openSSL
COMMUNITIES

Quantum-Resilient Confidential Computing

Securing Data in-use in the AI Era

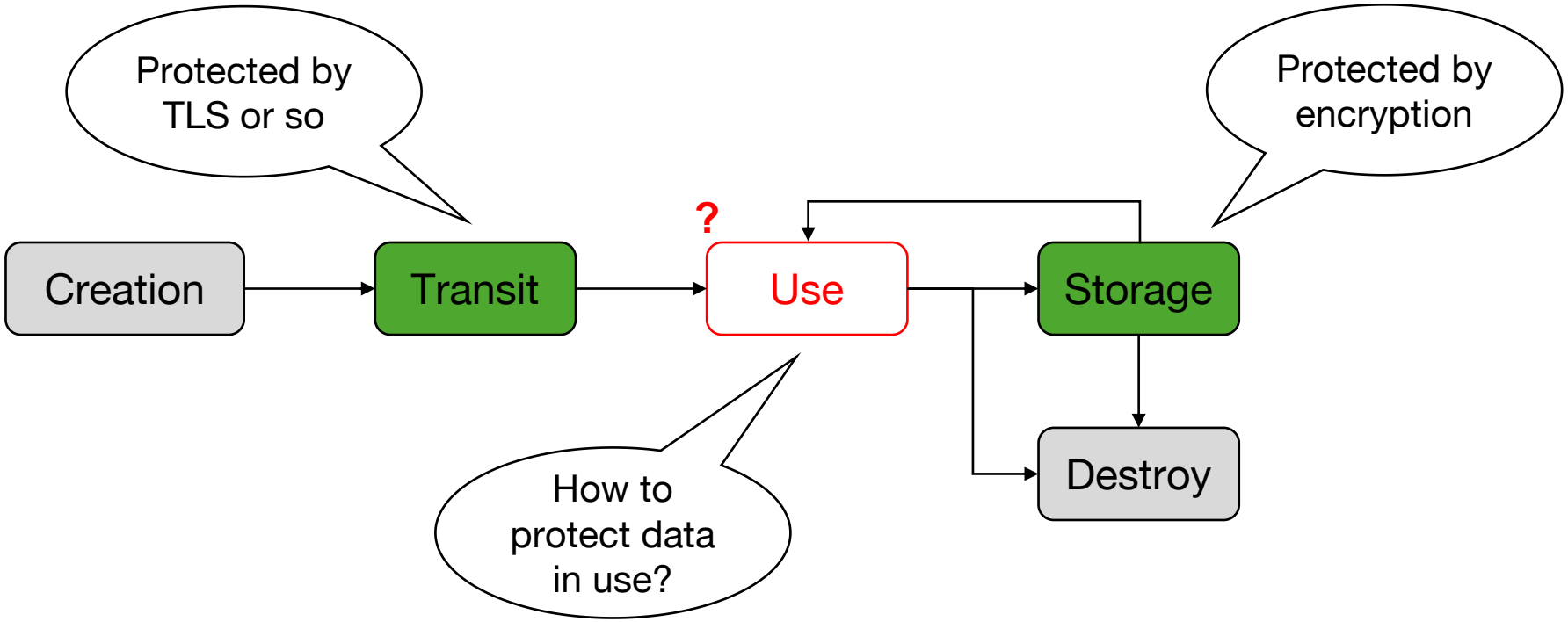
Paul Yang

OpenSSL Corporation Business Advisory Committee
OpenSSL Committer

Oct 2026

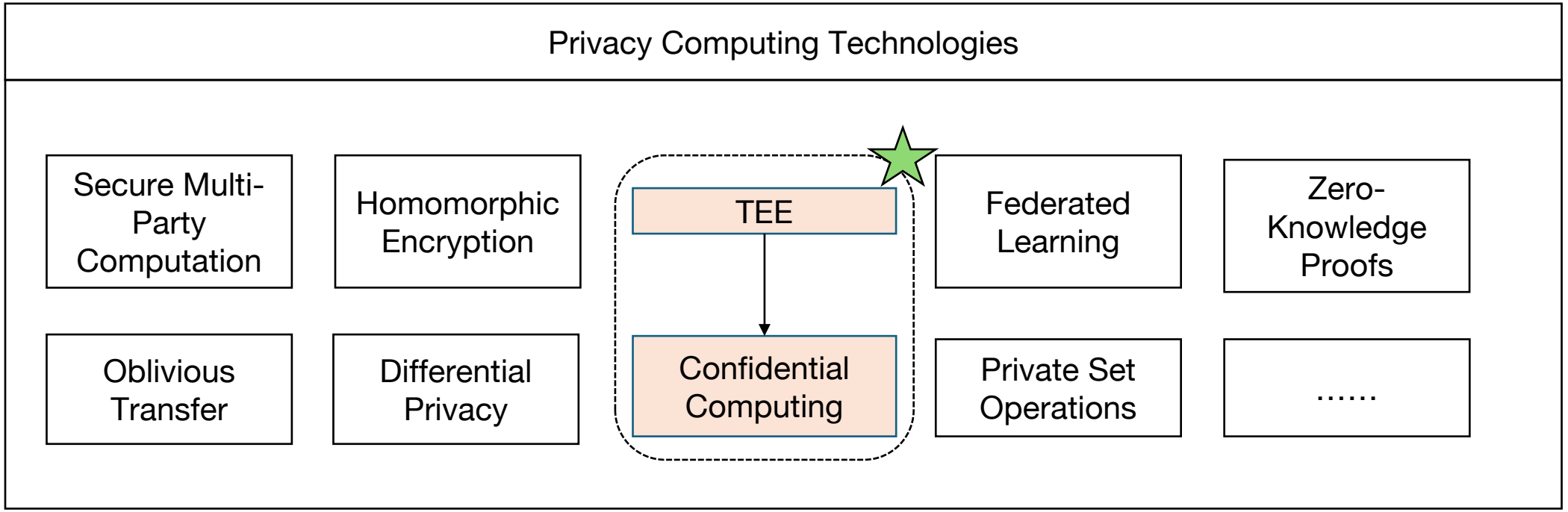
Issue of Data Security Today

The typical life cycle of data



Privacy Computing is the solution

Privacy Computing

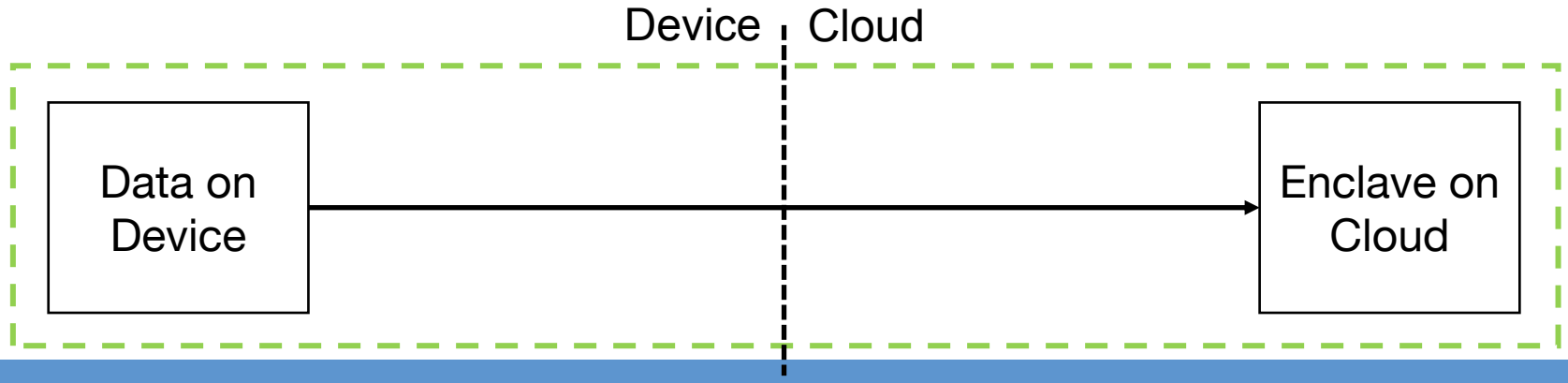


Privacy computing is a set of technologies—mainly SMPC, HE, DP, FL, ZKP, PSO, OT and Trusted Execution Environments—that allow multiple parties to jointly compute on sensitive data without revealing their raw inputs to each other.

What is Confidential Computing

Confidential computing is a technology that protects data while it is **being processed (data in-use)** by running it inside a secure, isolated **hardware-based** environment - TEE.

TEE(Trusted Execution Environment) is a hardware-level capability found in CPUs, GPUs, and other chips. It leverages cryptographic techniques such as encryption and signing, built upon a hardware root of trust, and serves as the core enabling technology for confidential computing.



