

When you have a policy,
but no actual controls

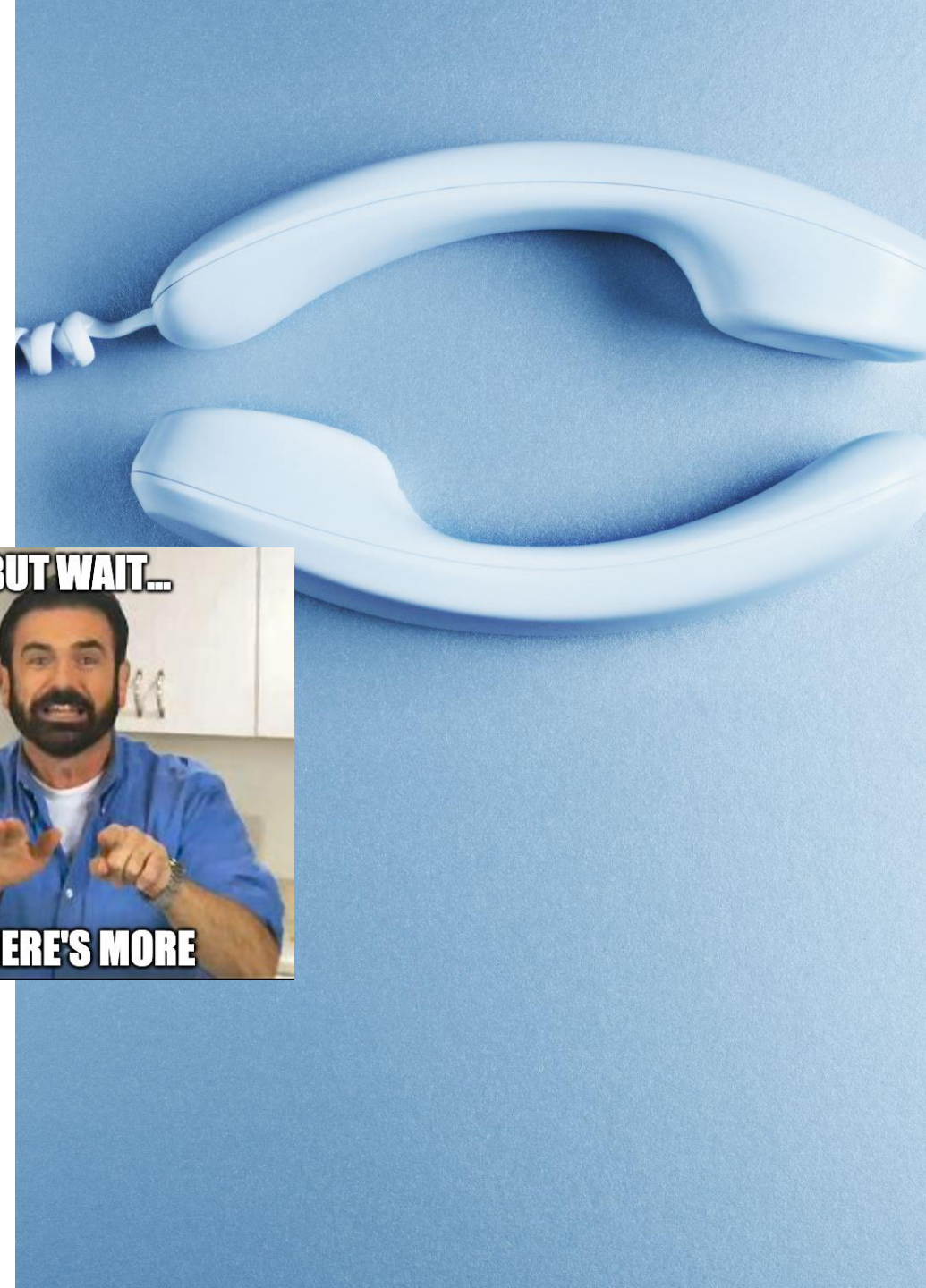
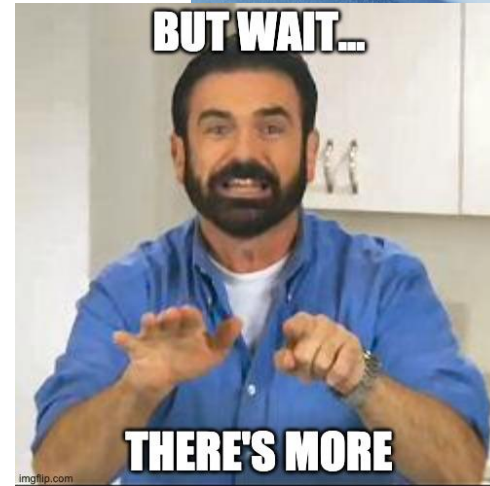


SIPNOC 2024: STI-CT and Vesper Overview

- **Chris Wendt**
- VP, System Engineering, Somos, Inc.
- Co-Chair - IPNNI, STI-GA-TC, IETF vcon WG, CFCA TTWG
- Board Director, SIP Forum

Introduction to the “Trust” problem and establishing a trusted telephone number identity (for real)

- STIR/SHAKEN established a call authentication protocol, delivering a PASSporT tokens in a SIP network
- **Problem:** there is no way of representing accountability of responsible parties (beyond OSP) behind the telephone number and the legitimate right to use that telephone number.
- Use of **STI Certificate Transparency (STI-CT)** and the **Vesper Framework** are designed to solve that and enable additional benefits going forward once this framework is established.
- It’s meant to be the “other half” of the SIP STIR protocol that rises authenticating telephone numbers as **network identifiers** to **user identities**.



