IETF Remote Attestation Architecture Overview

Dave Thaler (dthaler@microsoft.com)

Remote ATtestation procedureS (RATS) WG

- Internet Engineering Task Force working group chartered to do architecture and standardize data formats
  - But not protocols or code
  - WG documents: https://tools.ietf.org/wg/rats/

- RATS arch doc and CCC deep dive whitepaper reference each other and have several participants in common
  - https://tools.ietf.org/html/draft-ietf-rats-architecture
  - https://confidentialcomputing.io/whitepaper-02-latest

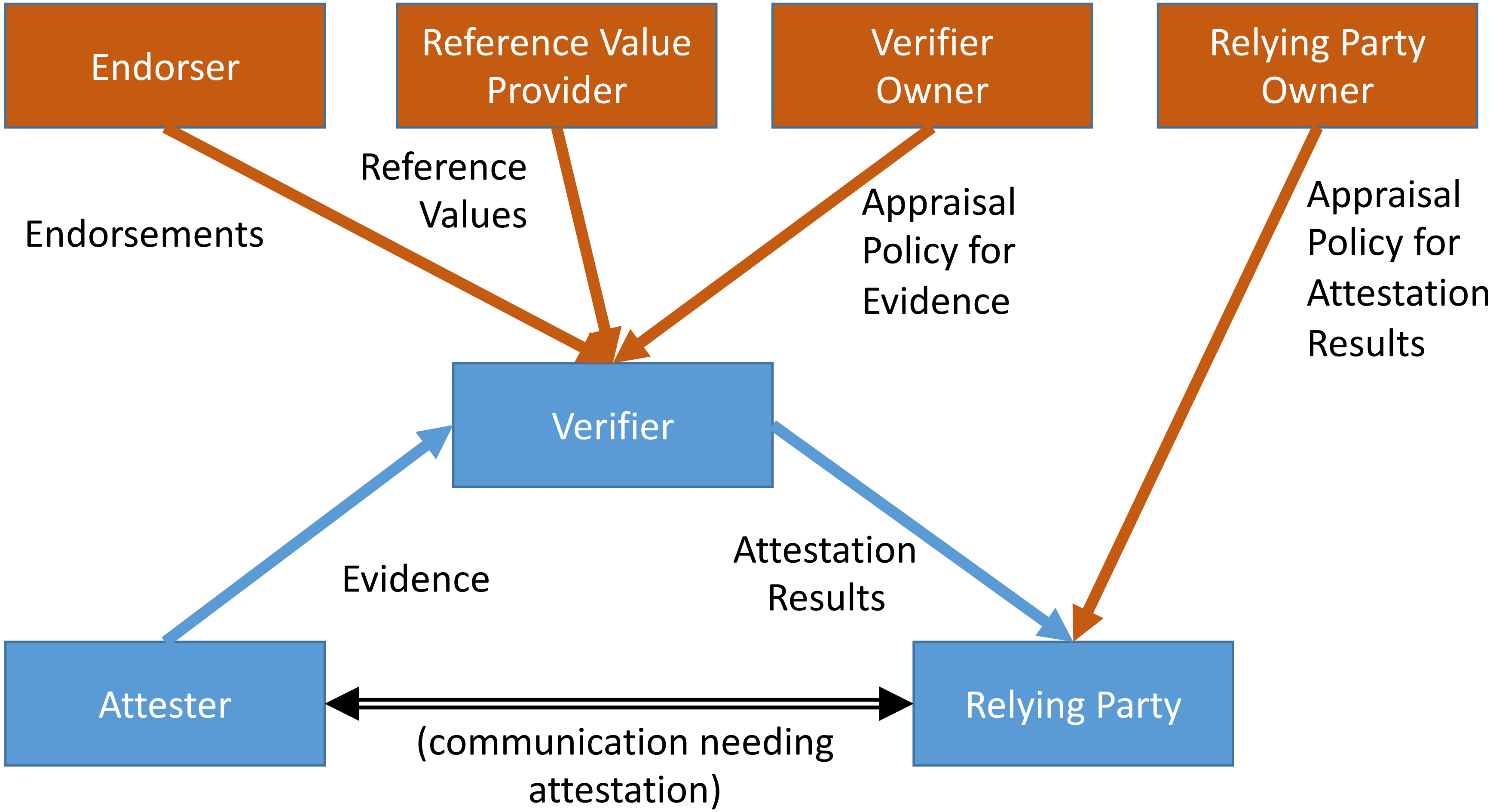
What is attestation

- Systems that have been attested and **verified to be in a good state (for some value of "good")** can improve overall system posture.
- For example:
  - A bank back-end system might refuse to transact with another system that is not known to be in a good state.
  - A healthcare system might refuse to transmit electronic healthcare records to a system that is not known to be in a good state.
- In Remote Attestation Procedures (RATS), one peer (the "Attester") produces believable information about itself - Evidence - to enable a remote peer (the "Relying Party") to decide whether to consider that Attester a trustworthy peer or not.

