Challenges of Remote Attestation for Confidential Computing (CC) Workloads

Yogesh Deshpande

Principal Engineer – Arm

Yogesh.Deshpande@arm.com

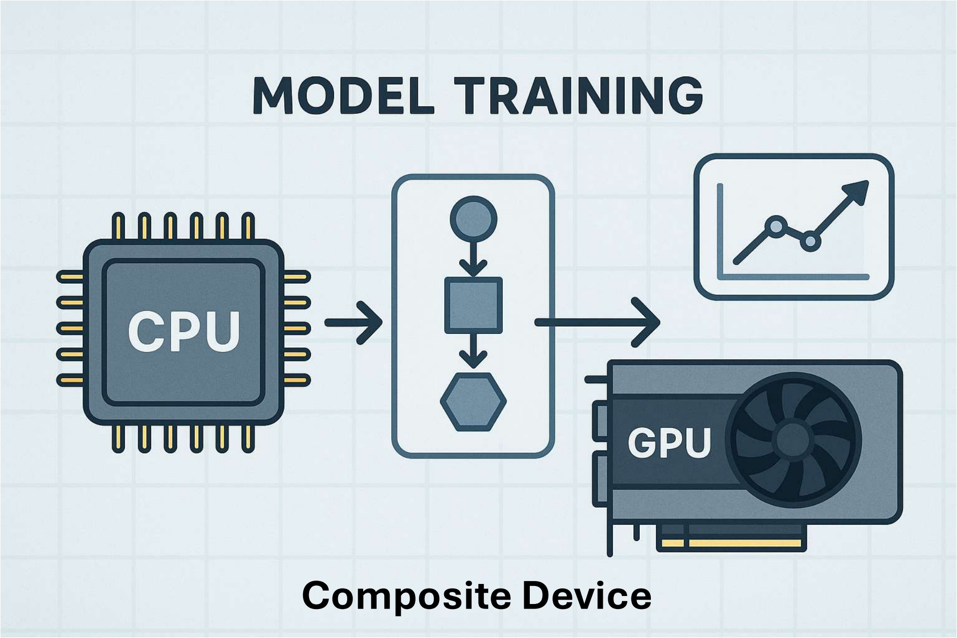
LinkedIn: https://www.linkedin.com/in/yogesh-deshpande-1454b71/