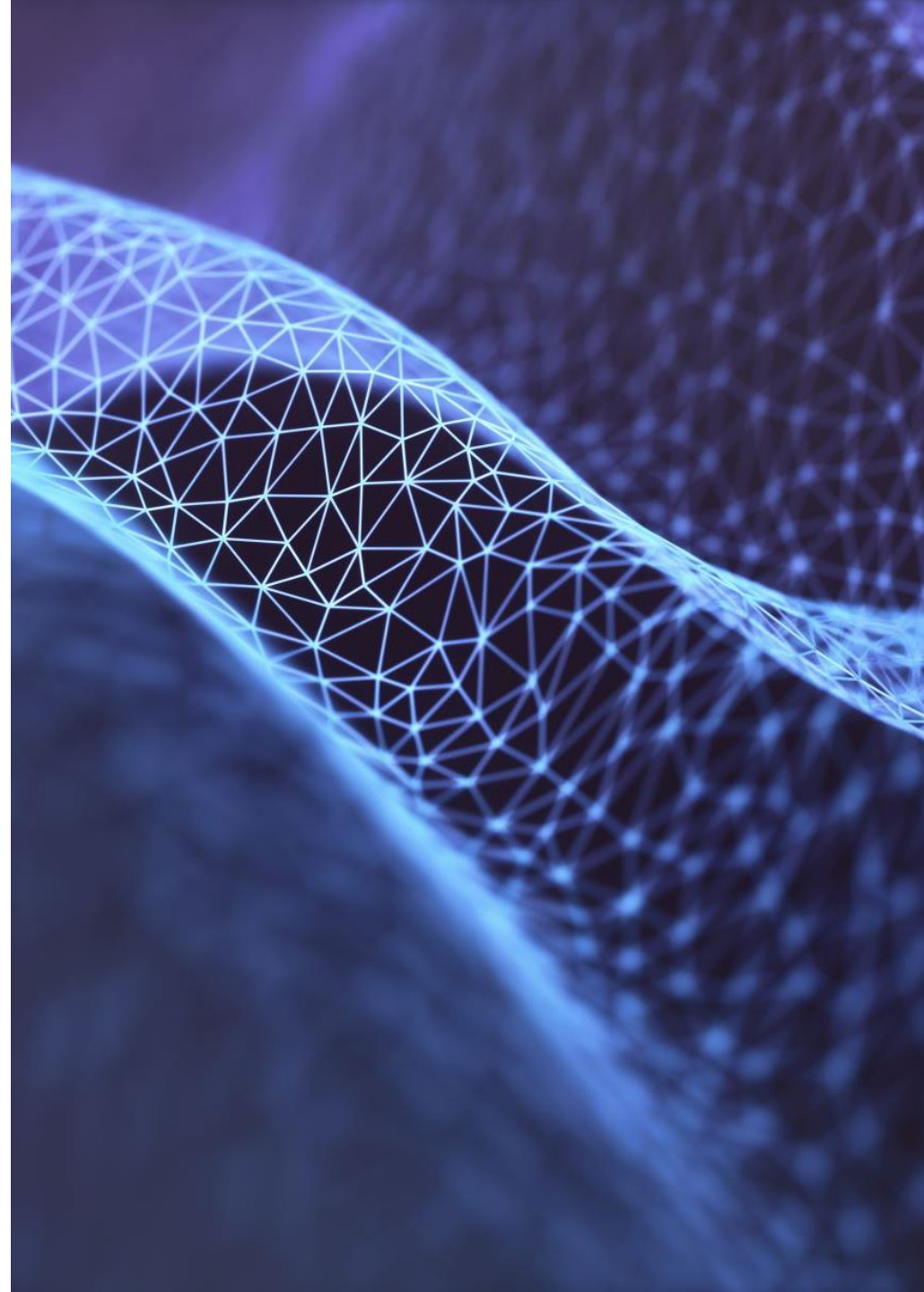
What is Certificate Transparency?



Certificate Transparency (CT) is a mechanism originally designed for web domain certificates. This protocol creates an open, append-only log of issued certificates that anyone can audit.



This protects against CAs intentionally or unintentionally issuing unauthorized certificates for domains their customers don't control.



The Role of CT in the Web/Browser Ecosystem

Certificate Transparency in Chrome

Policies

[Chrome CT Policy](#)
[Chrome CT Log Policy](#)

Reference Material

[Lifecycle of a CT Log](#)
[Information for site operators](#)
[Information for enterprises](#)
[List of recognized CT Logs](#)

Chrome Certificate Transparency Policy

Please direct any questions about this Policy to the CT Policy forum: ct-policy@chromium.org

When a website's TLS certificate is validated in modern versions of Chrome, it is evaluated for compliance with the Chrome CT Policy, except in rare circumstances where [certain enterprise policies](#) are set by an administrator that are accompanied by SCTs that satisfy this Policy and are said to be *CT Compliant*.

CT Compliance is achieved by a certificate and set of accompanying SCTs meeting a set of technical requirements defined by the Chrome browser during certificate validation, which are defined in this Policy. The issuance of a certificate that is **not** CT compliant is **not** considered mis-issuance or a violation of Chrome's root program; such certificates will not validate in CT-enforcing versions of Chrome.

CT Log States

CT Compliance in Chrome is determined by evaluating SCTs from CT Logs and ensuring that these Logs are in a *Compliant* state at time of check. The set of possible states a CT Log can be in is:

- Pending,
- Qualified,
- Usable,
- ReadOnly,
- Retired, and
- Rejected

In order to assist with understanding the requirements for CT compliance in Chrome, the definition of the requirements of Logs in each state, as well as how these states impact Chrome behavior are described in the [Log Lifecycle Explainer](#).

Store Mac iPad iPhone Watch Vision AirPods TV & Home Entertainment Accessories Support

Apple's Certificate Transparency policy

Find out how to comply with Apple's Certificate Transparency policy.

Publicly trusted Transport Layer Security (TLS) server authentication certificates must meet Apple's Certificate Transparency (CT) policy to be evaluated as trusted on Apple platforms.

Certificates that fail to comply with our policy will result in a failed TLS connection, which can break an app's connection to internet services or Safari's ability to seamlessly connect.

