

Formal Specification and Verification of Attestation in Confidential Computing

Muhammad Usama Sardar

TU Dresden

November 8, 2023

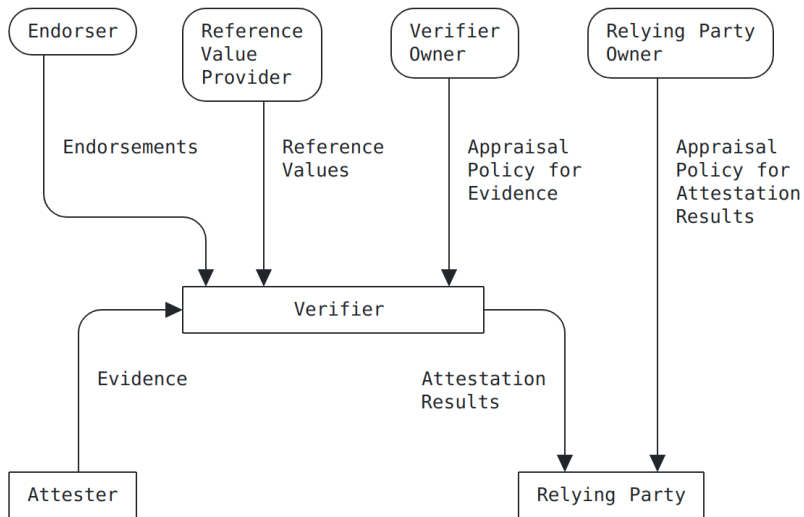
Funded by CPEC



Agenda

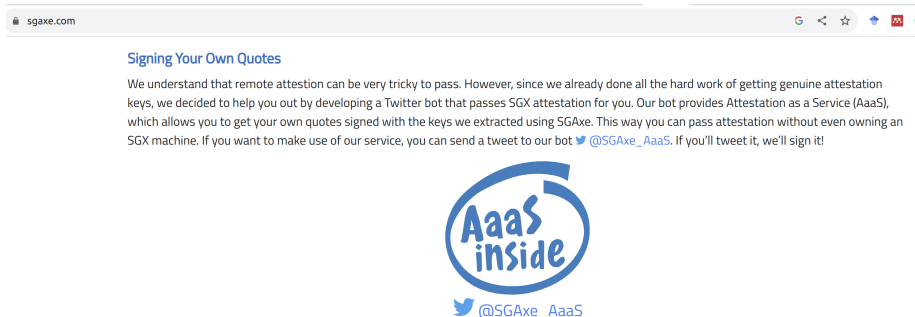
- 1 Motivation
- 2 Approach
- 3 Results
- 4 Summary

We all know RATS¹



¹Birkholz et al., *Remote Attestation procedureS (RATS) Architecture*, 2023.

But is RATS sufficient for CC (e.g., SGX)?²




The screenshot shows a web browser window with the address bar displaying "sgaxe.com". The page content includes a blue heading "Signing Your Own Quotes" followed by a paragraph of text explaining a service for SGX attestation. Below the text is a blue circular logo with the text "AaaS inside" and a Twitter handle "@SGAxe_AaaS" with a bird icon.

sgaxe.com

Signing Your Own Quotes

We understand that remote attestation can be very tricky to pass. However, since we already done all the hard work of getting genuine attestation keys, we decided to help you out by developing a Twitter bot that passes SGX attestation for you. Our bot provides Attestation as a Service (AaaS), which allows you to get your own quotes signed with the keys we extracted using SGAXe. This way you can pass attestation without even owning an SGX machine. If you want to make use of our service, you can send a tweet to our bot [@SGAxe_AaaS](#). If you'll tweet it, we'll sign it!