Date: 01st February 2026

FOSDEM 2026

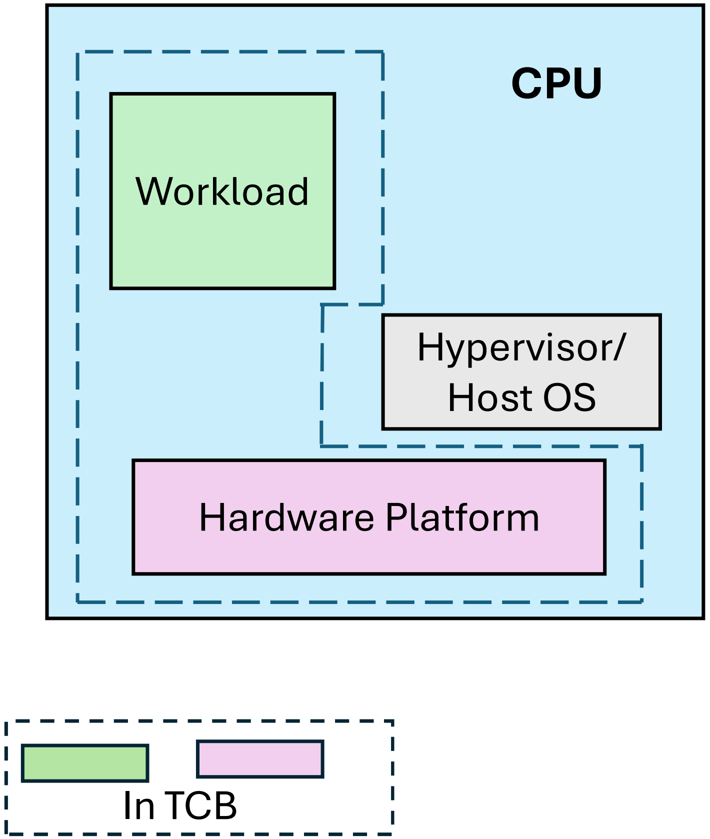
Challenges of CC Remote Attestation

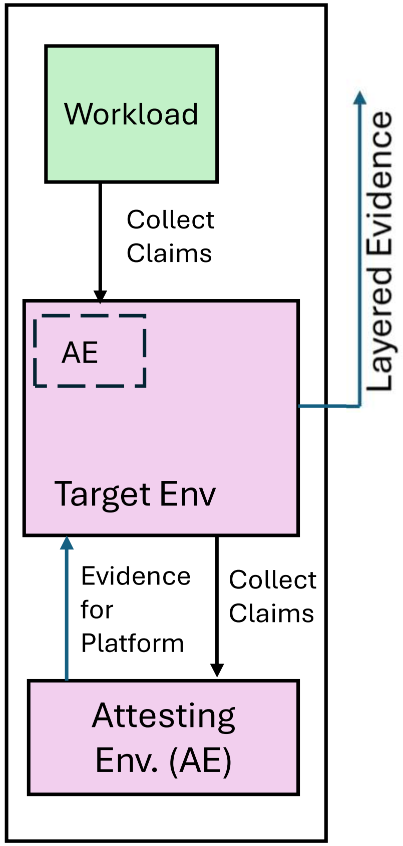
Confidential Computing Model Training Use Case



- A CC system is often a complex system, comprising multiple Root of Trusts, CPU and one or more GPUs
- Effectively an Attester is in fact a Collection of Attesters – **Composite Attester**
- Every individual component has its own remote attestation
- One needs to assess the trustworthiness of the entire composition – prior to making trust based decisions

Challenges of CC Remote Attestation





- A Workload runs on a CPU in a Confidential Computing Environment
- Workload runs on a Hardware Platform
- Workload and Hardware Platform comes from a different parts of supply chain
- One needs to assess the trustworthiness of the entire composition – prior to making trust decisions
- CPU itself is a Layered Attester- RFC 9334

The Confidential Computing (CC) Use Case is a Composite Attestation Use Case

What is the impact ??

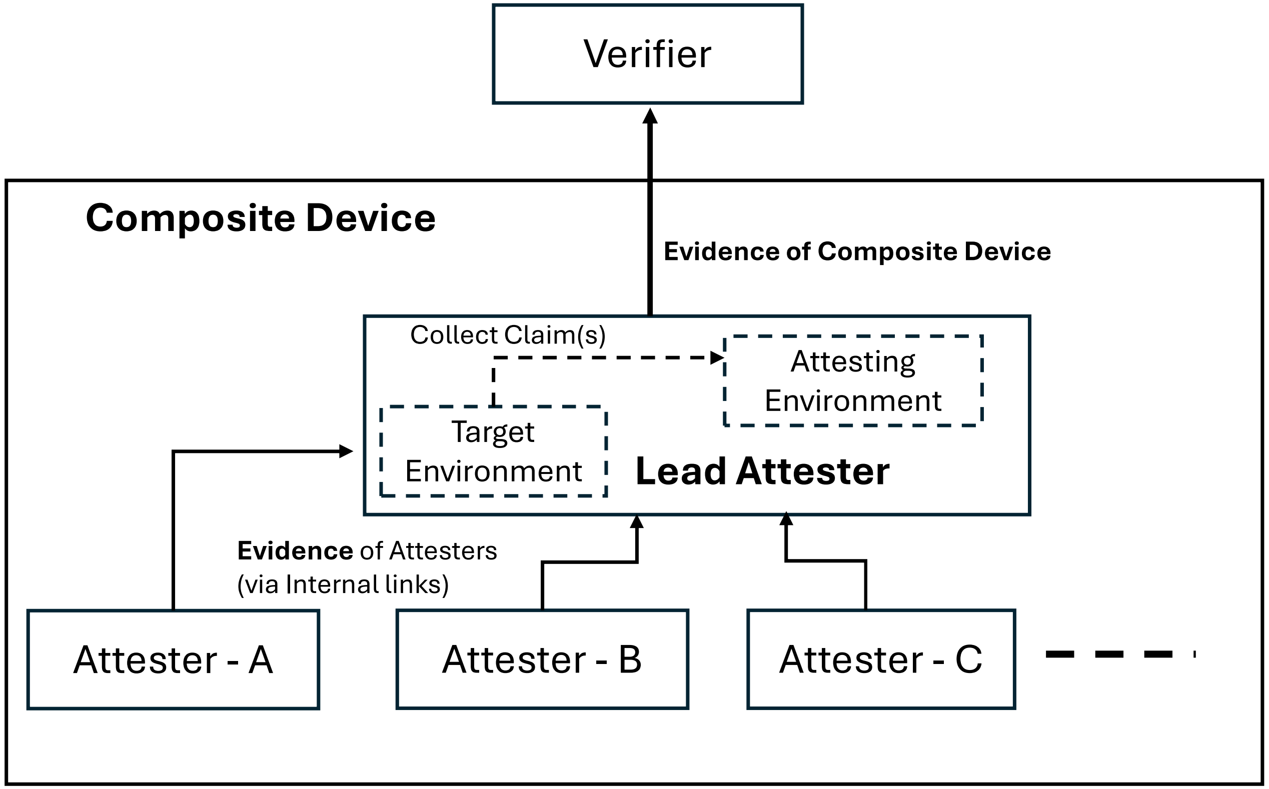
Composite Attestation story poses many questions

