# Understanding Trust Relationships in Cloud-Based Confidential Computing

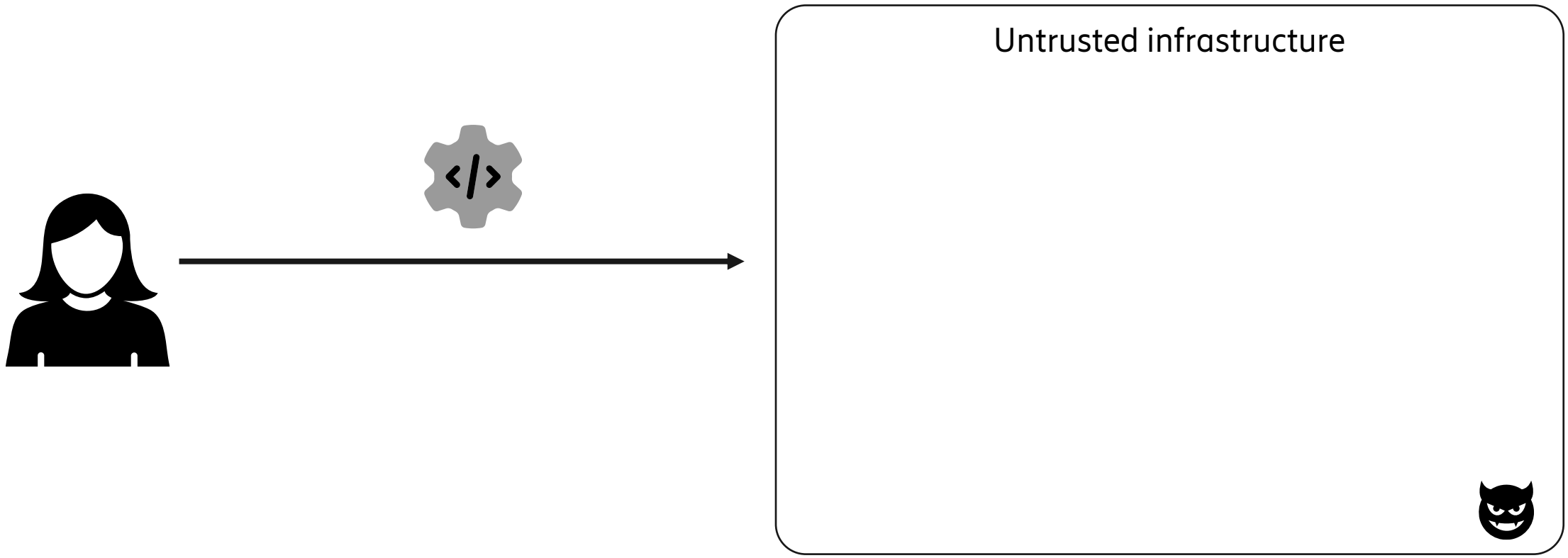
Gianluca Scopelliti, Christoph Baumann, Jan Tobias Mühlberg