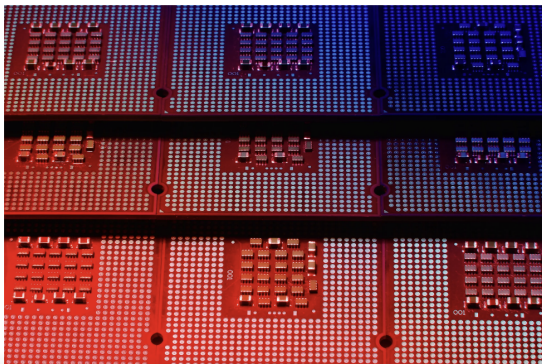
[@SGAxe_AaaS](#)

²www.sgaxe.com

LILY HAY NEMAN SECURITY APR 24, 2023 1:12 PM

Intel Let Google Cloud Hack Its New Secure Chips and Found 10 Bugs

To protect its Confidential Computing cloud infrastructure and gain critical insights, Google leans on its relationships with chipmakers.



PHOTOGRAPH: GETTY IMAGES

³Wired, *Intel Let Google Cloud Hack Its New Secure Chips and Found 10 Bugs*, 2023.

Outline

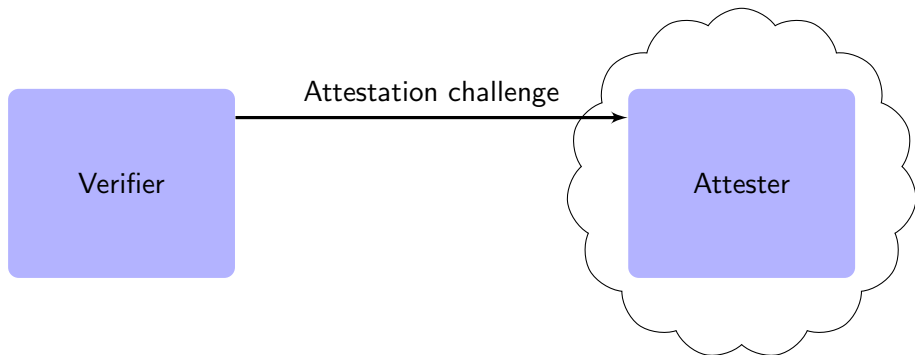
1 Motivation

2 Approach

3 Results

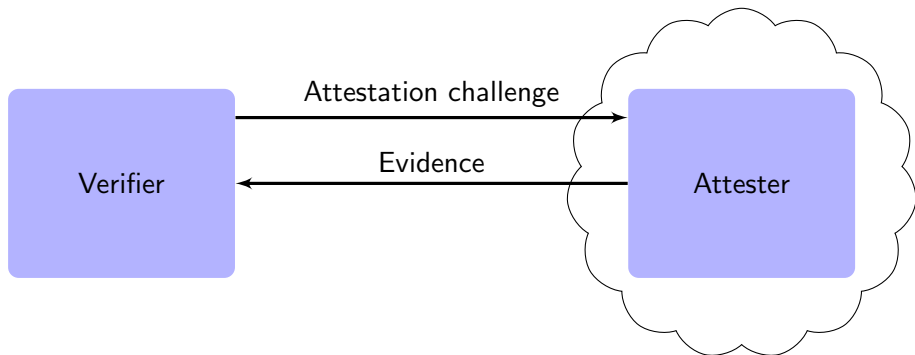
4 Summary

Architecturally-defined Attestation in CC⁴



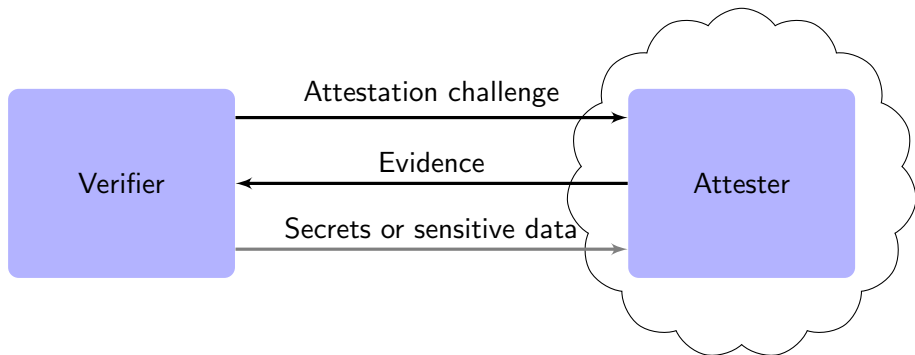
⁴Sardar and Fetzer, *Confidential Computing and Related Technologies : A Review*, 2021.