- How does one construct Attestation Evidence of a Composite Attester ?
- How does Supply Chain Endorsements for individual components, be linked to provide a single view of a Composite Attester to a Verifier ?
- What is the impact on Attestation Verifiers ?
- Can a single Verifier perform Appraisal of such an Attester ? If not,
    - How does the Attestation Results from component Verifiers be joined to form a consistent view of Device Trustworthiness to the Relying Party?

Composite Attester Evidence - Challenges

- Multiple component evidence combined to form a Composite Evidence
- The format and nature of claims in each component Evidence is different
- The Composite Evidence, MAY not have a single authority responsible for complete Evidence.
- How does one bind the individual component Evidence to protect the integrity of the collection ?
- How can one establish a specific component Evidence (example Workload) can be allowed/not allowed to be combined with a specific Platform Evidence ?
- Who owns the Appraisal policy?

Composite Evidence Ideas



- Perhaps we need a **Lead Attester - LA**
- LA collects Evidence from individual Attesters
- LA needs to specify the composition semantics
- LA communicates externally to a Verifier

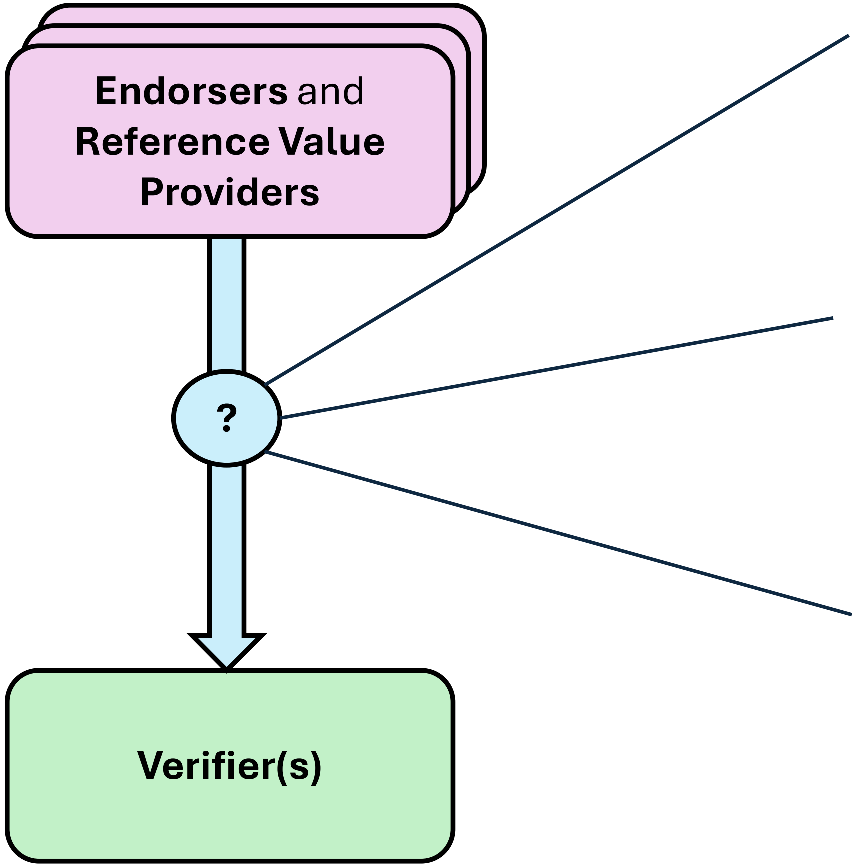
Evidence Standards and OSS Veraison Implementation