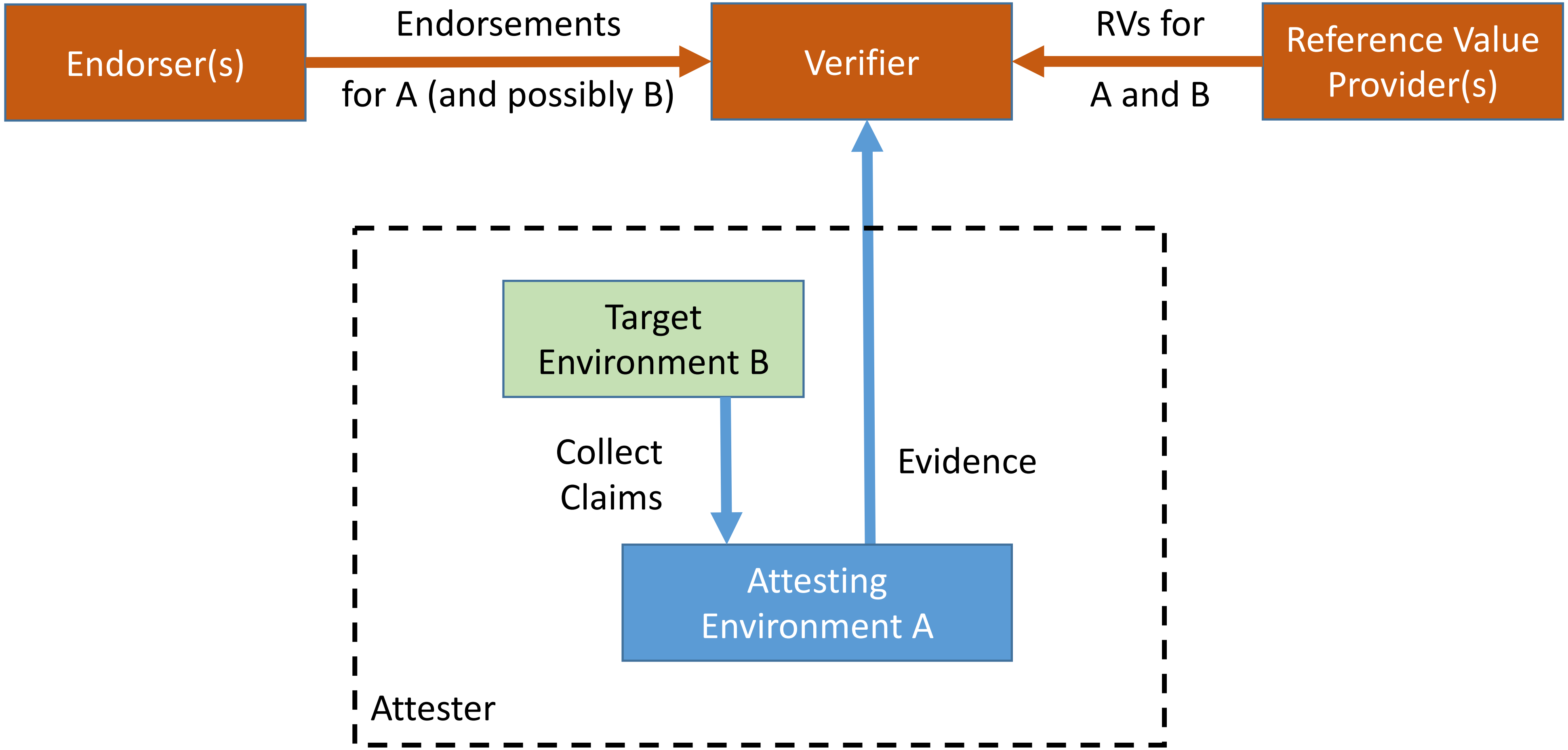
Example use cases

1. Network Endpoint Assessment
2. Confidential Machine Learning Model Protection
3. Confidential Data Protection
4. Critical Infrastructure Control
5. Trusted Execution Environment Provisioning
6. Hardware Watchdog
7. FIDO Biometric Authentication