Why Confidential Computing

	Confidential Computing	Other Technologies
Principle	Based on a secure environment isolated by hardware	Based on pure mathematical operations via cryptographic algorithms
Performance	High — performance loss is within 2% of non-TEE environments	Low — e.g., AES performance is 10^4 to 10^5 times faster than fully homomorphic encryption
Transparency	Strong — applications are unaware	Weak — specific algorithms have limited application scenarios; apps require adaptation
Universality	Medium — TEE requires hardware support, but is becoming a default CPU capability	High — no special hardware required
Cost	Mediium	High — requires more compute resources to offset performance issues, plus app adaptation
AI Friendliness	High	Low

TEE Support by Major Chip Vendors

Vendor	TEE Technology	Granularity	Typical Use
Intel	SGX (Software Guard Extensions) — Enclave-level TDX (Trust Domain Extensions, 4th Gen Xeon+) — VM-level Confidential Computing	Per-process (SGX) / Per-VM (TDX)	SGX: app-level enclaves, blockchain nodes; TDX: lift-and-shift confidential VMs
AMD	SEV (Secure Encrypted Virtualization) SEV-ES (Encrypted State) SEV-SNP (Secure Nested Paging, EPYC Milan/Genoa+) — VM-level with integrity protection	Per-VM (full memory encryption by AMD Secure Processor)	Confidential VMs in public cloud (Azure CVM, GCP CVM, AWS Nitro Enclaves w/ SEV)
ARM	TrustZone — Secure World / Normal World partition (runs Trusted OS like OP-TEE/Trusty) ARM CCA / Realms (ARMv9-A, newer server chips) — VM-level Realm TEE	System-wide SW/NW partition (TrustZone) / Per-Realm (CCA)	Mobile/IoT (biometrics, DRM, keystore); CCA: cloud confidential compute on ARM servers (Ampere Altra Max, Graviton4 preview)
NVIDIA	GPU Confidential Computing TEE (Hopper/Blackwell) — works with Intel TDX or AMD SEV-SNP CVM	Per-VM	Confidential AI inference or training

Two Pillars of Confidential Computing

- Remote Attestation - Prove the TEE is Authentic
  - Measure: TEE hashes all code/data into ‘Evidence’
  - Sign: HW RoT signs the Evidence + platform ID
  - Compare: Verifier compares Evidence against expected value (baseline)
  - Verify: Client verifies the result and decides whether to trust

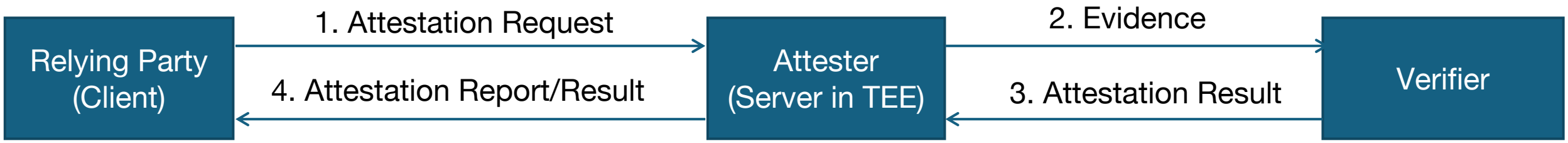
RFC 9334: Remote ATtestation procedureS (RATS) Architecture

H. Birkholz, D. Thaler, M. Richardson, N. Smith, W. Pan

INFORMATIONAL

Abstract

In network protocol exchanges, it is often useful for one end of a communication to know whether the other end is in an intended operating state. This document provides an architectural overview of the entities involved that make such tests possible through the process of generating, conveying, and evaluating evidentiary Claims. It provides a model that is neutral toward processor architectures, the content of Claims, and protocols.



