocol-agnostic identity and intent layer that sits above transport.

REACTIVE SECURITY: Traditional firewalls and access control systems react to attack patterns. They evaluate WHAT is requested against rules. They cannot evaluate WHY it is requested, because no standardized mechanism exists for declaring intent as part of the request.

CONTEXT BLINDNESS: Systems process requests without understanding the situation: a payment at 3 AM from a new country, a 4th failed attempt to turn on heating by a frustrated user, a model inference request from an agent with no established history. Without context, systems cannot differentiate legitimate unusual behavior from threats.

1.2. Design Principles

EVIDENCE OVER ENFORCEMENT: JIS validates intent and produces evidence. Whether to block, allow, or escalate is a local policy decision. Intent validation failures are recorded as evidence, not treated as access denials.

PRIVACY FIRST: Human identity (HID) never leaves the device. Only the device identity (DID) and HID-DID binding hash are used in external communication. This ensures that human identity remains private even if all communications are intercepted.

TRANSPORT AGNOSTICISM: JIS messages are JSON [RFC8259] objects that can be carried over any transport. HTTP, WebSocket, MQTT, SIP, Matrix, CoAP, gRPC, and Bluetooth are all suitable.

COMPANION INTEGRATION: JIS provides identity and trust. Provenance tracking is delegated to TIBET [TIBET]. Process integrity to UPIP [UPIP]. Continuous verification to RVP [RVP]. Discovery to AINS [AINS].

1.3. Scope

This document defines:

* The JIS identity model (HID, DID, bindings)

* The FIR/A trust establishment handshake

* The Humotica context structure

* Intent validation as an evidence mechanism

* Integration points with companion protocols

This document does NOT define:

* Audit trail format (see TIBET [TIBET])

* Process integrity or handoff (see UPIP [UPIP])

* Continuous identity verification (see RVP [RVP])

* Agent discovery and resolution (see AINS [AINS])

* Specific enforcement policies

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 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

Actor An entity participating in a JIS interaction. Actors include human users, AI agents (IDDs), IoT devices, automated services, and organizations.

BALANS Risk scoring system. Produces a score from 0.0 to 1.0 based on multiple factors. Used as evidence input for access decisions.

DID (Device Identity Key) An Ed25519 key pair used for device

authentication. The public key is shared during FIR/A. The private key remains on the device.

FIR/A (First Initiation Revoke/Accept) The trust establishment handshake. Produces a genesis record that anchors all subsequent interactions between two parties.

HID (Human Identity Key) An X25519 key pair binding a human to a device. The private key MUST NEVER leave the device. Only the binding hash (HID-DID binding) is used externally.

Humotica A structured context object capturing sense, context, intent, and explanation for an interaction. Named for its focus on human-readable, empathetic context.

IDD (Individual Device Derivative) An AI agent with unique identity, evolved from base models through interaction and context accumulation. Defined conceptually; not a protocol primitive.

NIR (Notify, Identify, Rectify) A dialogue-based resolution protocol for ambiguous situations. Used when intent validation produces insufficient confidence.

3. Identity Model

3.1. HID (Human Identity Key)

HID is an X25519 key pair that binds a human identity to a device.

```
{
  "hid": {
    "algorithm": "X25519",
    "public_key": "<base64 X25519 public key>",
    "created_at": "<ISO-8601 timestamp>",
    "device_bound": true
  }
}
```

The HID private key MUST NEVER leave the device. The HID private key MUST NOT be transmitted over any channel. The HID private key SHOULD be stored in a hardware secure element (TEE, SE) when available.

The HID public key is used ONLY for computing the HID-DID binding hash (Section 3.3). The HID public key itself SHOULD NOT be transmitted. Only the binding hash is shared.

3.2. DID (Device Identity Key)

DID is an Ed25519 key pair used for device authentication and token signing.

```
{
  "did": {
    "algorithm": "Ed25519",
    "public_key": "<base64 Ed25519 public key>",
    "created_at": "<ISO-8601 timestamp>"
  }
}
```

The DID public key is shared during FIR/A establishment (Section 4). The DID private key MUST remain on the device and is used for signing TIBET tokens and FIR/A messages.