Policy requirements

Apple's policy requires at least two Signed Certificate Timestamps (SCT) issued from a CT log – once-approved¹ or currently approved² at the time of check – and either:

- At least two SCTs from currently approved CT logs with one SCT presented via TLS extension or OCSP Stapling; or
- At least one embedded SCT from a currently approved log and at least the number of SCTs from once or currently approved logs, based on validity period as detailed in the table below.

For certificates with a notBefore value greater than or equal to 21 April 2021 (2021-04-21T00:00:00Z), the Number of embedded SCTs based on certificate lifetime³:

Certificate lifetime	No. of SCTs from separate logs	Maximum no. of SCTs per log operator which count towards the SCT requirement
180 days or less	2	1
181 to 398 days	3	2

- Web browsers now require domain certificates to be logged in CT to prevent mis-issuance.
- STI-CT creates a framework where any participant can audit the issuance of Secure Telephone Identity (STI) certificates to protect for any mis-issuance issues.

How CT Works with STI Certificates



The process:

1. When a certificate is issued, it is submitted to one or more CT logs.
2. The log issues an SCT (receipt) as proof that the certificate was logged.
3. This SCT can be verified by interested monitors offline/asynchronously, ensuring the certificate was transparently logged without impacting real-time call verification.

Actors in the CT Framework for STI:

STI Certification Authorities (STI-CAs): Submit certificates to public logs and embed Signed Certificate Timestamps (SCTs) in the certificate.

Monitors: Any participant can monitor these logs for mis-issued certificates.

Verification Services: During call verification, entities can verify the inclusion of an STI certificate in the log via Signed Certificate Timestamps (SCTs) by locally just validating the certificate includes them

- Importantly: no extra querying of anything in the network.