IETF Standard	Purpose
Concise Message Wrapper (CMW)	Acts as an Envelope to transport RATS Messages securely. Can be effectively used to collect multiple Component Evidence to form a payload for Composite Evidence
Entity Attestation Token (EAT)	EAT SubMods can be used to represent collection of Evidence
Taxonomy of Composite Attesters Work In Progress	Clarifies and extends the meaning of Composite Attester. Documents various class of Composite Attesters
EAT Profile for Device Attestation Work In Progress	An EAT Profile which provides a standardized Evidence format for Device Assignment when Devices such as GPU, network adapter etc. are assigned to a Confidential VM

Veraison Library Name	Purpose	Location
Entity Attestation Token	Library to build Attestation Evidence	github.com/**veraison**/eat
CMW	Library to build Concise Message Wrapper	github.com/**veraison**/cmw
RATS Deamon - RATSd	An Attester daemon to collect Composite Evidence from multiple independent RoTs	github.com/**veraison**/ratsd
EAT based DA (WIP)	Library for Device Attestation Token	github.com/**veraison**/da

VERAISON

Endorsement Distribution - Challenges



Fragmentation

- ➢ Vendor-specific APIs and lookup mechanisms
- ➢ Multiple data formats

Complexity

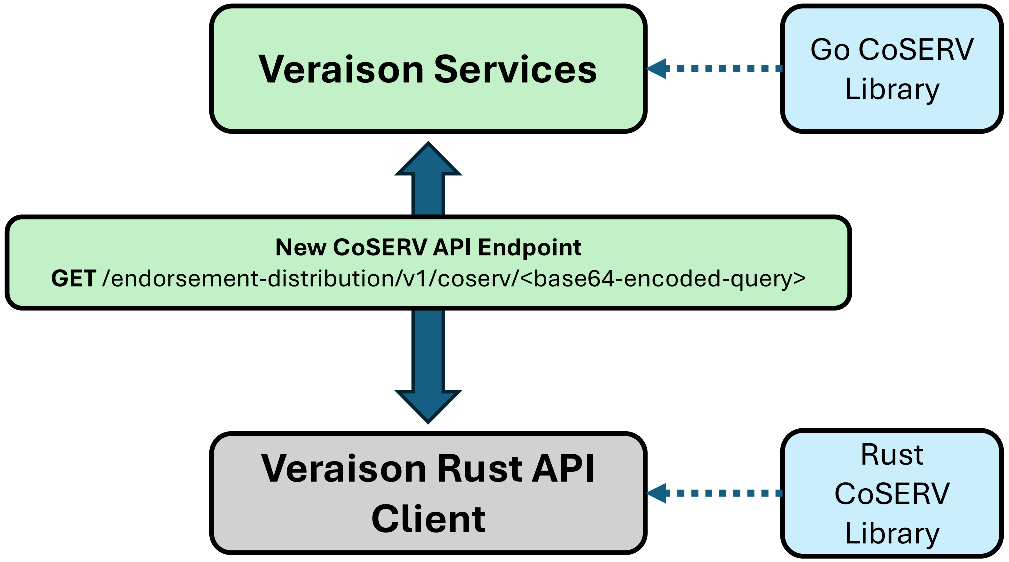
- ➢ Many supply-chain providers across hardware, firmware and software
- ➢ Need to map attester evidence to relevant artifacts

