Attestation and Verification

The elephant in the confidential computing room

sameo@rivosinc.com - LPC 2022
Protecting Data in Use

Confidential Computing is the protection of data in use by performing computation in a hardware-based Trusted Execution Environment. (CCC whitepaper)
Protecting Data in Use

Confidential Computing is the protection of data in use by performing computation in a hardware-based Trusted Execution Environment. (CCC whitepaper)

Protecting data in use is of limited value if you can’t trust who generates and uses it.
Protecting Data in Use

Confidential Computing is the protection of data in use by performing computation in a hardware-based Trusted Execution Environment. (CCC whitepaper)

Protecting data in use is of limited value if you can’t trust who generates and uses it.

Can you trust your guest SW stack?

Can you trust your Confidential Compute hardware?
Protecting Data in Use

Confidential Computing is the protection of data in use by performing computation in a hardware-based Trusted Execution Environment. (CCC whitepaper)

Protecting data in use is of limited value if you can’t trust who generates and uses it

Can you trust your guest SW stack?

Can you trust your Confidential Compute hardware?

Attest and Verify

Confidential Computing without attestation and verification is not confidential
Our Main Focus

- Confidential Computing Node (TDX, SEV, PF, etc)
- Linux Kernel
- KVM
- Guest Kernel
- Guest Workload
- TVM

Get Evidence
What We Tend to Forget

- Confidential Computing Node (TDX, SEV, PF, etc)
  - Linux Kernel
    - KVM
    - Guest Kernel
      - Guest Workload
        - TVM
          - Get Evidence
            - Send Evidence
          - Send Evidence
            - Magic Box
What We Tend to Forget

- Confidential Computing Node (TDX, SEV, PF, etc)
  - Linux Kernel
  - KVM
  - Guest Workload
- TVM
- Magic Box

- Send Evidence
- Inject secret
- Establish secure channel

- Get Evidence
What We Tend to Forget

- TVM
  - Get Evidence
  - Send Evidence
- Guest Workload
  - Inject secret
  - Establish secure channel
- Guest Kernel
- Magic Box a.k.a. Relying Party
- Linux Kernel
- KVM
- Confidential Computing Node (TDX, SEV, PF, etc)
The IETF RATS Magic Box

TVM

Send Evidence

Relying Party

Secret
The IETF RATS Magic Box

- TVM
- Relying Party
- Verifier

Flow:
- Send Evidence from TVM to Relying Party
- Send Evidence from Relying Party to Verifier
- Secret from Relying Party to TVM
- Attestation Results from Verifier to Relying Party
The IETF RATS Magic Box

- TVM
- Relying Party
- Verifier
- Reference Values Provider

Flow:
- TVM → Relying Party → Verifier → Reference Values Provider
- Reference Values Provider → TVM
- Relying Party → Verifier
- Verifier → Reference Values Provider

Labels:
- Send Evidence
- Secret
- Attestation Results
- Reference Measurements
The IETF RATS Magic Box

TVM

Relying Party

Verifier

TVM Supply Chain

Signed Reference Values

Reference Values Provider

Send Evidence

Send Evidence

Secret

Attestation Results

Reference Measurements

Reference Values
The IETF RATS Magic Box

1. Send Evidence
2. Protocol?
3. Format?
4. Send Evidence
5. Reference Values Provider
6. TVM Supply Chain
7. Signed Reference Values
8. Reference Measurements
9. Policies
10. Verifier Owner
11. Protocol?
12. Secret
13. Attestation Results
The IETF RATS Magic Box

- TVM
- Relying Party
- Verifier
- Reference Values Provider

- Send Evidence
- Send Evidence
- Manufacturer Endorsement?
- Reference Measurements
- Signed Reference Values

- Protocol?
- Format?
- Protocol?
- Format?
- Protocol?
- Format?
- Protocol?
The IETF RATS Magic Box

TVM

Relying Party

Verifier

TVM Supply Chain

Reference Values Provider

Protocol? Format?

Send Evidence

Send Evidence

Manufacturer Endorsement?

Reference Measurements

Protocol? Format?

Format+ Protocol?

Signed Reference Values

Protocol? Format?

Format? Protocol?

Protocol? Format?

Format?

Secret

Attestation Results

Policies

Verifier Owner
End-to-End Confidential Computing

Platform enabling is only one part of the confidential computing chain...

The rest is “only” about deploying and interacting with attestation services

Interfaces, protocols, formats, and manufacturer interactions are very fragmented

Usually manufacturer or/and cloud provider specific

Multiple IETF initiatives to clear the mess (RATS)

Plumbing an evidence into an attestation service is very challenging
The Confidential Containers Approach

Generic and open Interfaces, plugin architecture for existing manufacturer solutions
The Confidential Containers Approach

Generic and open Interfaces, plugin architecture for existing manufacturer solutions

Generic Relying Party protocol - [KBS protocol (HTTPS & JSON Web Encryption)](KBS protocol (HTTPS & JSON Web Encryption))

Hardware agnostic Evidence format - [TCG DICE or RATS EAT](TCG DICE or RATS EAT)

Manufacturer-pluggable Verifier architecture - [Attestation Service framework](Attestation Service framework)

Manufacturer agnostic Reference Value Provider Service - [The RVPS crate](The RVPS crate)

   With Support for modern and open source supply chain architectures - [In-toto+SLSA](In-toto+SLSA)

Open, cloud native format for Policies - [Open Policy Agent](Open Policy Agent)
Calling Out for Standardization

TVM

Relying Party

Verifier

Reference Values Provider

TVM Supply Chain

Signed Reference Values

in-toto/VLSA

Reference Measurements

Manufacturer Backend Plugins

KBS Protocol

TCG DICE RATS-EAT

Send Evidence

Send Evidence

Attestation Results

Policies

Verifier Owner

KBS Protocol

RATS-EAT

RATS-EAT

RATS-EAT

OPA

KBS Protocol

Protocol

TCG DICE RATS-EAT

KBS Protocol

Protocol

TCG DICE RATS-EAT

KBS Protocol

Protocol
Calling Out for Standardization

Let’s make the next confidential computing software stack simpler...

Simplify the interfaces between the TVM and the Relying Party

Simplify the interaction between the supply chain and the Relying Party

Simplify the verification policies description
Calling Out for Standardization

TVM → Relying Party:
- Secret
- !(Send Evidence)

Relying Party → Verifier:
- Attestation Results

Verifier → Reference Values Provider:
- Policies
- Reference Measurements

Reference Values Provider → TVM Supply Chain:
- Signed Reference Values

TVM Supply Chain → Reference Values Provider:
- ![Signed Reference Values]

Diagram:
- TVM
- Relying Party
- Verifier
- Reference Values Provider
Calling Out for Standardization

Attestation Service

TVM

Send Evidence

Secret

Verifier Owner

TVM Supply Chain

Reference Values

Provider

Measurements

Verifier

Owner

Policies

TVM

Supply Chain
Calling Out for Standardization

TVM

DICE or EAT Evidence

KBS Protocol

Attestation Service

JSON Web Token/Key

TVM Supply Chain
Calling Out for Standardization

- TVM
- Attestation Service
- TVM Supply Chain

- DICE or EAT Evidence
- KBS Protocol
- in-toto pipeline
- VLSA Provenance
- JSON Web Token/Key
Calling Out for Standardization

TVM

Attestation Service

TVM Supply Chain

DICE or EAT Evidence

KBS Protocol

in-toto pipeline

VLSA Provenance

Open Policy Agent

Verification Policies

JSON Web Token/Key

Policies