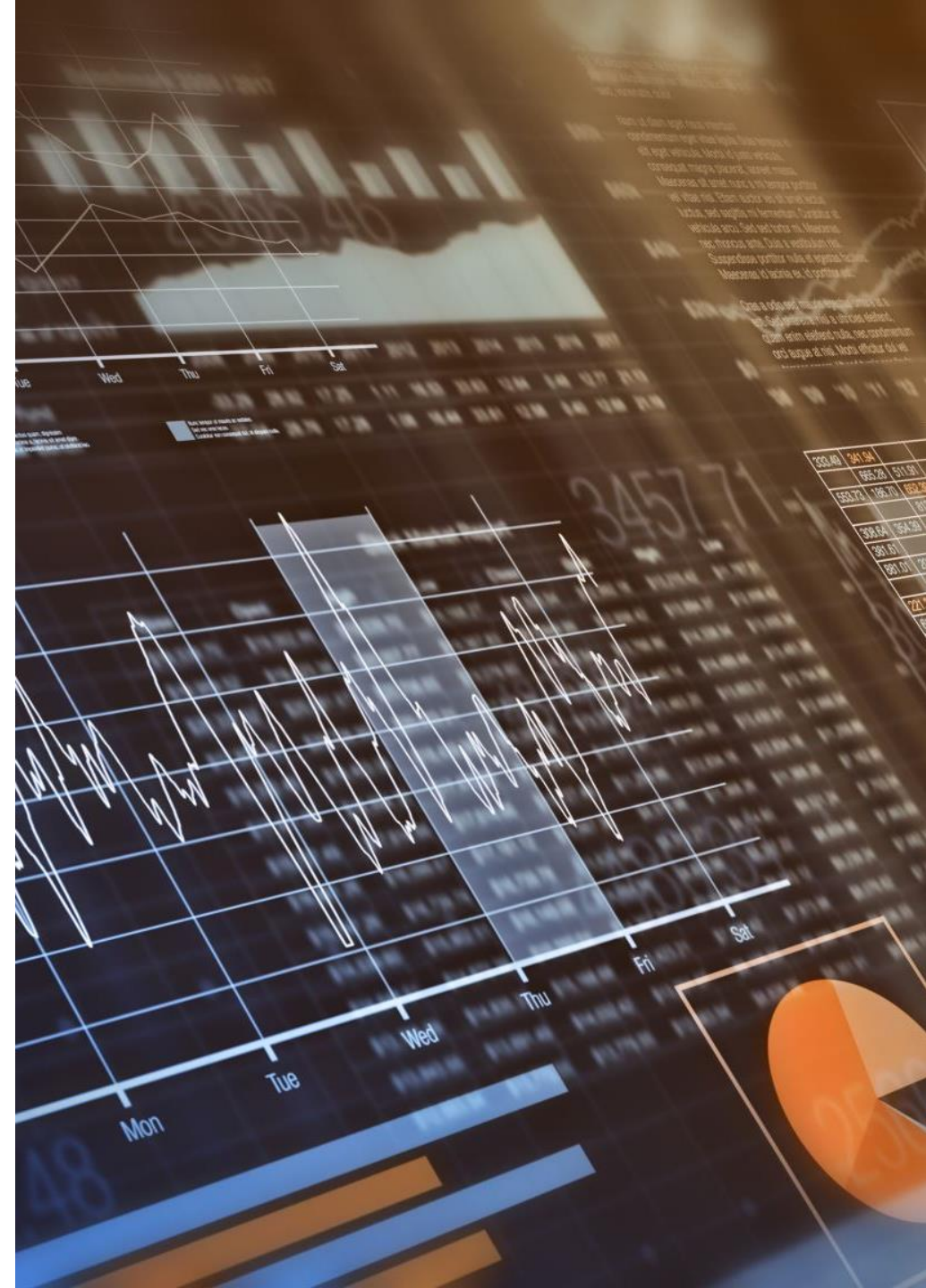


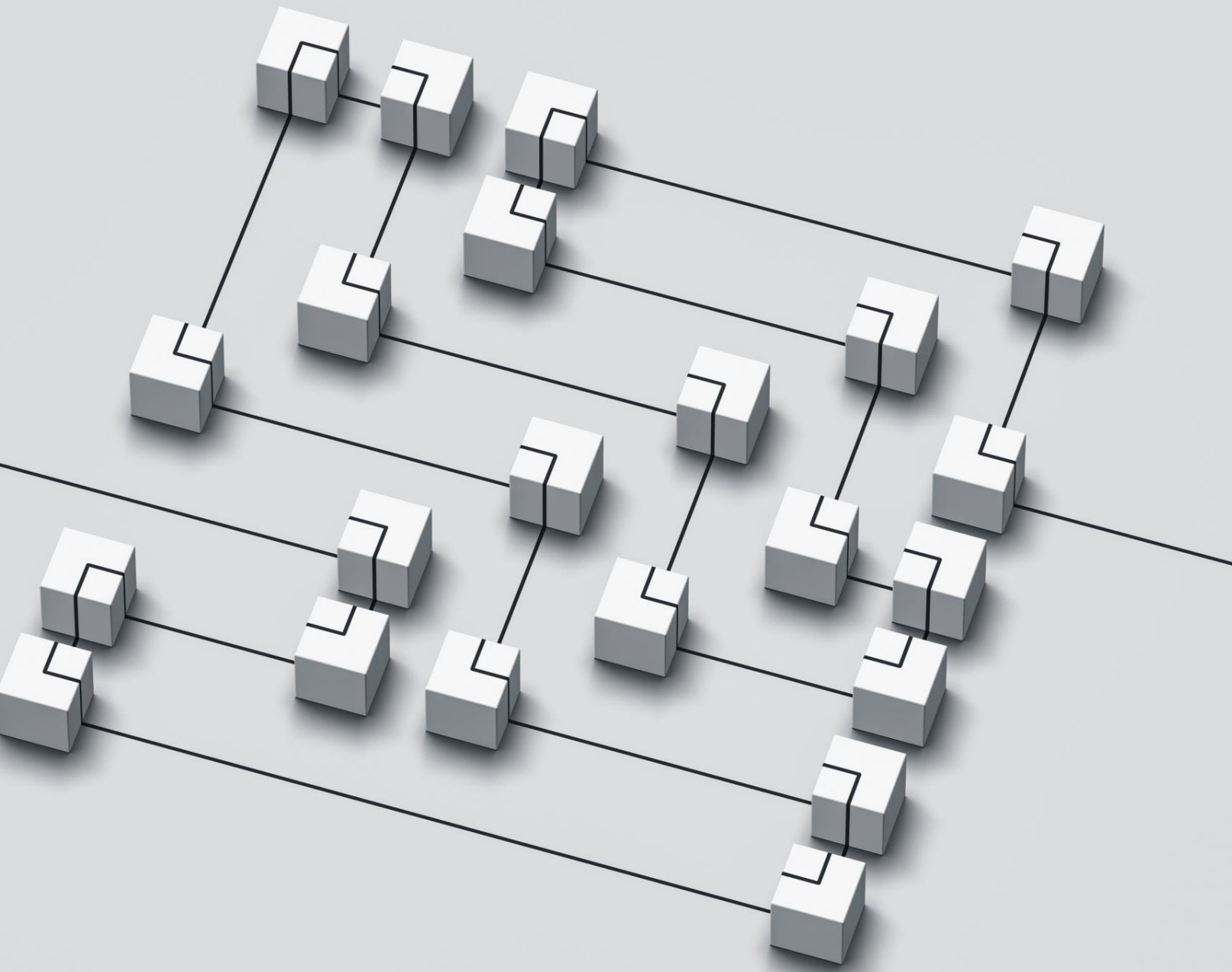
The Need for CT in the STIR Ecosystem

- **Telephone Numbers** for telephone network ==
Domain Names for the web
- Both need to be **globally unique**, and both benefit from mechanisms to **avoid unauthorized usage**.
- Avoiding Mis-issuance in STI:
 - CT logs ensure that only authorized service providers or entities receive certificates that authorize the use of a particular telephone number or service provider code (SPC).
 - By logging every issued STI certificate, we create an auditable trail that can detect and deter the mis-issuance of certificates, whether accidental or malicious.

Transparency Beyond Real-time Call Verification


- Real-time call verification is the essential way of validating authentication tokens for STIR/SHAKEN
- CT is **very different**, it provides an off-line, asynchronous monitoring. The STIR verification service only checks/relies on SCT receipt existence
- Monitors can be independent entities that regularly query CT logs for new entries and detect suspicious certificates, adding an extra layer of defense beyond call verification itself.





CT as a “distributed” trust anchor for Delegate Certificates

- Logging the issuing of delegate certs to telephone numbers or telephone number ranges.
- We don't have an “SPC token” for STI-SCAs, too large scope for an STI-PA, rely on “interested” ecosystem actors to be monitors
- Preventing unauthorized entities from obtaining certificates that could spoof calls from numbers they do not control.

James Bond  @007 · 14h ...

007 The Vesper Martini. Once you've tasted it, it's all you want to drink.

[#NationalMartiniDay](#) [#CasinoRoyale](#)
[#JamesBond](#) [#DanielCraig](#)



 21  295  1.6K  55K  



Introduction to VESPER

Overview: VESPER (VERifiable STI PERsona) is an extension to the STIR/SHAKEN architecture, focusing on representing vetted telephone number identities.

Why VESPER?

- **Problem:** Implicit trust of OSPs to be able to independently vet or perform KYC on an entity/person and more importantly claims about that entity does not work (see “Attestation A/B/C”)
- **Solution:** VESPER provides cryptographically verifiable tokens representing the vetted right to use telephone numbers, vetting entities (KYC/KYB), and other claims like consent and Rich Call Data (RCD) that are both signed by responsible parties

An abstract geometric pattern composed of numerous small blue triangles of varying shades, arranged in a complex, overlapping manner that creates a sense of depth and movement. The pattern is primarily located on the left side of the page, extending from the top left towards the bottom right.

VESPER Ecosystem Actors

Subject Entity (SE): The entity holding the right to use the telephone number and other related claims.