*7th Workshop on System Software for Trusted Execution (SysTEX'24)*

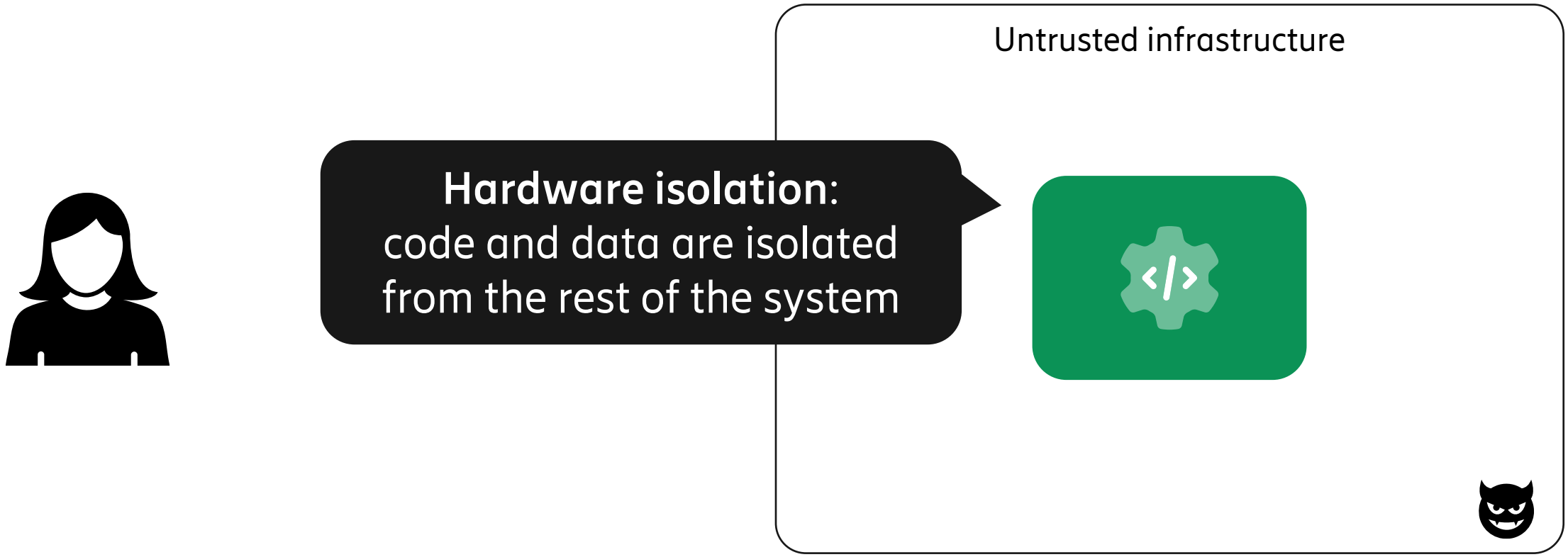
gianluca.scopelliti@ericsson.com



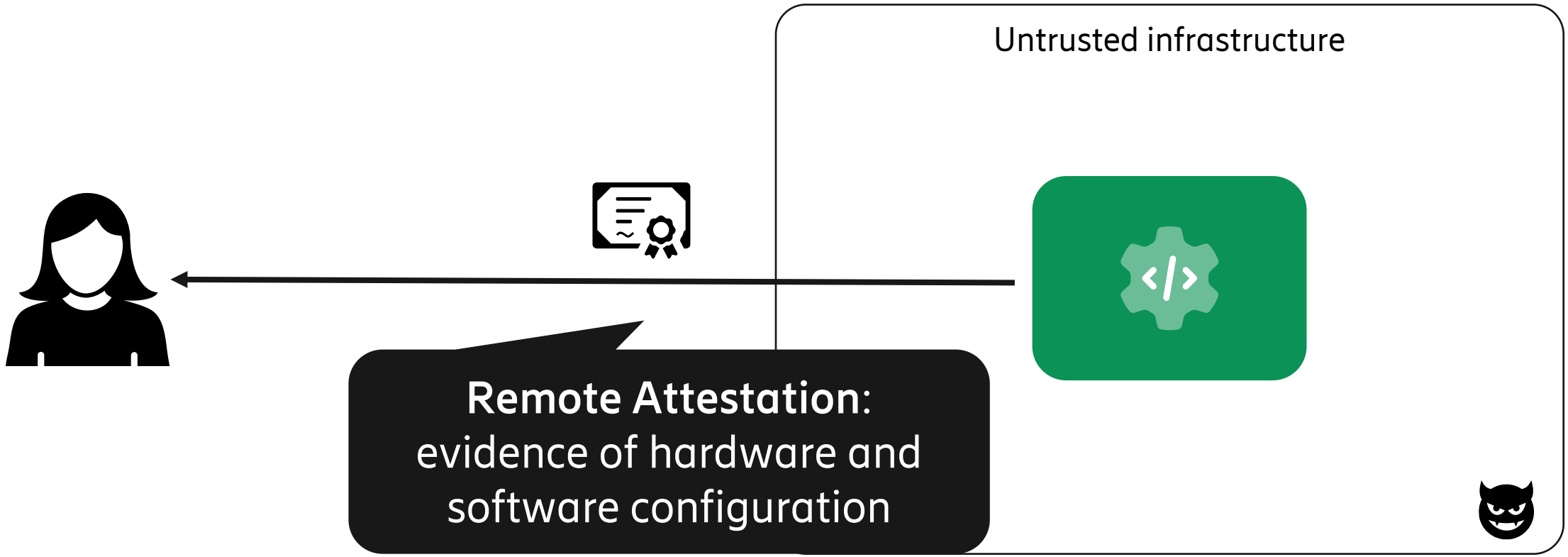
# Confidential Computing (CC)