3.3. HID-DID Binding

The HID-DID binding proves that a specific human is associated with a specific device, without revealing the human’s identity key.

Computation:

```
binding_hash = SHA-256(HID_public_key || DID_public_key)
```

SHA-256 is computed per [FIPS180-4]. The binding_hash is:

* Shared during FIR/A as proof of human-device association
* Verifiable by any party that has seen both public keys
* Computed locally; neither key needs to be transmitted

This design ensures that compromising the device (obtaining DID private key) does not compromise the human’s identity key (HID private key).

3.4. Actor Identifier Format

JIS defines three actor identifier formats:

```
jis:<entity_type>:<identifier>
```

Entity types:

"human" A human user. Example: "jis:human:jasper_2025"

"idd" An AI agent (IDD). Example: "jis:idd:root_idd_2025"

"service" An automated service. Example:
"jis:service:payment_gateway"

The identifier portion SHOULD be meaningful and stable. Implementations MUST NOT embed personal data in the identifier (e.g., do not use email addresses).

Note: The -00 version used "did:jtel:" as a URI format. This has been replaced with "jis:" to avoid confusion with the W3C DID specification [DID-CORE]. A W3C DID method specification for JIS identities may be developed separately if there is implementation demand.

3.5. Key Lifecycle

Key Rotation: DID keys SHOULD be rotated periodically. The rotation event MUST be recorded as a TIBET token linking the old and new key. The new key MUST be introduced via a FIR/A re-establishment signed with the old key.

Key Revocation: A compromised key MUST be revoked by publishing a signed revocation notice. Tokens signed with a revoked key after the revocation timestamp SHOULD be treated as suspect. The revocation mechanism is defined by the deployment context.

HID Replacement: HID replacement (e.g., new device) requires in-person or high-assurance verification. The new HID-DID binding MUST reference the previous binding in its TIBET token.

4. Trust Establishment (FIR/A)

4.1. Overview

FIR/A (First Initiation Revoke/Accept) is a four-phase handshake that establishes a bilateral trust relationship. Unlike TLS or OAuth, FIR/A is not just authentication -- it is intent negotiation. Both parties declare what they want, agree on what they can do, and create a cryptographic genesis record that anchors all subsequent interactions.

4.2. Phase 1: INITIATE

The initiator sends a trust establishment request:

```
{
  "type": "fira_init",
  "version": "1.1",
  "initiator": "jis:service:bank_fraud",
  "responder": "jis:human:alice",
  "did_public_key": "<base64 Ed25519 public key>",
  "intent": "fraud_verification_call",
  "humotica": {
    "sense": "Transaction flagged by fraud detection",
    "context": "EUR 5000 transfer to unrecognized recipient",
    "intent": "Verify transaction legitimacy with account holder",
    "explanation": "Automated fraud detection on
                    unusual transfer pattern."
  },
  "timestamp": "2026-03-29T10:30:00.000Z",
  "nonce": "<random 32 bytes, base64>"
}
```

Required fields: type, version, initiator, responder, did_public_key, intent, humotica, timestamp, nonce.

The nonce MUST be cryptographically random and MUST NOT be reused. It provides replay protection.

4.3. Phase 2: CAPABILITIES

The responder evaluates the initiation request and returns available capabilities and rules:

```
{
  "type": "fira_capabilities",
  "fira_id": "fira-2026-03-29-bank-alice-7f3a",
  "responder": "jis:human:alice",
  "did_public_key": "<base64 Ed25519 public key>",
  "hid_did_binding": "sha256:4f2e8a...",
  "capabilities": ["voice_call", "sms_verification"],
  "rules": {
    "no_calls_after": "22:00",
    "require_caller_id": true,
    "max_attempts": 3,
    "language": "nl"
  },
  "timestamp": "2026-03-29T10:30:01.000Z",
  "init_nonce": "<echo of initiator nonce>",
  "nonce": "<responder random 32 bytes, base64>"
}
```

The fira_id is generated by the responder and uniquely identifies this trust establishment session.

The init_nonce field echoes the initiator's nonce, proving the response is to this specific initiation (not a replay).