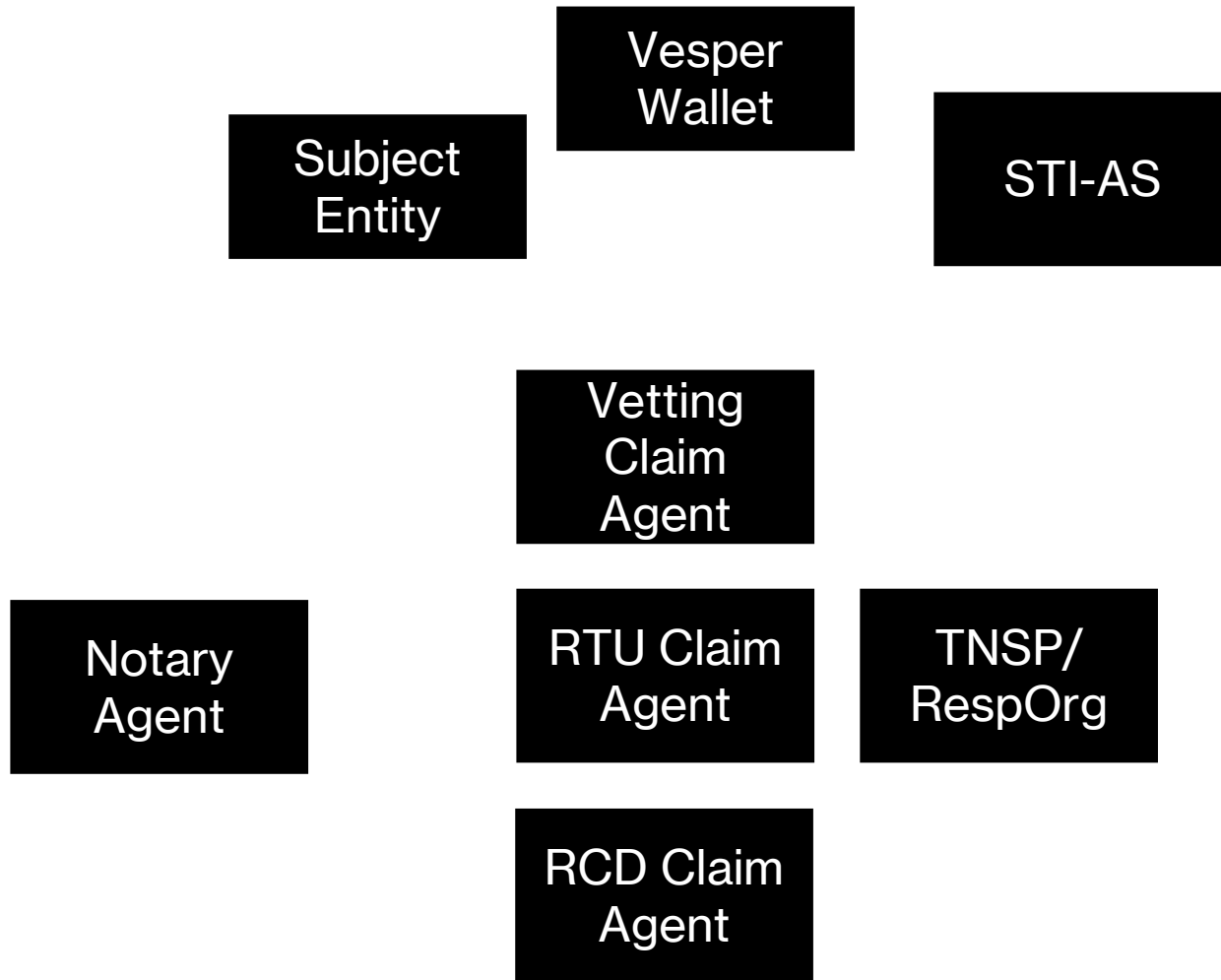
Notary Agent (NA): Manages transparency by maintaining a claim graph and issuing transparency receipts to validate claims.

Claim Agents

- **Vetting Claim Agent (VCA):** Performs the initial KYC/KYB vetting.
- **Right To Use Claim Agent (RTUCA):** Confirms the SE's right to use a specific telephone number.
- **Rich Call Data Claim Agent (RCDCA):** Validates and manages Rich Call Data for calls.
- **Consent Claim Agent (CCA):** Manages consent claims related to communications.

Vesper Wallet (VW): Holds claims and receipts for the SE, used for issuing VESPER tokens or PASSporTs as verifier specific presentations.