Scalability

- ➢ Need for flexible conveyance mechanisms: push, pull, pub-sub
- ➢ Caches are essential for performance

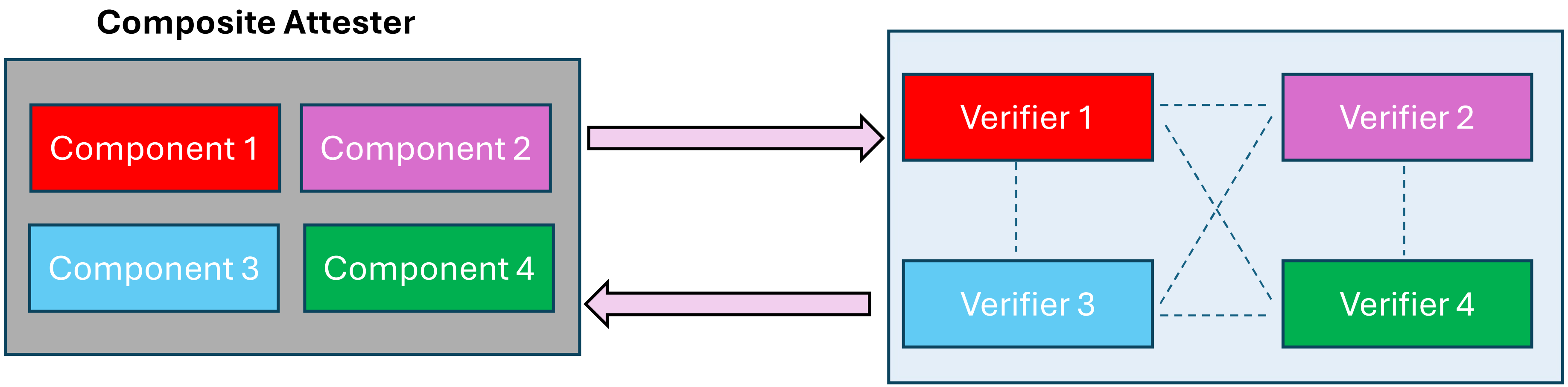
Endorsement Distribution – Standards and Open-Source

- Concise Selector for Endorsements and Reference Values (CoSERV)
- Adopted item of RATS WG
- Founded on existing CoRIM data model
- A common query/result data format for the industry, specialized for endorsement artifacts
- Transport-agnostic
- Cache-friendly HTTP bindings
- Flexible conveyance options
  - Bundle one or more CoRIM files from source providers
  - Smart aggregation into verifier-friendly packages
  - Support for other formats via CMW wrapping


Veraison Services
Go CoSERV Library
New CoSERV API Endpoint
GET /endorsement-distribution/v1/coserv/<base64-encoded-query>
Veraison Rust API Client
Rust CoSERV Library

Veraison service can act as **endorsement distributor** with support for CoSERV API endpoint – PoC and demo coming soon!