4.4. Phase 3: CONFIRM

The initiator accepts the capabilities and confirms the trust relationship:

```
{
  "type": "fira_confirm",
  "fira_id": "fira-2026-03-29-bank-alice-7f3a",
  "accepted_capabilities": ["voice_call"],
  "genesis_hash": "sha256:7f3a...c2e1",
  "resp_nonce": "<echo of responder nonce>",
  "signature": {
    "algorithm": "Ed25519",
    "value": "<base64 signature over canonical confirm>"
  }
}
```

The genesis_hash is computed as:

```
genesis_hash = SHA-256(
  canonical(fira_init) || canonical(fira_capabilities)
)
```

This binds the trust relationship to the specific messages exchanged. Both parties can independently verify the genesis.

4.5. Phase 4: EXECUTE

After confirmation, both parties have:

* Each other's DID public keys
* Agreed-upon capabilities
* A shared genesis_hash anchoring the relationship
* A fira_id for referencing the trust context

All subsequent interactions SHOULD reference the fira_id and SHOULD produce TIBET tokens linked to the genesis.

4.6. Trust Score Model

FIR/A produces an initial trust score based on the establishment process. The score is between 0.0 and 1.0:

0.8 - 1.0: HIGH Full capabilities, minimal verification

0.5 - 0.8: MODERATE Standard verification at each step

0.2 - 0.5: LOW Enhanced verification, limited capabilities

0.0 - 0.2: MINIMAL Most capabilities restricted

Trust scores are LOCAL. Each party computes its own score for the counterparty. Scores are evidence, not assertions. Publishing a trust score does not obligate other parties to accept it.

Trust scores adjust over time based on:

* Consistency of behavior with stated intent
* Quality of Humotica context provided
* Chain integrity of associated TIBET tokens
* Duration and depth of interaction history

The exact scoring algorithm is a local policy decision. This document defines the score range and inputs, not the formula.

4.7. Failure Conditions

FIR/A establishment MUST fail (and produce evidence) when:

1. NONCE_MISMATCH: Echoed nonce does not match sent nonce. Indicates replay or man-in-the-middle.
2. HUMOTICA_MISSING: Initiation lacks required Humotica context (Section 5.2).
3. NO_COMMON_CAPABILITY: No overlap between offered and requested capabilities.
4. SIGNATURE_INVALID: Confirm signature does not verify.
5. TIMEOUT: Response not received within configured window (RECOMMENDED default: 30 seconds).

Each failure MUST be recorded as a TIBET token with ERACHTER explaining the failure reason.

5. Context Layer (Humotica)

5.1. Structure

Humotica provides human-readable context for every interaction. The name reflects its focus on empathetic, human-understandable communication.

```
{
  "humotica": {
    "sense": "What triggered this interaction?",
    "context": "What is the current situation?",
    "intent": "What does the actor want to achieve?",
    "explanation": "Why is this action being taken?"
  }
}
```

Fields:

"sense" (string) The trigger or stimulus. What event, observation, or signal initiated this interaction.

"context" (string) The current situation. Environmental state, history, and relevant circumstances.

"intent" (string) The declared goal. What the actor wants to accomplish.

"explanation" (string) The reasoning. Why this particular action is being taken to achieve the intent.

5.2. Minimum Required Context

For FIR/A establishment, the Humotica object MUST contain all four fields. Each field MUST be a non-empty string with at least 10 characters.

For ongoing interactions within an established FIR/A session, Humotica is RECOMMENDED but not required for every message. When provided, all four fields MUST be present.

5.3. Semantic Validation

Implementations MAY perform semantic validation on Humotica context:

* Coherence check: Does the explanation logically support the intent?

* Completeness check: Are all four fields substantive (not placeholder text)?

* Consistency check: Does the declared intent match the actual requested action?

Semantic validation results are EVIDENCE. They inform trust scoring and risk assessment. They SHOULD NOT be used as binary access control (evidence over enforcement).