Architecturally-defined Attestation in CC⁴



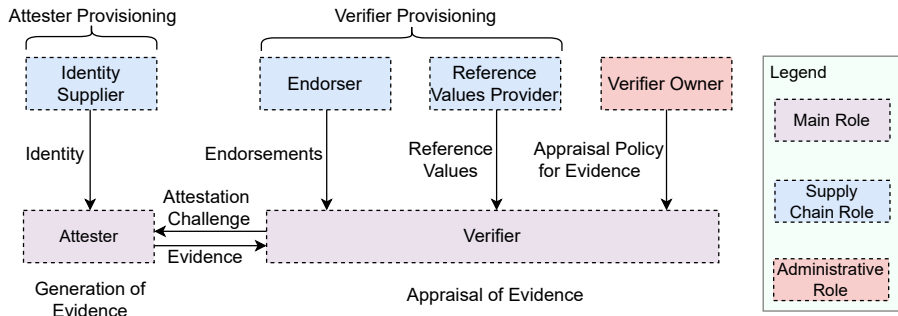
⁴Sardar and Fetzer, *Confidential Computing and Related Technologies : A Review*, 2021.

Architecturally-defined Attestation in CC⁴



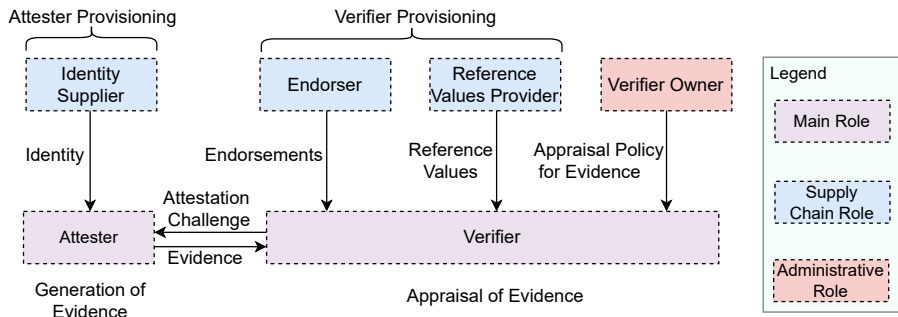
⁴Sardar and Fetzer, *Confidential Computing and Related Technologies : A Review*, 2021.