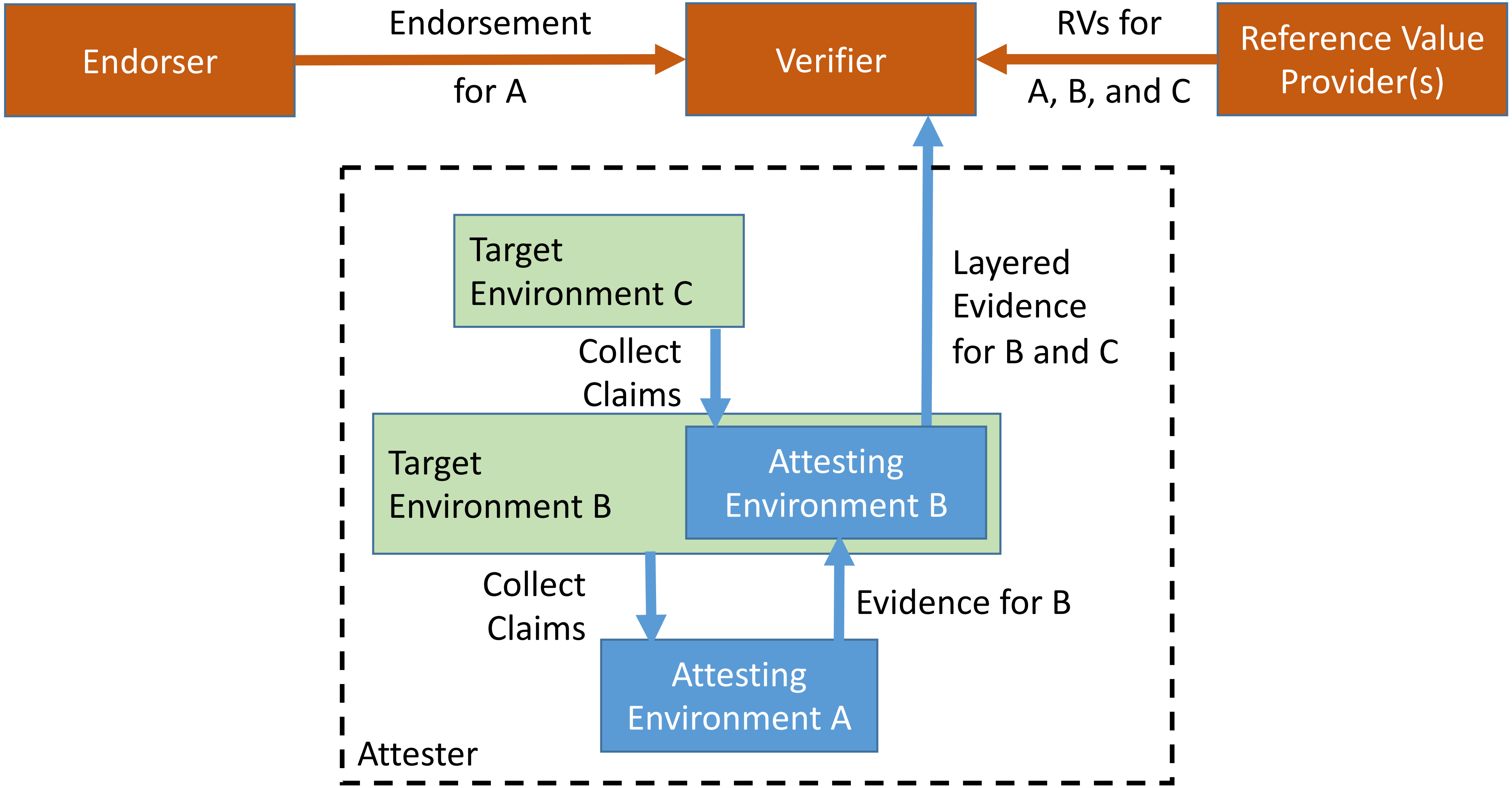
RATS Architecture: Conceptual Data Flow



Two types of environment in attester…



In general these can be chained…



Endorser

Endorsement for A

Verifier

RVs for A, B, and C

Reference Value Provider(s)