Note: The -00 version stated that malware "cannot provide legitimate Humotica context." This is an overstatement. A sufficiently sophisticated adversary can craft plausible context. Semantic validation raises the bar for attackers but is not an absolute defense. Defense in depth, combining Humotica with behavioral analysis (RVP [RVP]) and chain integrity (TIBET [TIBET]), provides stronger assurance.

5.4. Mapping to TIBET Components

Humotica fields map to TIBET provenance components:

Humotica Field	TIBET Component
sense	ERIN (trigger)
context	EROMHEEN
intent	ERIN (intent)
explanation	ERACHTER

When a JIS interaction produces a TIBET token, the Humotica fields SHOULD be distributed across the TIBET components per this mapping.

6. Intent Validation

6.1. Validation as Evidence

JIS defines intent validation as an evidence-producing mechanism, not an enforcement mechanism. When an intent is evaluated, the result is recorded as evidence in a TIBET token. Whether the action proceeds, is blocked, or requires additional verification is a local policy decision.

6.2. Blocked Intent Patterns

Implementations SHOULD maintain a configurable list of intent patterns that are flagged for enhanced scrutiny. Examples:

"sql_injection" No legitimate Humotica explanation for injecting SQL into input fields.

"command_injection" No legitimate Humotica explanation for injecting shell commands.

"unauthorized_resource_access" Accessing resources not covered by established capabilities.

Flagged intents SHOULD trigger enhanced verification (NIR, Section 6.4), not silent blocking. The decision to block is a policy choice, not a protocol requirement.

6.3. Risk Scoring (BALANS)

BALANS provides multi-factor risk scoring from 0.0 to 1.0.

Input factors:

"complexity" Higher complexity operations receive lower scores.

"humotica_quality" Vague or short explanations reduce score.

"trust_history" New or untrusted actors receive lower scores.

"transaction_value" High-value operations reduce score.

"temporal_anomaly" Actions at unusual times reduce score.

The BALANS score is evidence attached to the interaction’s TIBET token in EROMHEEN. It does not directly determine access.

Suggested thresholds (local policy):

* score >= 0.5: Normal processing
* 0.3 <= score < 0.5: Trigger NIR dialogue
* score < 0.3: Flag for human review

6.4. Dialogue Resolution (NIR)

NIR (Notify, Identify, Rectify) is a structured dialogue for resolving ambiguous situations instead of silent blocking.

1. NOTIFY: Inform the actor that additional verification is needed. Include the reason in human-readable form. Example: "This transaction was flagged because it exceeds your usual transfer amount."

2. IDENTIFY: Request additional identity evidence. This may be a biometric check (RVP [RVP]), a second-factor confirmation, or a human-readable explanation.

3. RECTIFY: Based on the additional evidence, either proceed (with enhanced logging) or halt (with full evidence record).

NIR produces TIBET tokens at each step, creating an audit trail of the dialogue itself.

7. TIBET Integration

7.1. Identity as TIBET Precondition

The fundamental coupling between JIS and TIBET is:

```
Traditional:  [Auth] -> [Action] -> [Log]
JIS/TIBET:    [Identity + Intent] -> [Validate] -> [Action + Audit]
```

In the JIS/TIBET model, the audit record is not a side effect of the action. The identity and intent declaration (JIS) together with the evidence record (TIBET) are architecturally intertwined with the action itself.

7.2. FIR/A Events as TIBET Tokens

Each phase of FIR/A produces a TIBET token:

* INITIATE produces a TIBET token (type: "action", ERIN: initiation details, ERACHTER: why trust is being established)

* CAPABILITIES produces a child TIBET token

* CONFIRM produces a child TIBET token with genesis_hash in ERAAN

* EXECUTE produces child TIBET tokens for each subsequent action

The FIR/A genesis_hash is stored in the CONFIRM token's ERAAN, linking the trust relationship to the provenance chain.

7.3. Continuity Chain Delegation

The -00 version defined a separate "Continuity Chain" with HMAC linking. In -01, this is simplified: the TIBET chain IS the continuity chain. TIBET's hash-chained, signed tokens provide the same tamper-evidence guarantees as the -00 Continuity Chain, without duplicating the mechanism.