Architecturally-defined Attestation



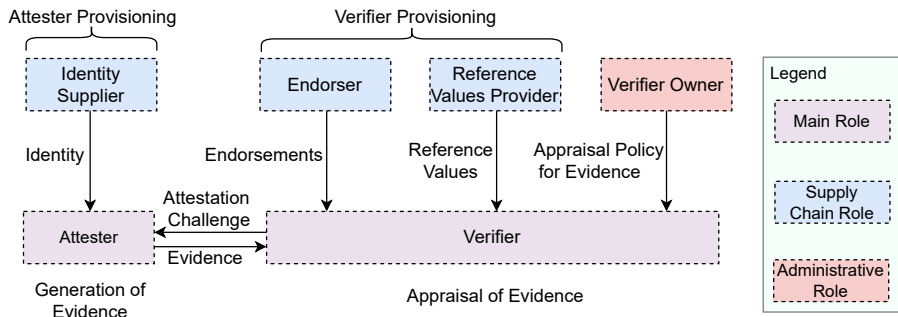
- Holistic coverage of phases

Architecturally-defined Attestation



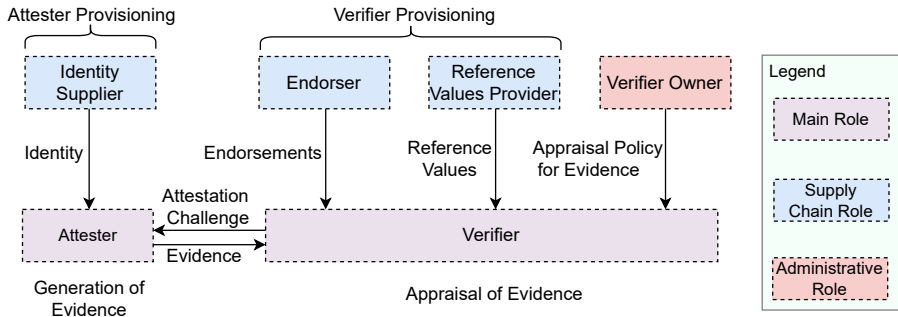
- Holistic coverage of phases
 - Provisioning

Architecturally-defined Attestation



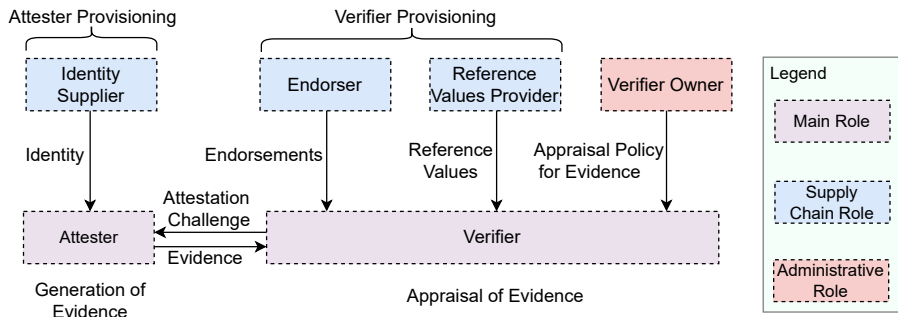
- Holistic coverage of phases
 - Provisioning
 - Attester Provisioning

Architecturally-defined Attestation



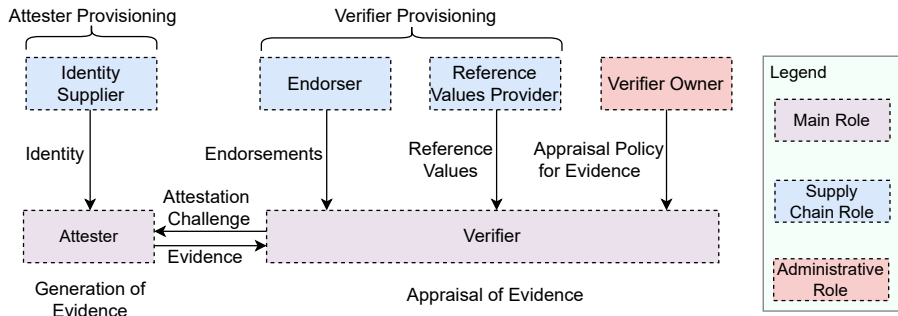
- Holistic coverage of phases
 - Provisioning
 - Attester Provisioning
 - Verifier Provisioning

Architecturally-defined Attestation



- Holistic coverage of phases
 - Provisioning
 - Attester Provisioning
 - Verifier Provisioning
 - Initialization

Architecturally-defined Attestation



- Holistic coverage of phases
 - Provisioning
 - Attester Provisioning
 - Verifier Provisioning
 - Initialization
 - Attestation Protocol

Contributions

- Most detailed formal model of Intel TDX attestation, including

Contributions

- Most detailed formal model of Intel TDX attestation, including
 - Certificate chain

Contributions

- Most detailed formal model of Intel TDX attestation, including
 - Certificate chain
 - Verifier steps

Contributions

- Most detailed formal model of Intel TDX attestation, including
 - Certificate chain
 - Verifier steps
 - Initialization phase