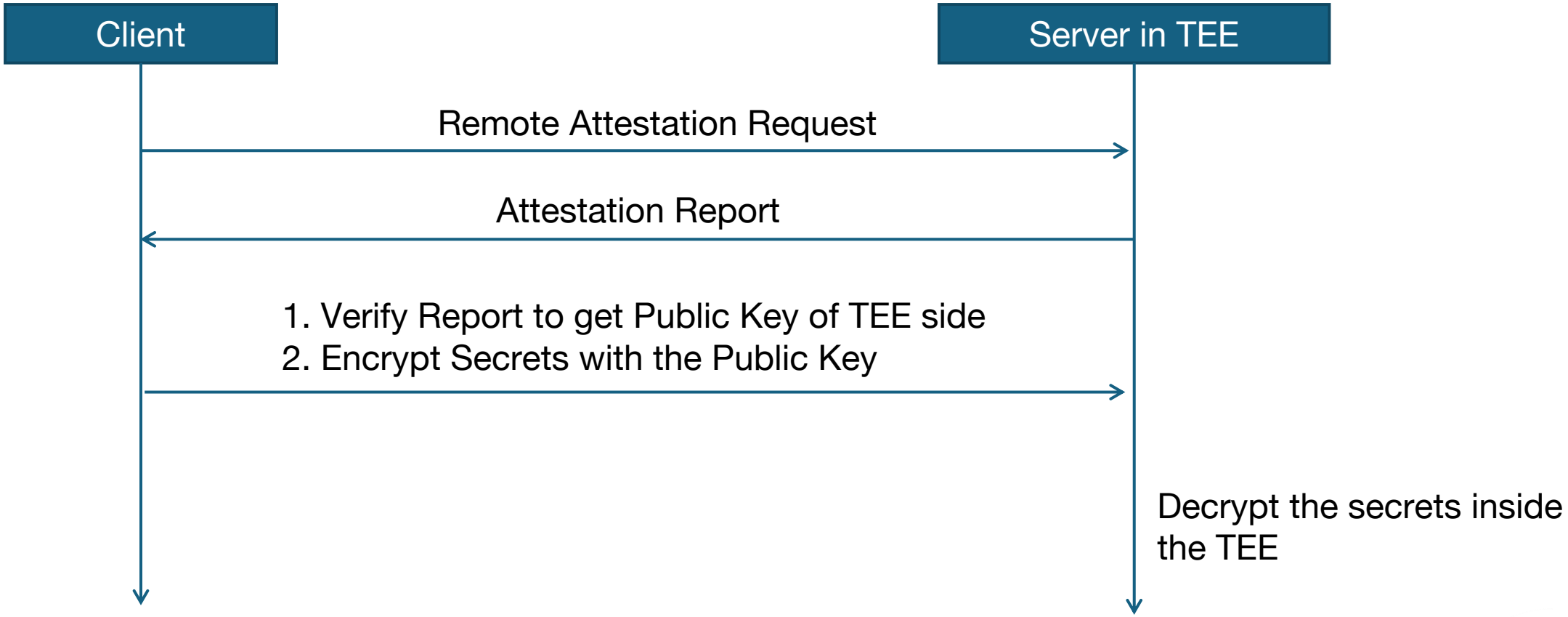
e.g. A user who has sensitive data to upload to the cloud/server

e.g. Cloud Services Provider

e.g. Intel Trust Authority

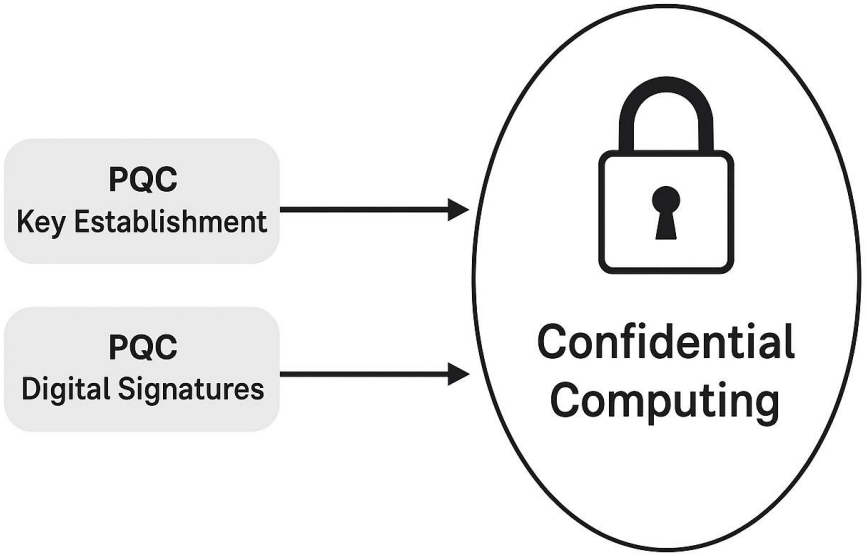
Two Pillars of Confidential Computing

- Key Provisioning - Inject Secrets Only into Verified TEEs



PQC and Confidential Computing

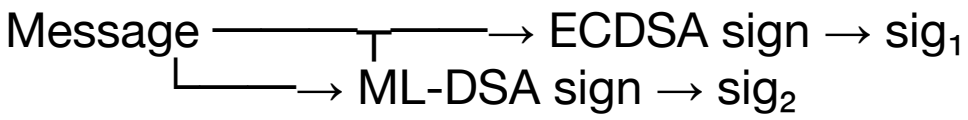
- Confidential computing technology relies on cryptography heavily
- Mainly focused on:
  - Digital signature algorithms
  - Key exchange mechanisms
  - Key generation
- Thus, PQC-transition also affects
  - Remote attestation
  - TLS
  - Combination of RA and TLS