Target Environment C

Collect Claims

Layered Evidence for B and C

Target Environment B

Attesting Environment B

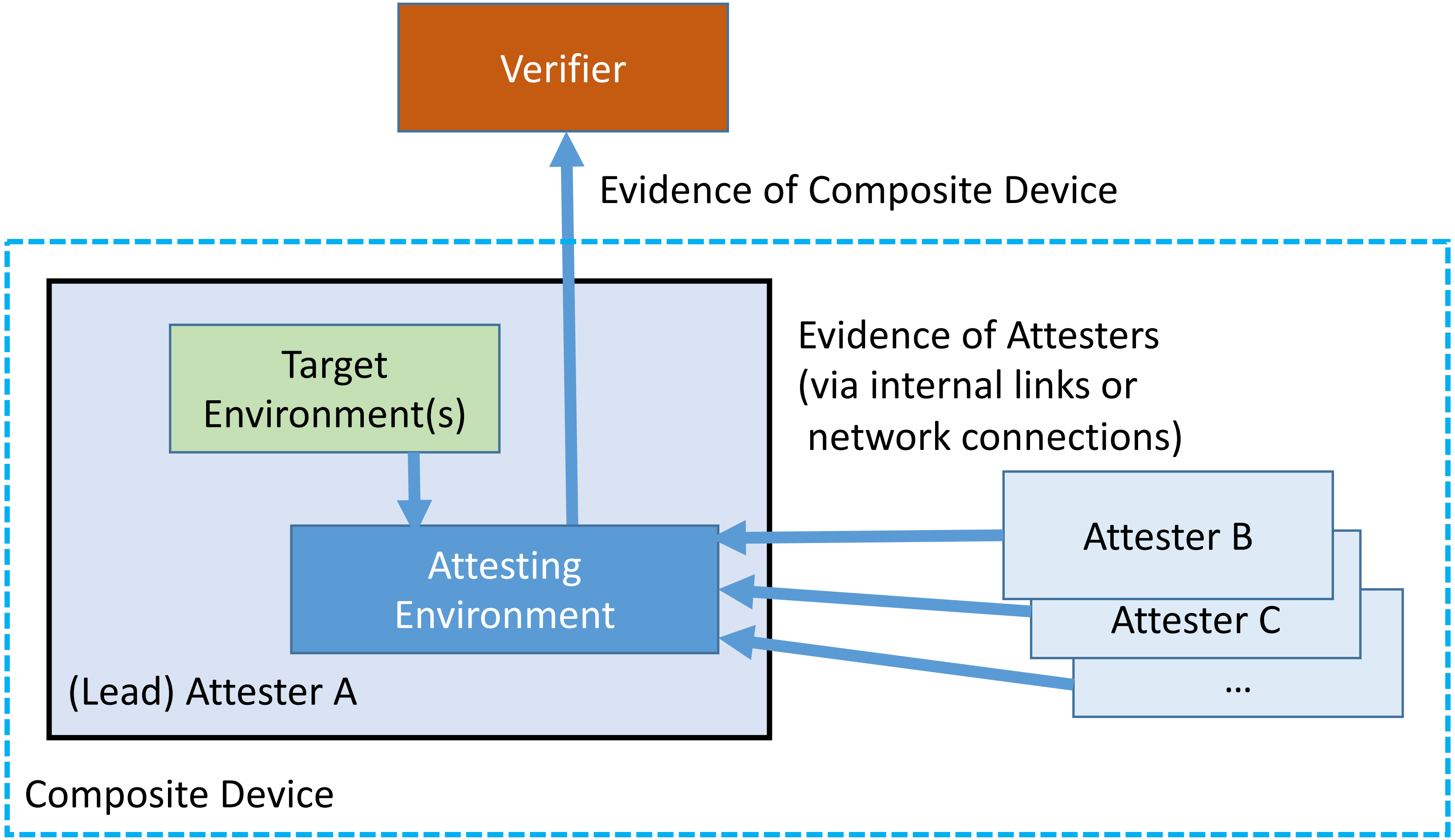
Collect Claims

Evidence for B

Attesting Environment A