Contributions

- Most detailed formal model of Intel TDX attestation, including
 - Certificate chain
 - Verifier steps
 - Initialization phase
 - Variable measurements

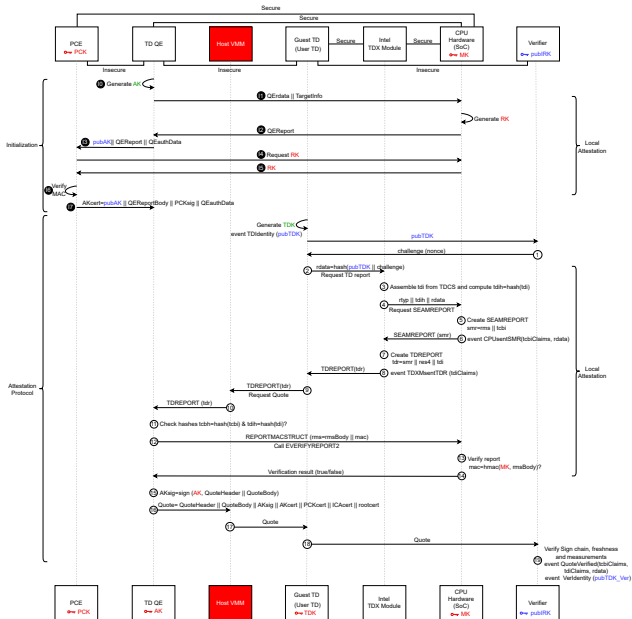
Contributions

- Most detailed formal model of Intel TDX attestation, including
 - Certificate chain
 - Verifier steps
 - Initialization phase
 - Variable measurements
- Formal proof of [insecurity](#) of Intel's claimed TCB

Contributions

- Most detailed formal model of Intel TDX attestation, including
 - Certificate chain
 - Verifier steps
 - Initialization phase
 - Variable measurements
- Formal proof of [insecurity](#) of Intel's claimed TCB
- First formal analysis of Arm CCA attestation

TDX Model



Challenge: Complicated designs with vague and outdated specs and very little support⁵

Intel Community / Developer Software Forums / Software Development Technologies / Intel® Software Guard Extensions (Intel® SGX)

1400 Diskussionen

index 1 in tdx tcb components

Abonnieren

Mehr Aktionen ▾



Usama S
Einsteiger

09-04-2023 • 01:11 AM • 111 Aufrufe



In the "Get TDX TCB Info" flow (<https://api.portal.trustedservices.intel.com/documentation#pcs-tcb-info-tdx-v4>), step 4 states:

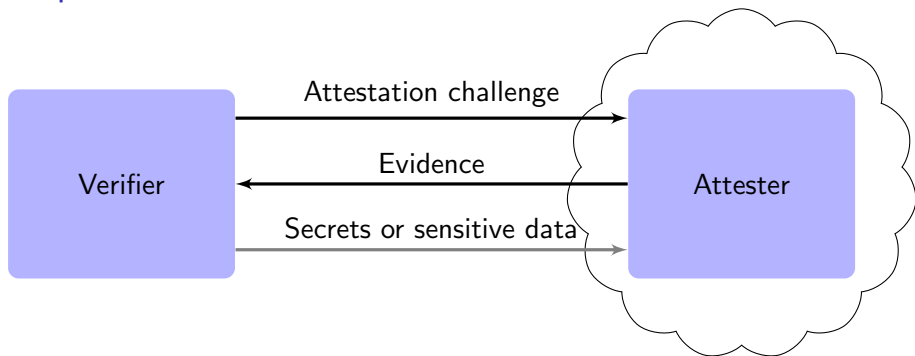
"For the selected TCB level verify that SVN at index 1 in tdx tcb components array matches the value of SVN at index 1 in TEE TCB SVNs array (from TD Report in Quote). In case of a mismatch the selected TCB level should be rejected as TCB Info that was used for the comparison is not supported for this platform configuration."

My question is:

What is so special about index 1 that it requires an equality check? What does index 1 represent? Typically all SVNs have a non-equality check (>=) as in step 3 (a,b,c).