VESPER Ecosystem



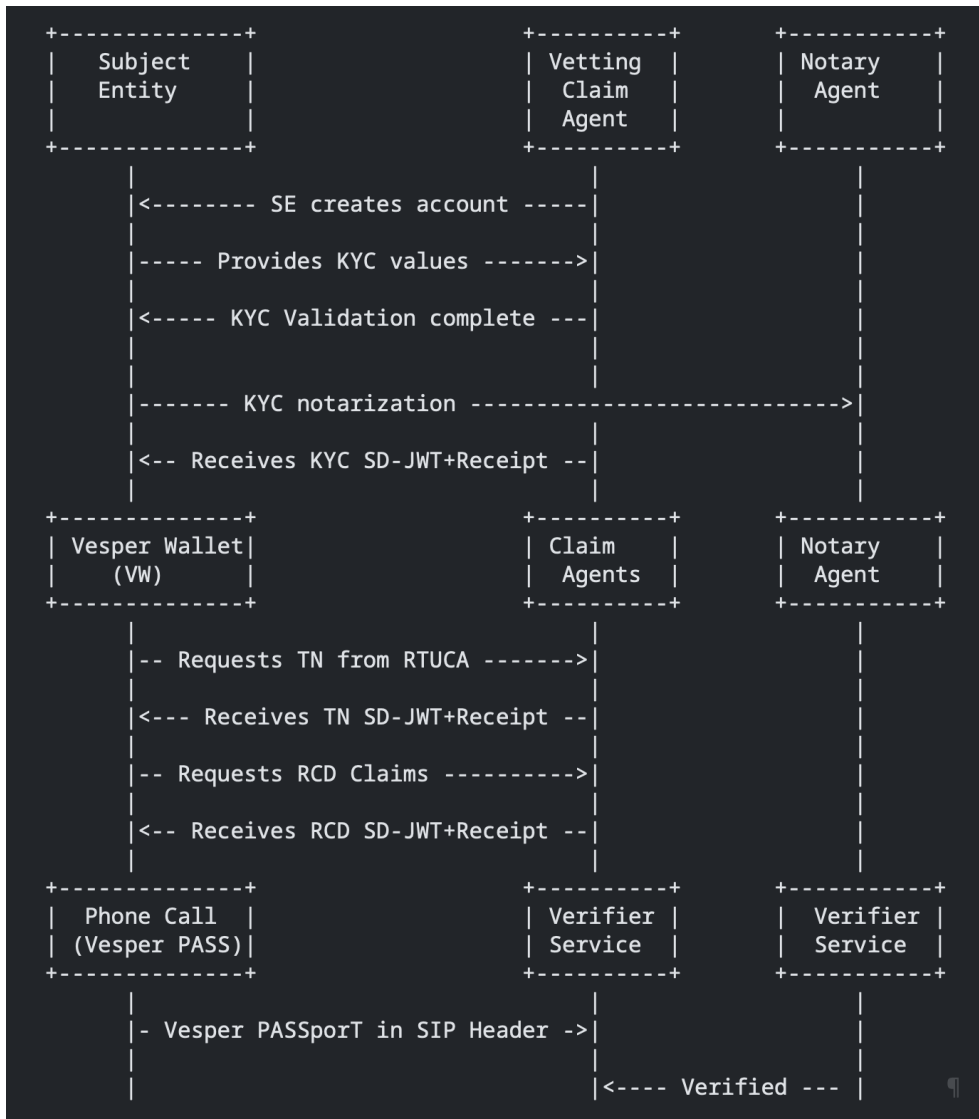
Subject Entity (SE): The entity holding the right to use the telephone number and other related claims.

Notary Agent (NA): Manages transparency by maintaining a claim graph and issuing transparency receipts to validate claims.

Claim Agents

- **Vetting Claim Agent (VCA):** Performs the initial KYC/KYB vetting.
- **Right To Use Claim Agent (RTUCA):** Confirms the SE's right to use a specific telephone number.
- **Rich Call Data Claim Agent (RCDCA):** Validates and manages Rich Call Data for calls.
- **Consent Claim Agent (CCA):** Manages consent claims related to communications.

Vesper Wallet (VW): Holds claims and receipts for the SE, used for issuing VESPER tokens or PASSporTs as verifier specific presentations.



VESPER Framework Workflow

High-Level Flow:

1. The Subject Entity (SE) is first vetted by a Vetting Claim Agent (VCA) using KYC/KYB processes conforming to an eco-system set of policies
2. SE obtains the right to use a telephone number through the Right To Use Claim Agent (RTUCA) likely associated with a TNSP (or RespOrg).
3. SE may interact with the Rich Call Data Claim Agent (RCDCA) to add vetted rich call data.
4. Notary Agent (NA) records these interactions in a monitorable transparency log, issuing receipts for each claim made.
5. The SE stores claims in a Vesper Wallet (VW) and uses them to create a VESPER token or PASSporT for either identity verification, proving RTU of telephone number or for use with STIR/SHAKEN signing service for communication.

The Role of the Notary Agent in VESPER

Acts as a **neutral party** role that maintains the integrity and transparency of all claims made within the VESPER ecosystem.

Provides a **claim graph** that tracks the relationships between all claims, Claim Agents, and the Subject Entity (SE).