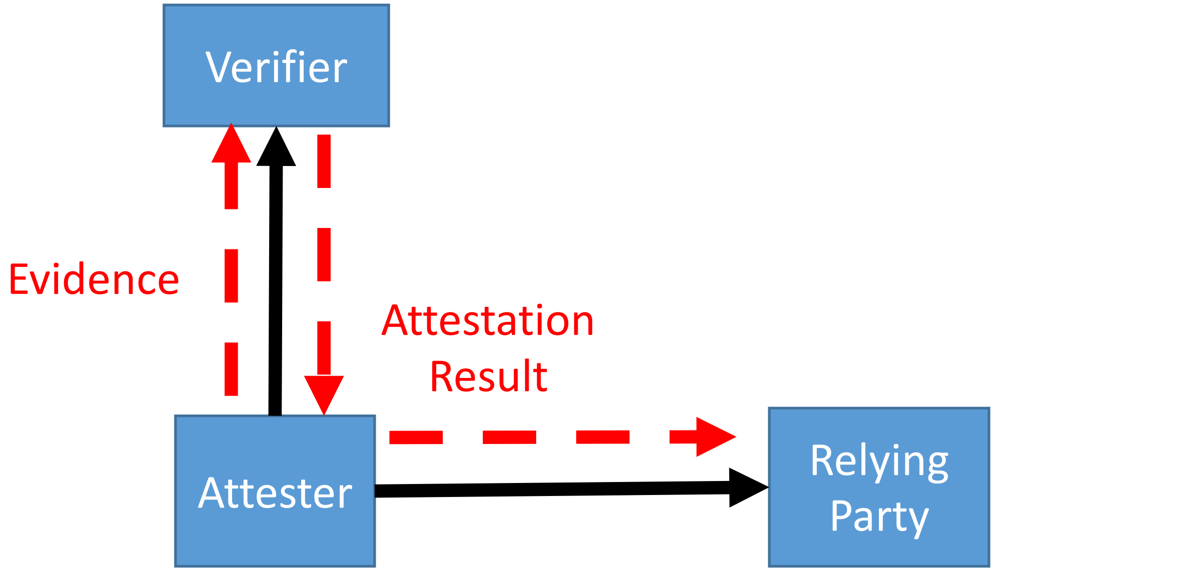
Attester

Might even have more complex devices…



Mapping conceptual data flow to protocols…

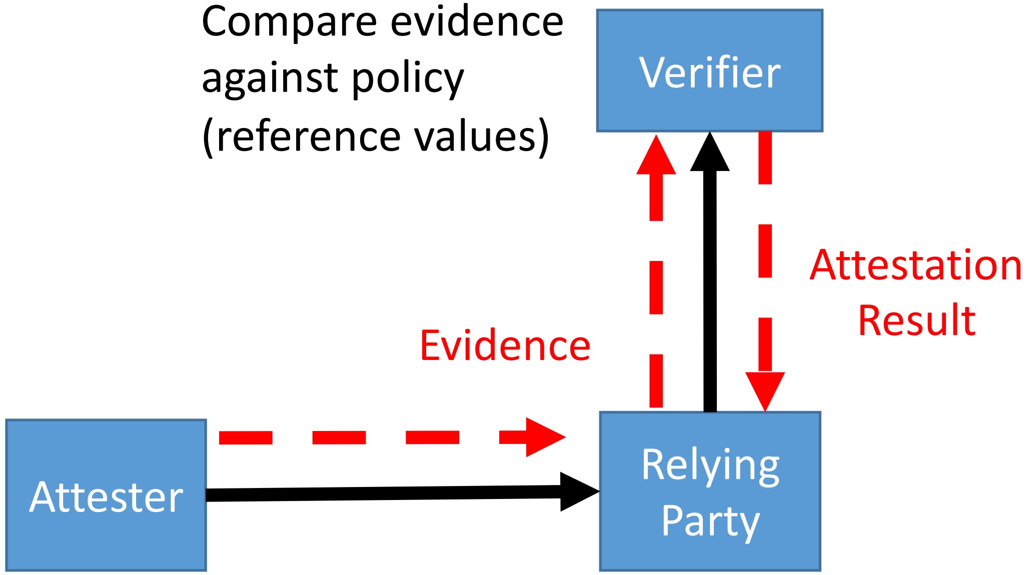
“Passport” model:



Compare attestation result against policy

“Background check” model:

Compare evidence against policy (reference values)

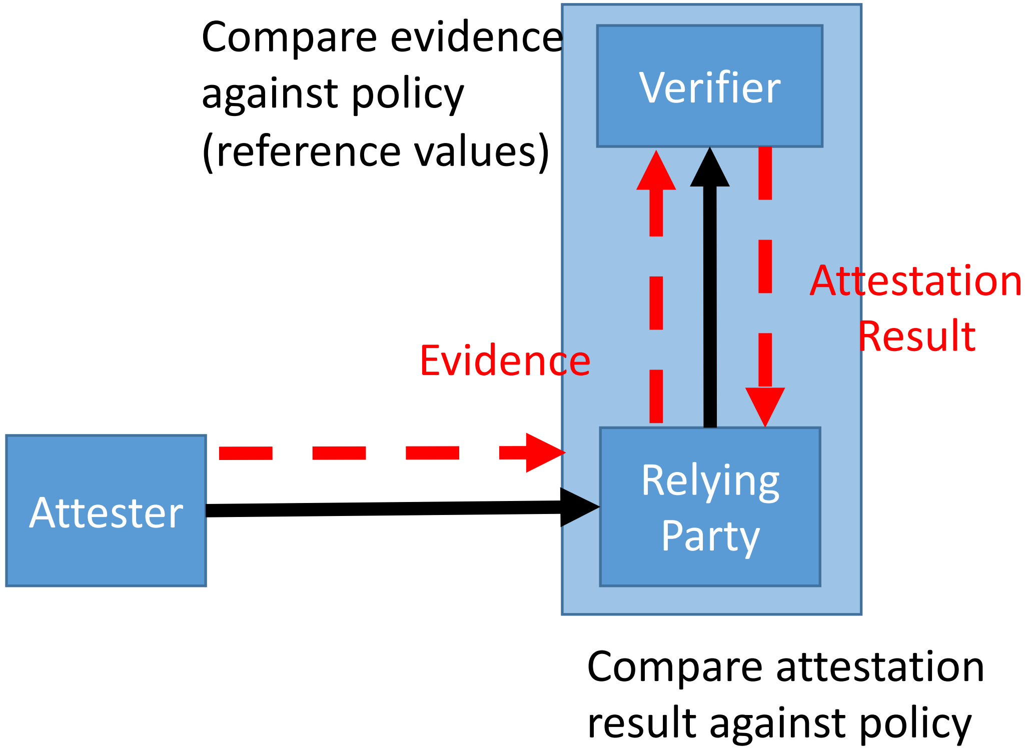


Compare attestation result against policy

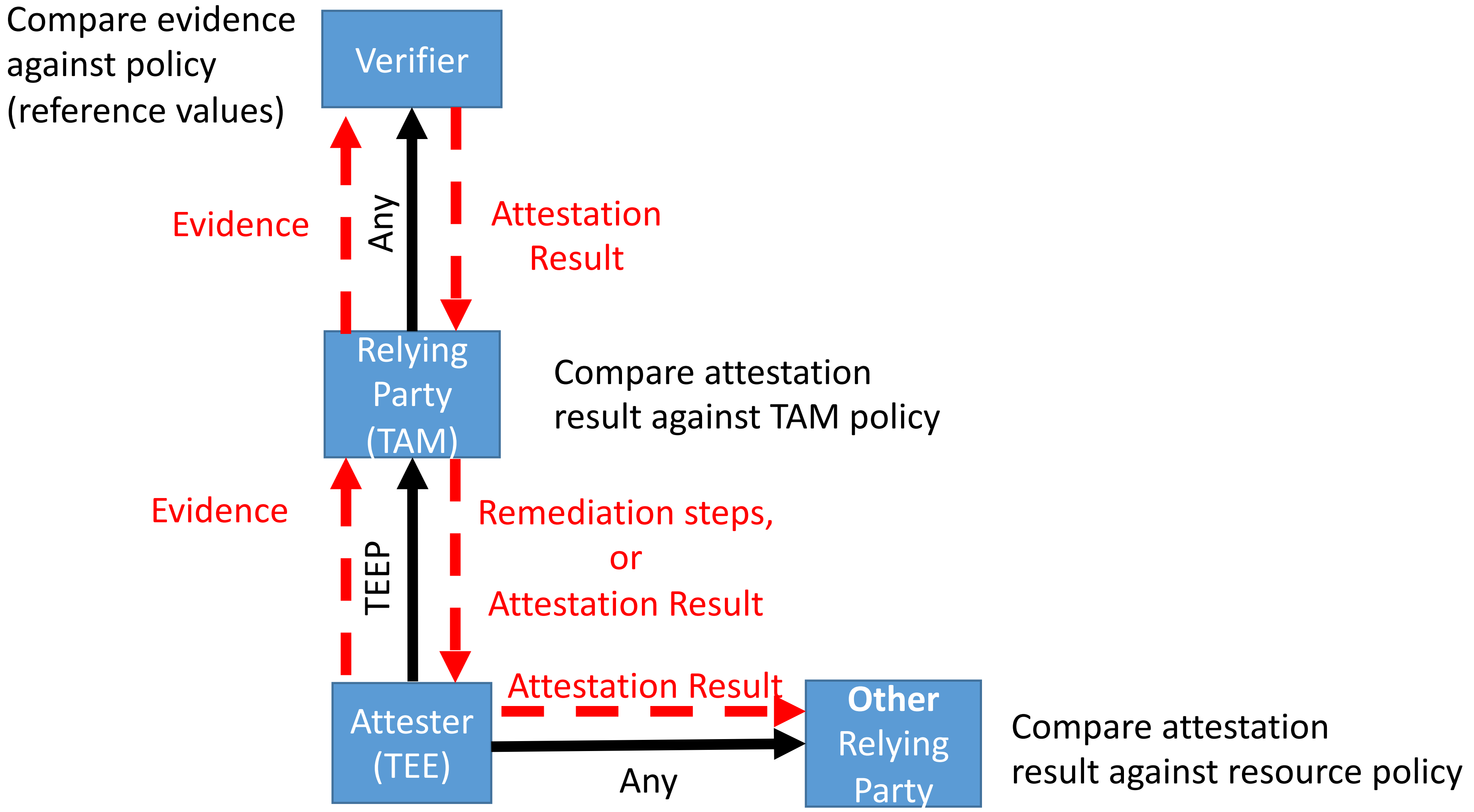
Variations on classic models

Verifier could also be combined into same device Relying Party

“Verifying RP” model:


Compare evidence against policy (reference values)
Verifier
Attestation Result
Evidence
Attester
Relying Party
Compare attestation result against policy