Verification of Composite Attesters



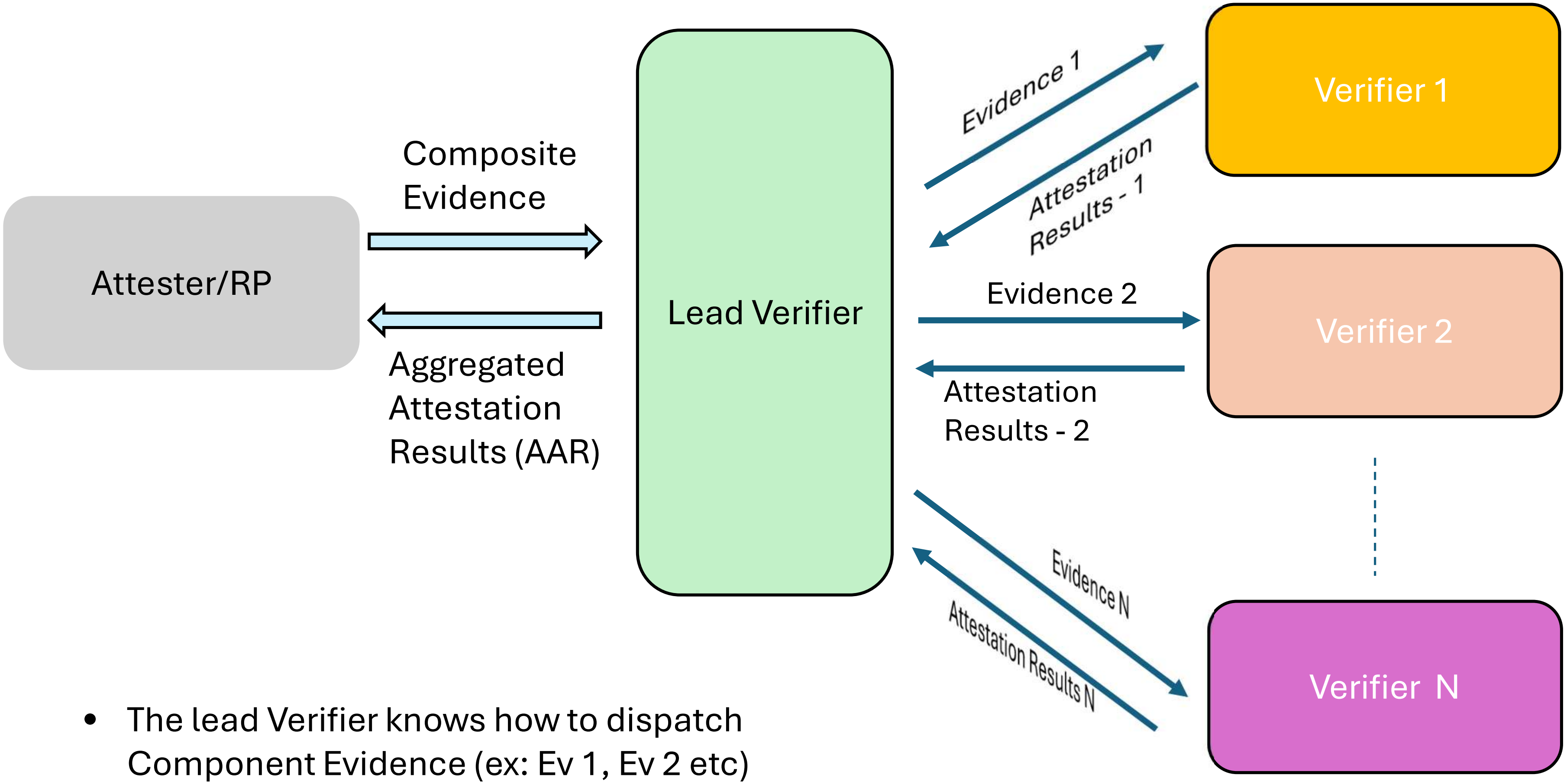
Need for Multiple Verifiers

- Verifiers from different Vendors
- Verifiers with different trust model & capability
- Different policies for Verification

Need for Multiple Verifiers

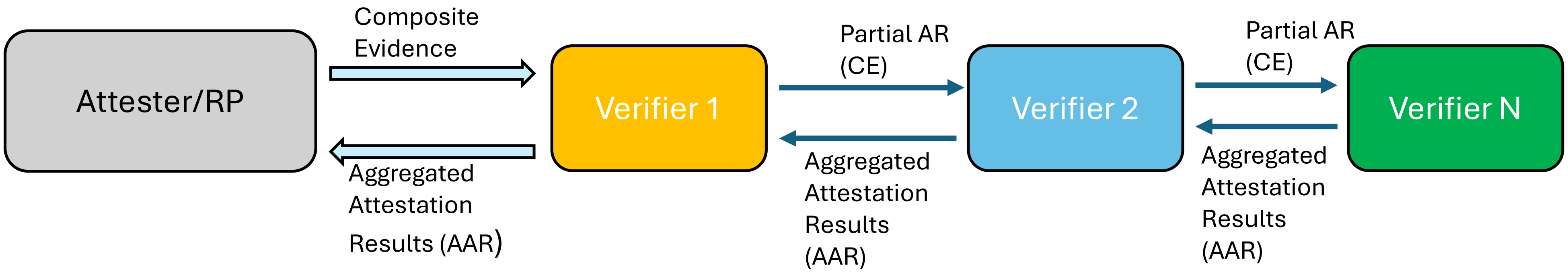
- There may not exist a single Entity to stand up to build all component Verification, due to
  - ➢ Lack of knowledge
  - ➢ Complexity
  - ➢ Cost concerns