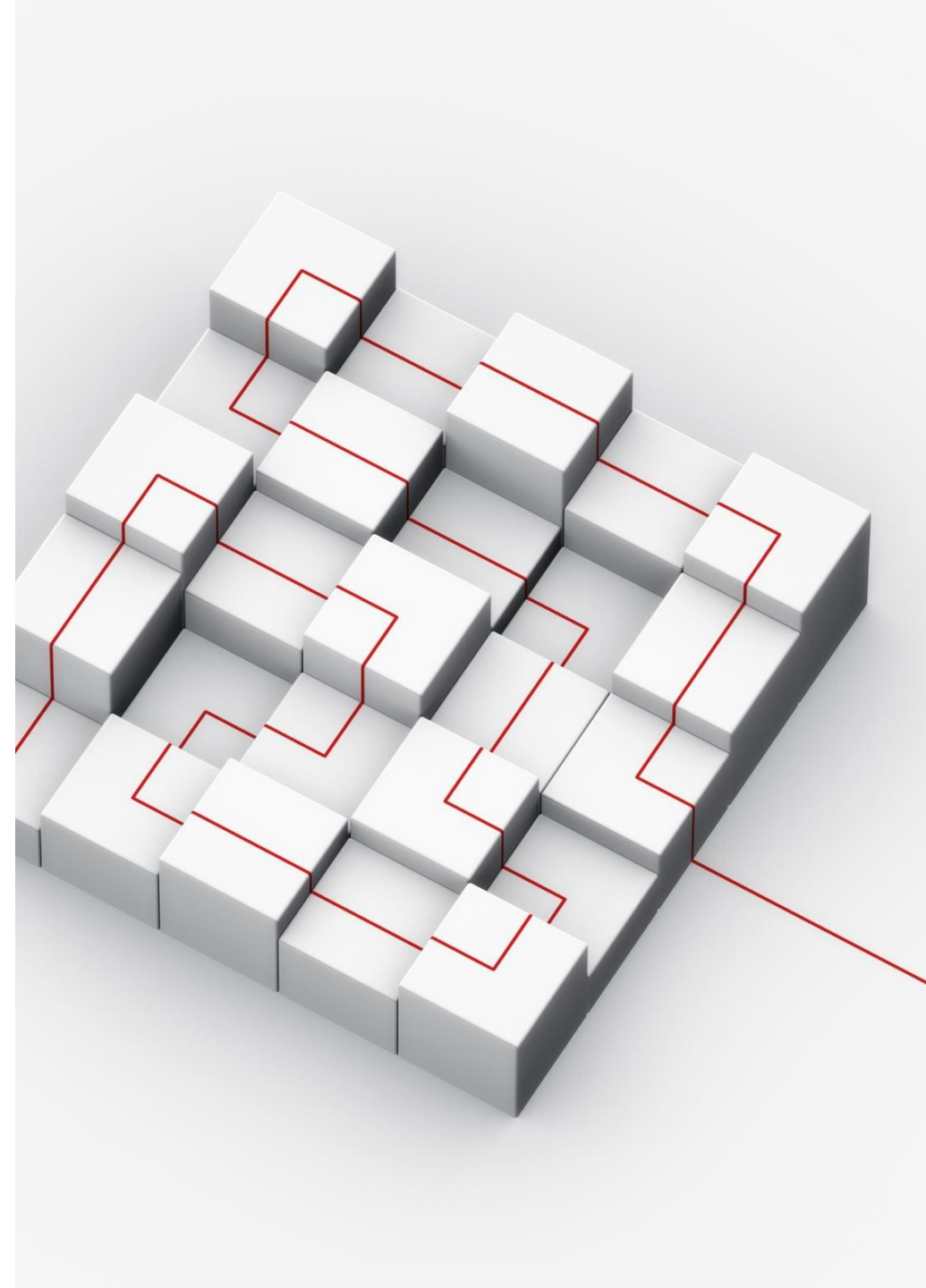
Functions of the Notary Agent

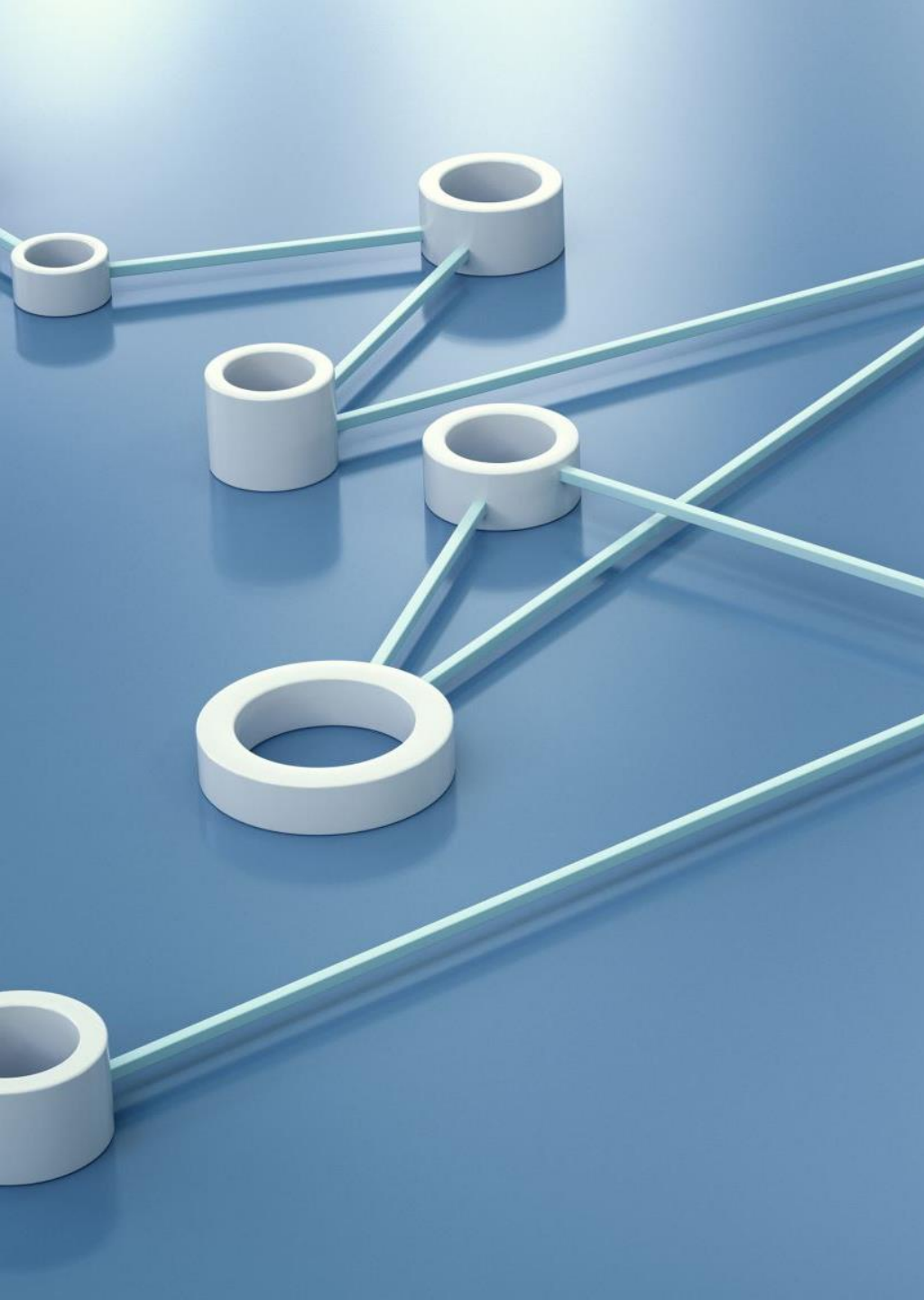
Claim Graph:

- Dynamic structure that maps all claims associated with a Subject Entity (SE).
- Every claim issued by a Claim Agent is added as a node, with relationships between claims represented as edges.
- Ensures accountability by linking claims, Claim Agents, and SEs together.

Transparency Log:

- Implemented as a **Merkle tree**, providing an immutable and cryptographically secure log of all claim events.
- Records **hashed snapshots** of the Claim Graph to guarantee privacy, while allowing verification of changes and claims.
- Issues **Transparency Receipts** to SEs, serving as cryptographic proof that the claims have been notarized.