Remote Attestation and Composite Sig

- Two approaches of composite signatures

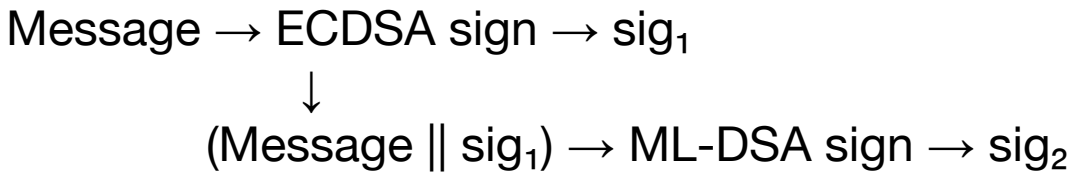
Parallel (dual-sign)



Composite = (sig_1, sig_2, alg IDs)

- Both signatures computed independently over same message
- Verifier checks both separately
- Pro: Simple, independent implementations, easy to audit
- Con: Need to verify twice due to the weak binding of the two signatures

Nested (sign-then-sign)

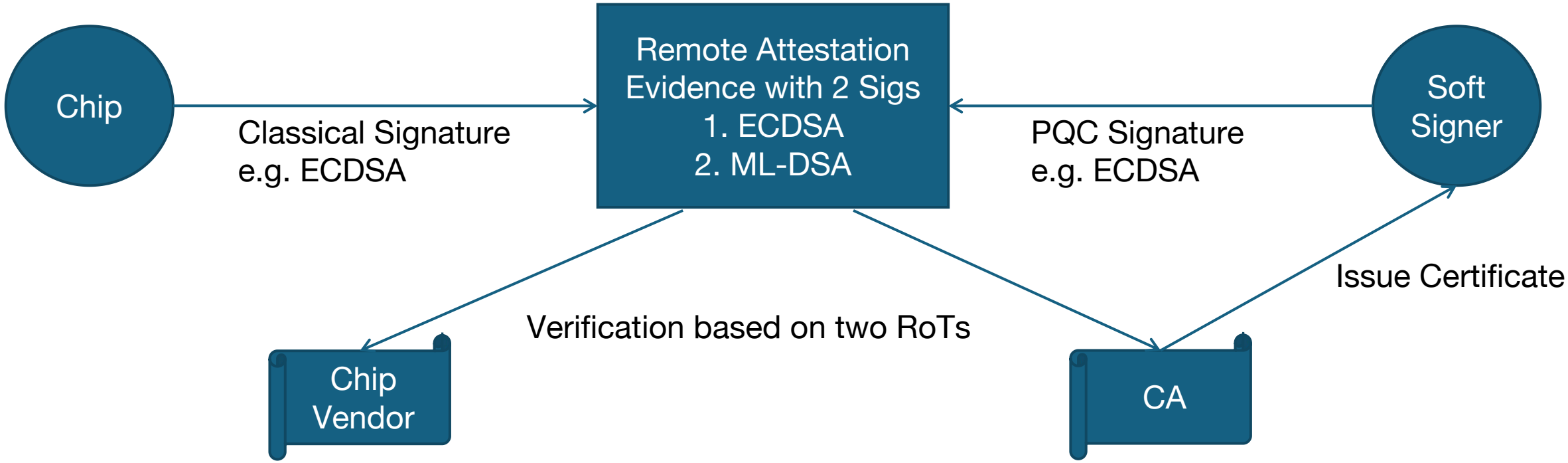


Composite = (sig_1, sig_2, alg IDs)

- Inner signature becomes part of outer signature's input
- Outer algorithm protects inner signature integrity
- Pro: Stronger binding, could only verify the outer layer only
- Con: Implementation complexity