Hierarchical Model



- The lead Verifier knows how to dispatch Component Evidence (ex: Ev 1, Ev 2 etc) to the suitable Verifier

Cascade Model



Attester/RP

Composite Evidence

Aggregated Attestation Results (AAR)

Verifier 1

Partial AR (CE)

Aggregated Attestation Results (AAR)

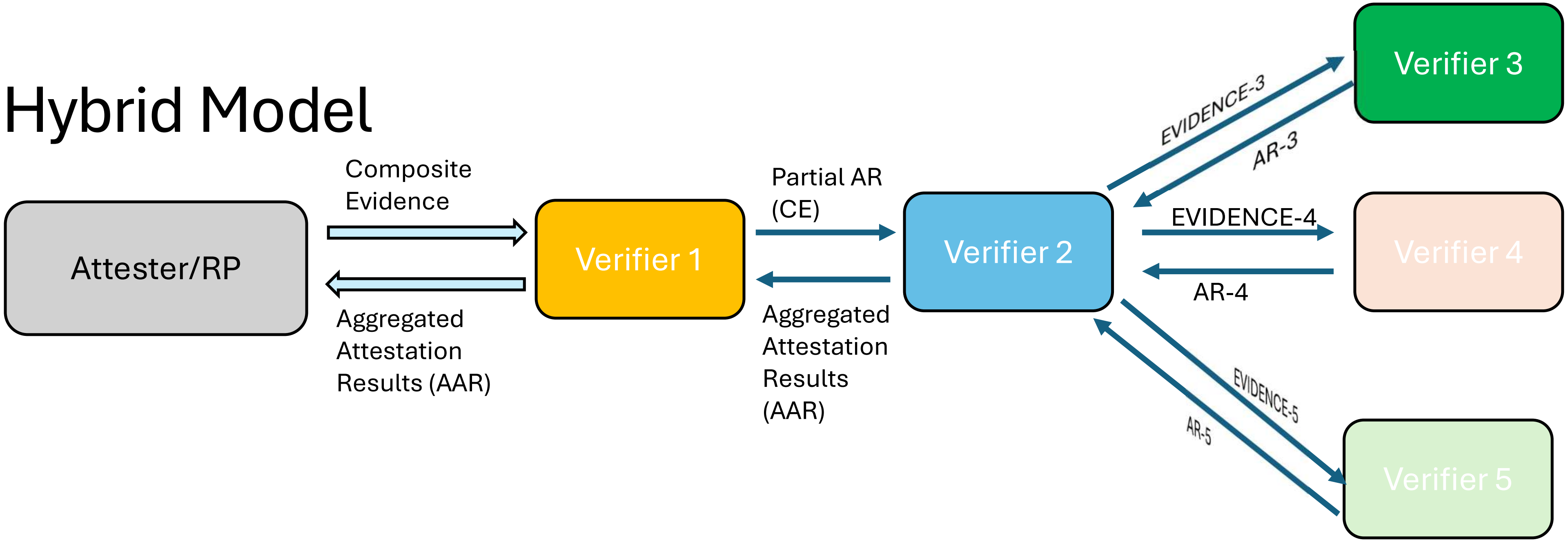
Verifier 2

Partial AR (CE)

Aggregated Attestation Results (AAR)

Verifier N

Hybrid Model



Attester/RP

Composite Evidence

Aggregated Attestation Results (AAR)

Verifier 1

Partial AR (CE)

Aggregated Attestation Results (AAR)