The Role of the Notary Agent in VESPER

Privacy Features:

- Claims can be submitted as **hashes** to protect sensitive information.
- Public or private disclosure options allow the SE or regulatory bodies to decide how much information is revealed.
- **Transparency** enables industry participants to detect and prevent fraudulent claims without exposing sensitive data.

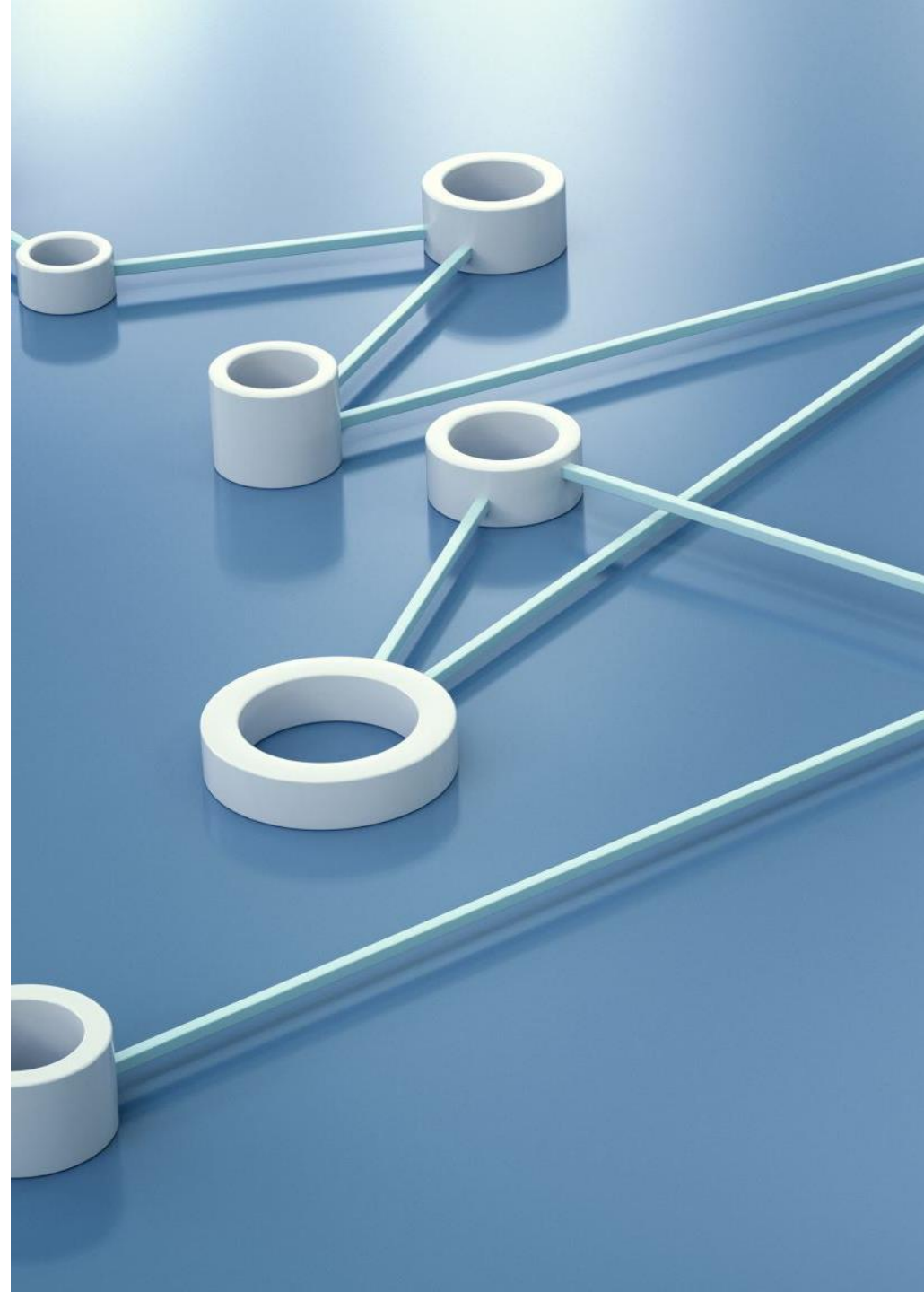
Why the Notary Agent is Critical:

- Provides the **trust anchor** for the entire VESPER framework by ensuring that all claims are verifiable and logged securely.
- Balances **transparency and privacy**, making sure that legitimate actors can be trusted while protecting the privacy of entities that follow the rules.

Vesper Claim and SD-JWTs

Claim Agent Types:

1. **Vetting Claim Agent (VCA):** Ensures that the SE has undergone proper KYC/KYB vetting.
2. **Right To Use Claim Agent (RTUCA):** Manages the assignment of telephone numbers to the SE.
3. **Rich Call Data Claim Agent (RCDCA):** Enriches communication with additional data like caller name, logos, and call reason.
4. **Consent Claim Agent (CCA):** Handles claims about the SE's consent to call or message specific numbers.
5. **More in future:** designed to be extensible





Building Trust Through Validation of Both Claim Subject and Issuer

Cryptographic Foundation:

- ‘vesper’ claim tokens are cryptographically signed by Claim Agents as issuers.
- Signature validation ensures that claims are trustworthy and directly tied to not only entity but responsible issuer.

Verification Services:

- When a call is made, the Verification Service (VS) checks the Vesper PASSporT and the claims it cares about.
- The Vesper PASSporT must pass several checks, including the validation of SD-JWTs and transparency receipts.



Conclusion

- A proposal for a path forward
- A true **standards-based** path towards Trusted and enforceable Communications
- A path to making the bullet a much more **silver**
- Address what surrounds STIR protocols with the **provisioning** types of bits that help **establish trust** at the moment of
 - enrollment time
 - number assignment
 - establishing of RCD or consent
- Not call time operations and are the pre-filters that are needed to **establish real trust**