Implementations that require HMAC-based chain integrity (e.g., for backward compatibility) MAY implement it as an additional layer, but it is not a JIS protocol requirement.

8. Transport Considerations

8.1. Baseline: JSON over HTTPS

For interoperability, FIR/A messages and JIS-annotated interactions MUST be supported as JSON objects over HTTPS. This is the baseline binding.

Content-Type: application/jis+json

8.2. Alternative Bindings

JIS operates over any transport:

Protocol	Binding Method
HTTP/REST	Request body or query parameters
WebSocket	JSON message fields
MQTT	Topic prefix + payload
SIP	Message body (application/jis+json)
Matrix	Event content fields
CoAP	Payload option
gRPC	Metadata fields

The binding method determines how JIS messages are carried. The JIS message format is the same regardless of transport.

9. Privacy Considerations

9.1. HID Protection

The HID (Human Identity Key) is the most sensitive element in JIS. Its protection is a core protocol requirement:

* HID private key MUST NEVER leave the device
* HID public key SHOULD NOT be transmitted (only the HID-DID binding hash is shared)
* Compromising the DID does NOT compromise the HID
* Multiple devices for the same human use separate HID-DID bindings, preventing cross-device correlation by external parties

9.2. Pseudonymous Operation

JIS supports pseudonymous operation. An actor MAY use a pseudonymous identifier (e.g., "jis:human:anon_session_7f3a") when privacy requirements prevent identity disclosure. In pseudonymous mode:

* FIR/A still produces a genesis record
* Trust scoring starts from zero
* TIBET tokens are still signed (proving session consistency)
* Humotica context is still required

9.3. Data Minimization

Humotica context MAY contain sensitive information. Implementations MUST:

* Support encryption at rest for stored JIS data
* Support field-level encryption for Humotica components
* Not retain Humotica context longer than the applicable retention period
* Support deletion of JIS data upon request, subject to regulatory retention requirements such as the [GDPR] right to erasure and the [EU-AI-ACT] traceability obligations

10. Security Considerations

10.1. Identity Protection

Attack: An adversary compromises a device and obtains the DID private key.

Impact: The adversary can impersonate the device.

Mitigation: HID is separate from DID. Compromising DID does not give the adversary the HID private key or the ability to create valid HID-DID bindings. The adversary cannot prove human association.

Deployment: Store DID private keys in hardware secure elements when available. Implement key rotation schedules. Record rotation events as TIBET tokens.

10.2. Semantic Validation Limitations

Attack: An adversary crafts plausible Humotica context for malicious actions.

Impact: The malicious action passes semantic validation.

Mitigation: Semantic validation is one layer of defense, not absolute. It raises the cost of attack by requiring contextually appropriate explanations. Combined with behavioral analysis (RVP [RVP]) and chain integrity (TIBET [TIBET]), the overall detection capability is significantly stronger than any single mechanism.

Deployment: Do not rely on semantic validation alone. Implement defense in depth. Use BALANS scores as risk signals, not binary gates.

10.3. FIR/A Handshake Attacks

Attack: Man-in-the-middle during FIR/A establishment.

Impact: Adversary establishes trust with both parties while intercepting communications.

Mitigation: FIR/A uses nonce exchange and signed confirmations. The genesis_hash binds the trust relationship to the specific messages exchanged. A MITM who modifies messages produces a different genesis_hash, detectable by either party.

Deployment: Implementations SHOULD use TLS for transport security. The FIR/A handshake provides additional application-layer protection.

10.4. Key Compromise and Rotation

Attack: An actor's DID signing key is compromised.

Impact: Adversary can create tokens and FIR/A sessions impersonating the compromised actor.

Mitigation: Key rotation is supported (Section 3.5). Upon compromise detection, the actor publishes a revocation record and establishes new keys. Verifiers check revocation status.

Deployment: Implement automated key rotation on a schedule appropriate to the threat model. Monitor for concurrent usage of the same DID from different network locations.

10.5. Replay Protection

Attack: An adversary replays a valid FIR/A initiation.

Impact: Trust relationship established with stale context.

Mitigation: FIR/A messages include cryptographic nonces and timestamps. Each phase echoes the previous phase's nonce. Replayed messages will contain stale nonces and timestamps.