Example use case: TEE Provisioning via TEEP



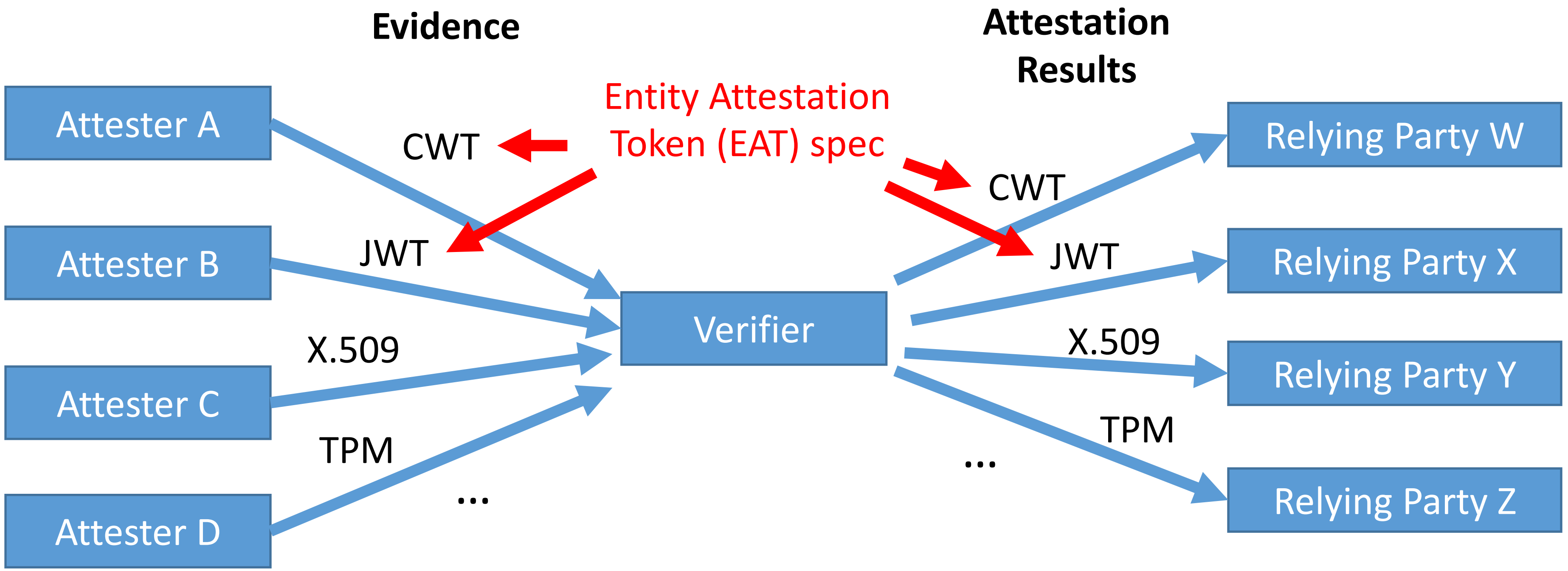
Compare evidence against policy (reference values)

Compare attestation result against TAM policy

Compare attestation result against resource policy

Relationship among formats

- Evidence, Attestation Results, and Endorsements can all have different claims formats
- There can be multiple formats possible for each one, including existing standard or proprietary formats, e.g.:



Freshness & replay protection

- Verifier cares about:
  - Was Evidence recently signed by Attester, not an old replay?
  - Are values of claims recent, not obsolete values in recent evidence?
- Relying Party cares about:
  - Was Attestation Result is recently signed by Verifier, not an old replay?
  - Are values of any claims recent, not obsolete values in recent results?
- How “recent” is up to the appraisal policy
- Details are up to the protocol, but there are three common ways…

Method 1: Timestamps

- Put timestamps in claims in Evidence and Attestation Results
- Requires roughly synchronized clocks
  - Requires a trusted source of time, internal or external
  - Requires secure time sync protocol (e.g., ntpsec inside TEE)
- Also adds claims about the signer’s time sync mechanism
- No additional messages or state at attestation time

Method 2: Nonces

- Receiver supplies nonce that sender must include in signed Evidence or Attestation Results
- No dependency on time sync or clocks at senders
- Receivers have to keep state to remember each nonce supplied until it’s used
- Receivers need a clock to “expire” nonces, but need not be synced
- Only addresses freshness of Evidence / Attestation Results, not freshness of claim values

Method 3: Epoch IDs

- Some “epoch ID distributor” periodically sends out epoch IDs to sender(s) and receiver(s)
- Senders use latest epoch ID in all messages in place of nonce
- Receivers check if received ID is in most recent set (e.g., of size 2)
- Receiver state is constant, compared to nonces
- Only epoch ID distributor requires a reliable clock
- “Recency” policy limited to a multiple of ID distribution period

Questions?