Verifier 2

EVIDENCE-3

AR-3

Verifier 3

EVIDENCE-4

AR-4

Verifier 4

EVIDENCE-5

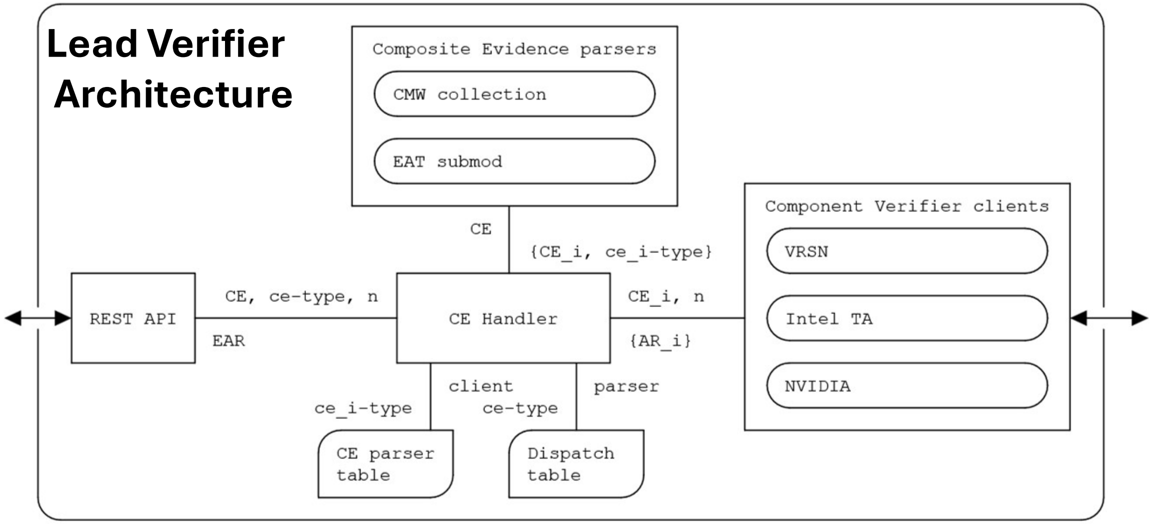
AR-5

Verifier 5

Multiple Verifier – Standard and Open-Source Work

- **IETF RATS Remote Attestation with Multiple Verifiers**
  - https://datatracker.ietf.org/doc/draft-deshpande-rats-multi-verifier/
  - Draft yet to be adopted in IETF RATS WG

- **Lead Verifier Implementation – Project Veraison**



Lead Verifier Veraison Project Board

https://github.com/orgs/veraison/projects/17

Attestation Results - Challenges

When Evidence is Composite, the Attestation Results may need to be Composite

When Attestation Results are Composite:

- Attestation Results MUST express the Composition Semantics
- Ease and simplicity of format, for the Relying Party
- Heterogeneity of Attestation Results coming from component-specific Verifiers
- How is the trust model reflected in the Combined Attestation Results ?
- The appraisal policy for Attestation Results must take care of each individual component as well as the coherence of the whole assembly

Attestation Results – Few Ideas

- Current Attestation Results Format (specifically EAR and AR4SI)
  - ➢ Need some modification to express the Topological relationship between the Appraised components

❖ **A**ttestation **R**esults for **S**ecure **I**nteractions – **AR4SI**
https://datatracker.ietf.org/doc/draft-ietf-rats-ar4si/

❖ **E**at **A**ttestation **R**esults - **EAR**
https://datatracker.ietf.org/doc/draft-fv-rats-ear/

❖ **EAR : github.com/veraison/ear**

❖ **RUST EAR: github.com/veraison/rustear**

❖ **Python EAR: https://github.com/veraison/python-ear**

CMW can also carry EAT Attestation Result (EAR)
This allows easy expression of Composition as CMW can express recursive topology (CMW inside a CMW)

- Have a Use Case which has Composite Attestation Story ?
  - Communicate to us via rats@ietf.org
  - Veraison Zulipchat (**https://veraison.zulipchat.com/**)

!! THANK YOU !!