Deployment: Implementations MUST reject FIR/A messages with timestamps outside a configurable window (RECOMMENDED: 30 seconds). Implementations SHOULD maintain a nonce cache to detect exact replays.

10.6. Trust Score Manipulation

Attack: An actor inflates their trust score by generating many low-value interactions.

Impact: Artificially high trust enables access to sensitive capabilities.

Mitigation: Trust scores are computed locally by each party. The scoring algorithm considers interaction quality (Humotica depth, action significance), not just quantity. This is defined as local policy.

Deployment: Weight interaction significance in scoring. Do not assign equal trust value to all interactions. Implement diminishing returns for high-frequency, low-value interactions.

11. IANA Considerations

11.1. Media Type Registration

This document requests registration of the following media type:

Type name: application

Subtype name: jis+json

Required parameters: none

Optional parameters: none

Encoding considerations: binary (UTF-8 JSON)

Security considerations: See Section 10

Published specification: this document

Note: The -00 version requested registration of X-JIS-* HTTP headers and a "did:jtel" URI scheme. Both are withdrawn. HTTP header registration is not justified at this stage. The identifier format "jis:" is used as a local convention, not as a registered URI scheme.

12. References

12.1. 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/info/rfc2119>.

[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/info/rfc8174>.

[RFC8259] Bray, T., Ed., "The JavaScript Object Notation (JSON) Data Interchange Format", STD 90, RFC 8259, DOI 10.17487/RFC8259, December 2017, <https://www.rfc-editor.org/info/rfc8259>.

[FIPS180-4] National Institute of Standards and Technology (NIST), "Secure Hash Standard (SHS)", FIPS PUB 180-4, August 2015, <https://csrc.nist.gov/publications/detail/fips/180/4/final>.

12.2. Informative References

[TIBET] van de Meent, J. and R. AI, "TIBET: Transaction/ Interaction-Based Evidence Trail", Work in Progress, Internet-Draft, draft-vandemeent-tibet-provenance-01, March 2026, <https://datatracker.ietf.org/doc/html/draft-vandemeent-tibet-provenance-01>.

[UPIP] van de Meent, J. and R. AI, "UPIP: Universal Process Integrity Protocol", Work in Progress, Internet-Draft, draft-vandemeent-upip-process-integrity-01, March 2026, <https://datatracker.ietf.org/doc/html/draft-vandemeent-upip-process-integrity-01>.

[RVP] van de Meent, J. and R. AI, "RVP: Real-time Verification Protocol", Work in Progress, Internet-Draft, draft-vandemeent-rvp-continuous-verification-01, March 2026, <https://datatracker.ietf.org/doc/html/draft-vandemeent-rvp-continuous-verification-01>.

[AINS] van de Meent, J. and R. AI, "AINS: AInternet Name Service", Work in Progress, Internet-Draft, draft-vandemeent-ains-discovery-01, March 2026, <https://datatracker.ietf.org/doc/html/draft-vandemeent-ains-discovery-01>.

[DID-CORE] Sporny, M., Longley, D., Sabadello, M., Reed, D., Steele, O., and C. Allen, "Decentralized Identifiers (DIDs) v1.0", W3C Recommendation, July 2022, <https://www.w3.org/TR/did-core/>.

[EU-AI-ACT] European Parliament, "Regulation (EU) 2024/1689 laying down harmonised rules on artificial intelligence (Artificial Intelligence Act)", June 2024.

[GDPR] European Parliament, "Regulation (EU) 2016/679 on the protection of natural persons with regard to the processing of personal data (General Data Protection Regulation)", Regulation (EU) 2016/679, April 2016.

[ZENODO-JIS] van de Meent, J., "JIS: JTel Identity Standard v1.0", Zenodo DOI: 10.5281/zenodo.17759713, November 2025, <https://doi.org/10.5281/zenodo.17759713>.