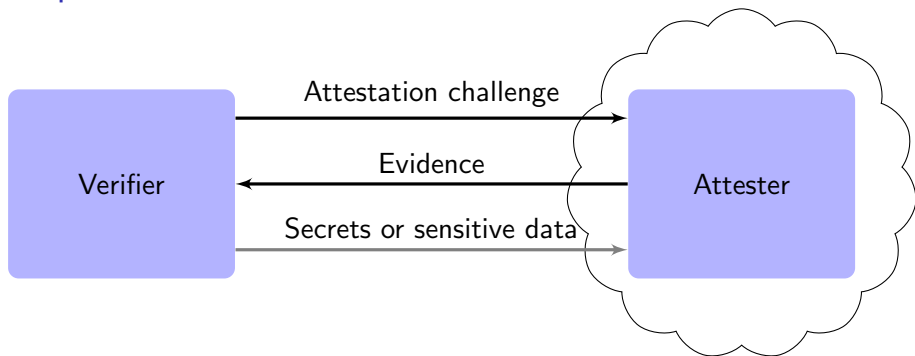
⁵<https://community.intel.com/t5/Intel-Software-Guard-Extensions/index-1-in-tdxtcbcomponents/m-p/1520194>

Properties



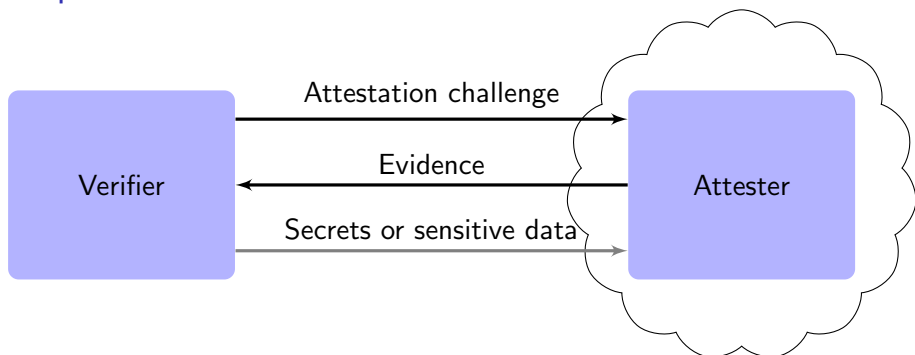
- Sanity checks

Properties



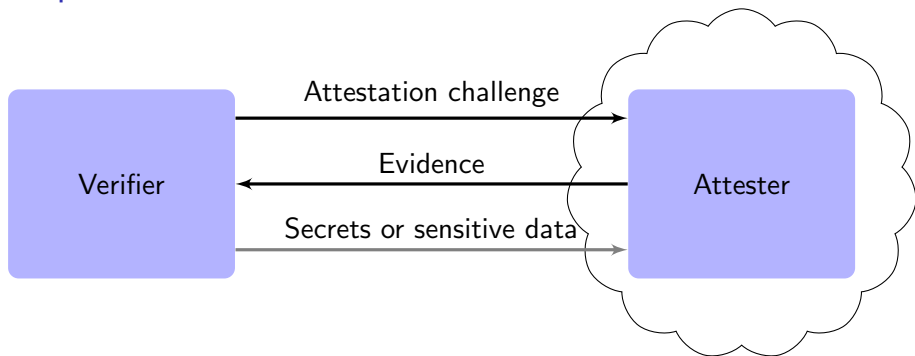
- Sanity checks
- Integrity of Evidence

Properties



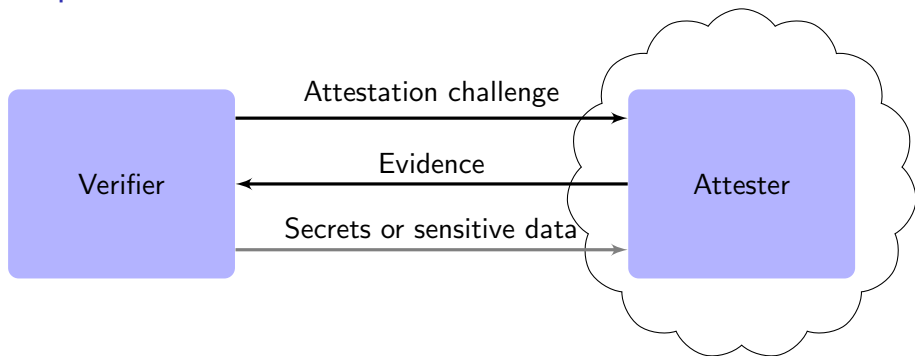
- Sanity checks
- Integrity of Evidence
- Freshness of Evidence

Properties



- Sanity checks
- Integrity of Evidence
- Freshness of Evidence
- Confidentiality/Secrecy of attestation-related keys

Properties



- Sanity checks
- Integrity of Evidence
- Freshness of Evidence
- Confidentiality/Secrecy of attestation-related keys
- Attester Authentication

Outline

1 Motivation

2 Approach

3 Results

4 Summary

TCB Claimed by Intel⁶

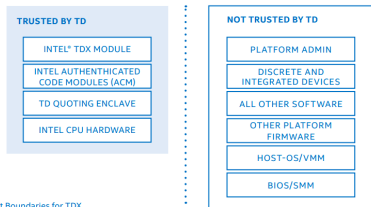
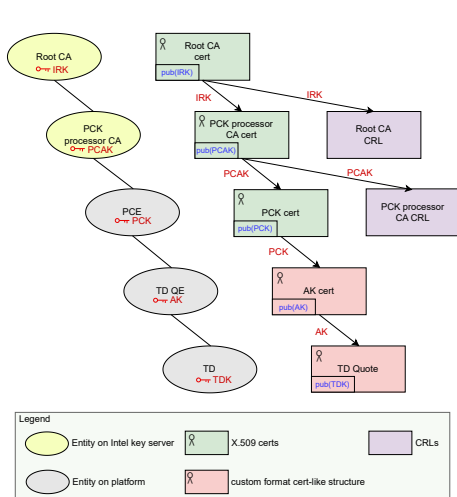


Figure 5.1. Trust Boundaries for TDX



⁶Intel, Intel (®) Trust Domain Extensions, 2021.

Verification Summary in ProVerif⁷

	Integrity	Freshness	Confidentiality	Authentication
Intel's claimed	✗	✗	✗	✗
Our proposed	✓	✓	✓	✗

```
.....
Verification summary:
Query not event(AKVerified(pubAK_1)) is false.
Query not event(CPUsentSMR(tcblClains_1,rdata_1)) is false.