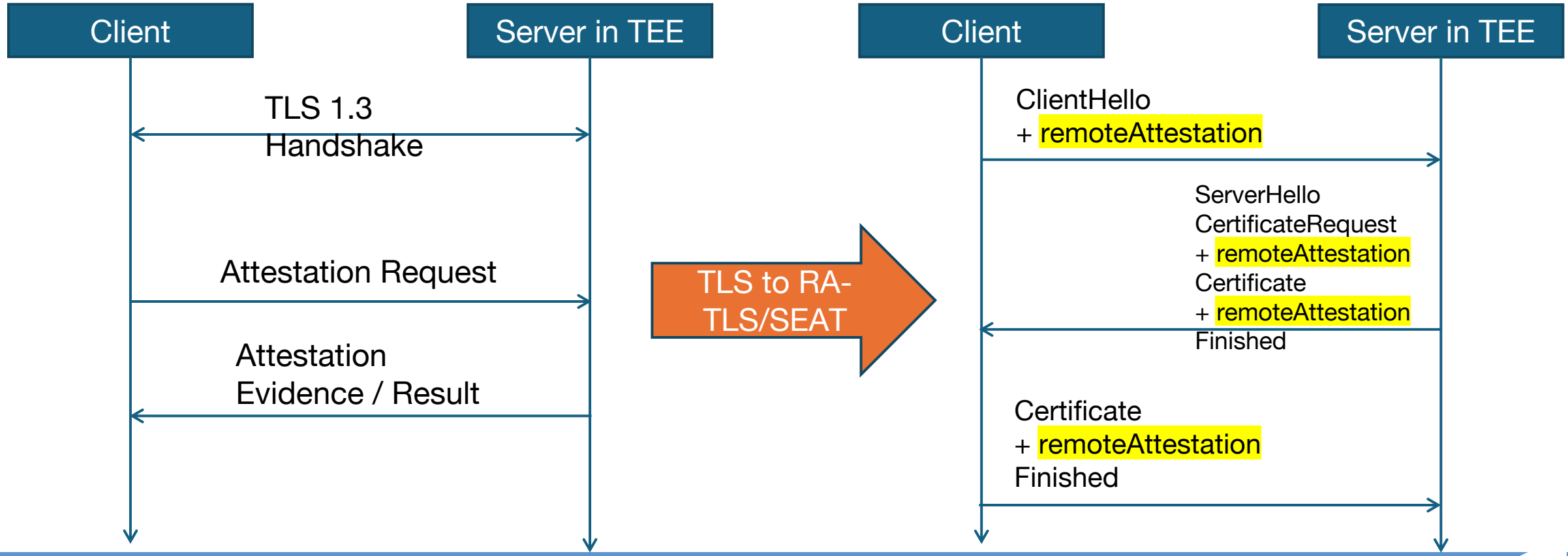
Remote Attestation and Composite Sig

- Composite signatures in a typical remote attestation scenario.



RA-TLS (the SEAT concept)

- TLS provides identity, not runtime integrity. SEAT binds remote attestation into the TLS 1.3 handshake by introducing new TLS extensions



RA-TLS and PQ-RA-TLS

- To survive a quantum adversary, we need PQ-RA-TLS

Dimension	RA-TLS	PQ-RA-TLS
TLS key exchange	ECDHE (X25519 / P-256)	Hybrid KEM (X25519 + ML-KEM) or ML-KEM only
TLS authentication	ECDSA / EdDSA / RSA certs	ML-DSA / SLH-DSA / Falcon, or hybrid (ECDSA + ML-DSA)
Attestation signing	ECDSA / ECDAA / RSA-based platform keys	PQ or hybrid platform keys
Long-term confidentiality	Vulnerable to harvest-now-decrypt-later	Resistant to quantum retrospective decryption
Key / signature sizes	Small	Much larger (especially ML-DSA / SLH-DSA)
Implementation impact	Existing OpenSSL / TEE SDKs	Needs PQ provider + TEE SDK upgrades

How will OpenSSL Power all of this?

- OpenSSL helps by enabling:
  - Quantum-safe TLS 1.3
    - ML-KEM hybrid key exchange
  - Quantum-safe code signing and certificates
    - ML-DSA or SLH-DSA with X.509
  - PQC-ready infrastructure stacks
    - NGINX, Apache
  - Crypt-agile development
    - The Provider mechanism swapping PQC without app changes
  - PQC compliance
    - NIST FIPS, NSA CNSA 2.0, EU PQC transition guidelines…

openSSL
COMMUNITIES