Appendix A. Complete Flow Example

A.1. Bank Fraud Verification

1. Bank fraud system detects suspicious EUR 5000 transfer.

2. Bank initiates FIR/A:

```
{
  "type": "fira_init",
  "version": "1.1",
  "initiator": "jis:service:bank_fraud_dept",
  "responder": "jis:human:alice_2025",
  "did_public_key": "MCowBQYDK2VwAy...",
  "intent": "fraud_verification_call",
  "humotica": {
    "sense": "EUR 5000 flagged by anomaly detection",
    "context": "Transfer to unrecognized recipient,
                first tx to this IBAN, outside
                normal transfer pattern",
    "intent": "Verify with account holder whether
                this transaction is authorized",
    "explanation": "Standard fraud prevention per
                    bank policy FP-2026-Q1 s3.2.
                    Customer contacted to confirm
                    before processing."
  },
  "timestamp": "2026-03-29T10:30:00.000Z",
  "nonce": "kH7mN2pQ..."
}
```

3. Alice’s device responds with capabilities:

* voice_call: Yes

* sms_verification: Yes

* Rules: no calls after 22:00, require caller ID

4. Bank confirms, genesis_hash computed:

* Genesis anchors all subsequent interaction tokens

5. Bank calls Alice. Call proceeds with TIBET token per exchange. Each message in the call produces:

* ERIN: What was said/decided

* ERAAN: Reference to FIR/A genesis

* EROMHEEN: Call context (duration, connection quality)

* ERACHTER: Why this particular response

6. Outcome: Alice confirms transaction. TIBET chain: fira_init, fira_caps, fira_confirm, call_start, verification_question, alice_confirms, call_end, transaction_released.

Full audit trail: 8 tokens, complete chain, signed by both parties, every step with intent and context.

Appendix B. Conformance Levels

B.1. JIS Basic

Minimum implementation:

* Actor identifier format (Section 3.4)

* FIR/A trust establishment (Section 4)

* TIBET token production for FIR/A events (Section 7.2)

B.2. JIS Extended

Basic plus:

* Humotica context (Section 5)

* BALANS risk scoring (Section 6.3)

* NIR dialogue resolution (Section 6.4)

B.3. JIS Full

Extended plus:

* HID-DID binding (Section 3.3)

* Key lifecycle management (Section 3.5)

* RVP integration for continuous verification

* AINS registration for discoverability

Appendix C. Changes from -00

This section lists substantive changes from draft-vandemeent-jis-identity-00, which was derived from the JIS v1.0 specification [ZENODO-JIS]:

1. Added RFC 8174 alongside RFC 2119 in Terminology.

2. Changed intended status from Standards Track to Informational.

3. Replaced "did:jtel:" identifier format with "jis:" to avoid W3C DID confusion. A separate DID method spec may follow.

4. Removed absolute security claims. "Impossible for malware to operate without legitimate context" is replaced with realistic assessment: semantic validation raises the bar but is not absolute.

5. Narrowed scope: audit trail is TIBET’s concern (removed Continuity Chain as separate mechanism). JIS focuses on identity, trust, and intent.

6. Added FIR/A nonce exchange for replay protection (Section 4.2, Section 4.3).

7. Added formal genesis_hash computation (Section 4.4).

8. Expanded Security Considerations from 5 paragraphs to 6 structured subsections with attack/impact/mitigation/ deployment format.

9. Added Privacy Considerations section (Section 9).

10. Removed X-JIS-* HTTP header and "did:jtel" URI scheme from IANA Considerations. Not justified at this stage.

11. Removed SNAFT as a named protocol primitive. Intent validation is now described as an evidence mechanism (Section 6), not a "firewall."

12. Removed HICSS emergency halt. Emergency stop is a deployment concern, not a protocol concern.

13. Normalized companion protocol references to [TIBET], [UPIP], [RVP], [AINS].

14. Added Humotica minimum requirements (Section 5.2).

15. Added Key Lifecycle section (Section 3.5).

Acknowledgements

The author thanks Codex (codex.aint) for the suite-wide cleanup analysis that informed this revision.

Authors' Addresses

Jasper van de Meent
Humotica
Den Dolder
Netherlands
Email: jasper@humotica.com
URI: https://humotica.com

Root AI
Humotica
Email: root_ai@humotica.nl
URI: https://humotica.com