Query not event(TDXMsentTDR(tdlClains_1)) is false.
Query not event(QuoteVerified(tcblClains_1,tdlClains_1,rdata_1)) is false.
Query not (event(TDIdentity(pubTDK_1)) && event(VerIdentity(pubTDK_Ver_1))) is false.
Query event(AKVerified(pubAK_1)) ==> event(AKsent(pubAK_1)) is true.
Query event(QuoteVerified(tcblClains_1,tdlClains_1,rdata_1)) ==> event(CPUsentSMR(tcblClains_1,rdata_1)) is false.
Query event(QuoteVerified(tcblClains_1,tdlClains_1,rdata_1)) ==> event(TDXMsentTDR(tdlClains_1)) is false.
Query lnj-event(QuoteVerified(tcblClains_1,tdlClains_1,rdata_1)) ==> lnj-event(CPUsentSMR(tcblClains_1,rdata_1)) is false.
Query lnj-event(QuoteVerified(tcblClains_1,tdlClains_1,rdata_1)) ==> lnj-event(TDXMsentTDR(tdlClains_1)) is false.
Query secret PCK_1,PCK is false.
Query secret PCAK is true.
Query secret AK_2,AK_1,AK is true.
Query secret MK_1,MK is true.
Query event(AKVerified(pubAK_PCE_1)) && event(AKsent(pubAK_1)) ==> pubAK_PCE_1 = pubAK_1 is true.
Query event(VerIdentity(pubTDK_Ver_1)) && event(TDIdentity(pubTDK_1)) ==> pubTDK_1 = pubTDK_Ver_1 is false.
.....
real    0m55,648s
user    0m55,432s
sys     0m0,132s
```

⁷Blanchet, Cheval, and Cortier, "ProVerif with lemmas, induction, fast subsumption, and much more", 2022.

Outline

1 Motivation

2 Approach

3 Results

4 Summary

Take-home

- Identified grey areas in RATS (e.g., Endorsements)

Take-home

- Identified grey areas in RATS (e.g., Endorsements)
 - can be more precise! Thanks to Dave for pursuing this!

Take-home

- Identified grey areas in RATS (e.g., Endorsements)
 - can be more precise! Thanks to Dave for pursuing this!
- Need for systematic design of attestation protocols

Take-home

- Identified grey areas in RATS (e.g., Endorsements)
 - can be more precise! Thanks to Dave for pursuing this!
- Need for systematic design of attestation protocols
- Open questions

Take-home

- Identified grey areas in RATS (e.g., Endorsements)
 - can be more precise! Thanks to Dave for pursuing this!
- Need for systematic design of attestation protocols
- Open questions
 - How to verify the Verifier?

Take-home

- Identified grey areas in RATS (e.g., Endorsements)
 - can be more precise! Thanks to Dave for pursuing this!
- Need for systematic design of attestation protocols
- Open questions
 - How to verify the Verifier?
 - How to verify the runtime configurations?

Take-home

- Identified grey areas in RATS (e.g., Endorsements)
 - can be more precise! Thanks to Dave for pursuing this!
- Need for systematic design of attestation protocols
- Open questions
 - How to verify the Verifier?
 - How to verify the runtime configurations?
 - Is RATS architecture better than split architecture?

Take-home

- Identified grey areas in RATS (e.g., Endorsements)
 - can be more precise! Thanks to Dave for pursuing this!
- Need for systematic design of attestation protocols
- Open questions
 - How to verify the Verifier?
 - How to verify the runtime configurations?
 - Is RATS architecture better than split architecture?

Take-home

- Identified grey areas in RATS (e.g., Endorsements)
 - can be more precise! Thanks to Dave for pursuing this!
- Need for systematic design of attestation protocols
- Open questions
 - How to verify the Verifier?
 - How to verify the runtime configurations?
 - Is RATS architecture better than split architecture?

Trusted until formally verified!

Key References



Birkholz, Henk et al. *Remote Attestation procedureS (RATS) Architecture*. RFC 9334. Jan. 2023. DOI: 10.17487/RFC9334. URL: <https://www.rfc-editor.org/info/rfc9334>.



Blanchet, Bruno, Vincent Cheval, and Véronique Cortier. "ProVerif with lemmas, induction, fast subsumption, and much more". In: *IEEE Symposium on Security and Privacy (S&P'22)*. Los Alamitos, CA, USA: IEEE Computer Society, May 2022, pp. 205–222. DOI: 10.1109/SP46214.2022.00013.



Intel. *Intel [®] Trust Domain Extensions*. Aug. 2021. URL: <https://cdrdv2.intel.com/v1/dl/getContent/690419>.



Sardar, Muhammad Usama and Christof Fetzer. *Confidential Computing and Related Technologies : A Review*. 2021. URL: https://www.researchgate.net/publication/356474602_Confidential_Computing_and_Related_Technologies_A_Review.



Wired. *Intel Let Google Cloud Hack Its New Secure Chips and Found 10 Bugs*. 2023. URL: <https://www.wired.com/story/intel-google-cloud-chip-security/> (visited on 04/25/2023).

Call to Action

- Bring your expertise:
<https://github.com/CCC-Attestation/formal-spec-TEE>
- Additional information: link here