# Confidential Computing (CC)



# Confidential Computing (CC)



# CC threat model from different perspectives

# CC threat model from different perspectives

## Threat Model – Untrusted Cloud Provider

- Cloud Provider not trusted with **confidentiality** or **integrity**
  - Can run malicious hypervisors (memory replay, memory aliasing, event injects, …)
  - Can run malicious guests (masquerade as other guests, unauthorized memory access…)
  - Can run malicious host applications (unauthorized memory access, …)
  - Can install malicious devices (unauthorized DMA to guest, …)
  - Can misconfigure the platform (insecure platform configuration, …)
  - Can launch guest incorrectly (maliciously altered initial guest image, enable of insecure configuration, …)
- Cloud Provider is trusted for **availability** of the guest.
  - Pushes the run button for the guest repeatedly

# CC threat model from different perspectives

## Threat Model – Untrusted Cloud Provider

- Cloud Provider not trusted with **confidentiality** or **integrity**
  - Can run malicious hypervisors (memory replay, memory aliasing, event injects, …)
  - Can run malicious guests (masquerade as other guests, unauthorized memory access…)
  - Can run malicious host applications (unauthorized memory access, …)
  - Can install malicious devices (unauthorized DMA to guest, …)
  - Can misconfigure the platform (insecure platform configuration, …)
  - Can launch guest incorrectly (maliciously altered initial guest image, enable of insecure configura
- Cloud Provider is trusted for **availability** of the guest.
  - Pushes the run button for the guest repeatedly

## 3. Core Functions of Confidential Computing

The three key functions in confidential computing are listed below:

1. Cryptography-based memory isolation
2. Support for remote attestation
3. CSP is not in the tenant's Trusted Computing Base (TCB)

# CC threat model from different perspectives

## Threat Model – Untrusted Cloud Provider

- Cloud Provider not trusted with **confidentiality** or **integrity**
  - Can run malicious hypervisors (memory replay, memory aliasing, event injects, …)
  - Can run malicious guests (masquerade as other guests, unauthorized memory access…)
  - Can run malicious host applications (unauthorized memory access, …)
  - Can install malicious devices (unauthorized DMA to guest, …)
  - Can misconfigure the platform (insecure platform configuration, …)
  - Can launch guest incorrectly (maliciously altered initial guest image, enable of insecure configura
- Cloud Provider is trusted for **availability** of the guest.
  - Pushes the run button for the guest repeatedly

## 3. Core Functions of Confidential Computing

The three key functions in confidential computing are listed below:

1. Cryptography-based memory isolation
2. Support for remote attestation
3. CSP is not in the tenant's Trusted Computing Base (TCB)

## CC Workload (TCB)

The customer workload, encapsulated inside a Trusted Execution Environment (TEE) includes the parts of the solution that are fully under control and trusted by the customer. The confidential computing workload is opaque to everything outside of the TCB using encryption.



ERICSSON KU LEUVEN ULB

# CC threat model from different perspectives

## Threat Model – Untrusted Cloud Provider

- Cloud Provider not trusted with **confidentiality** or **integrity**
  - Can run malicious hypervisors (memory replay, memory aliasing, event injects, …)
  - Can run malicious guests (masquerade as other guests, unauthorized memory access…)
  - Can run malicious host applications (unauthorized memory access, …)
  - Can install malicious devices (unauthorized DMA to guest, …)
  - Can misconfigure the platform (insecure platform configuration, …)
  - Can launch guest incorrectly (maliciously altered initial guest image, enable of insecure configura
- Cloud Provider is trusted for **availability** of the guest.
  - Pushes the run button for the guest repeatedly

## 3. Core Functions of Confidential Computing

The three key functions in confidential computing are listed below:

1. Cryptography-based memory isolation
2. Support for remote attestation
3. CSP is not in the tenant's Trusted Computing Base (TCB)

**Confidential computing solves the trust problem of the cloud**

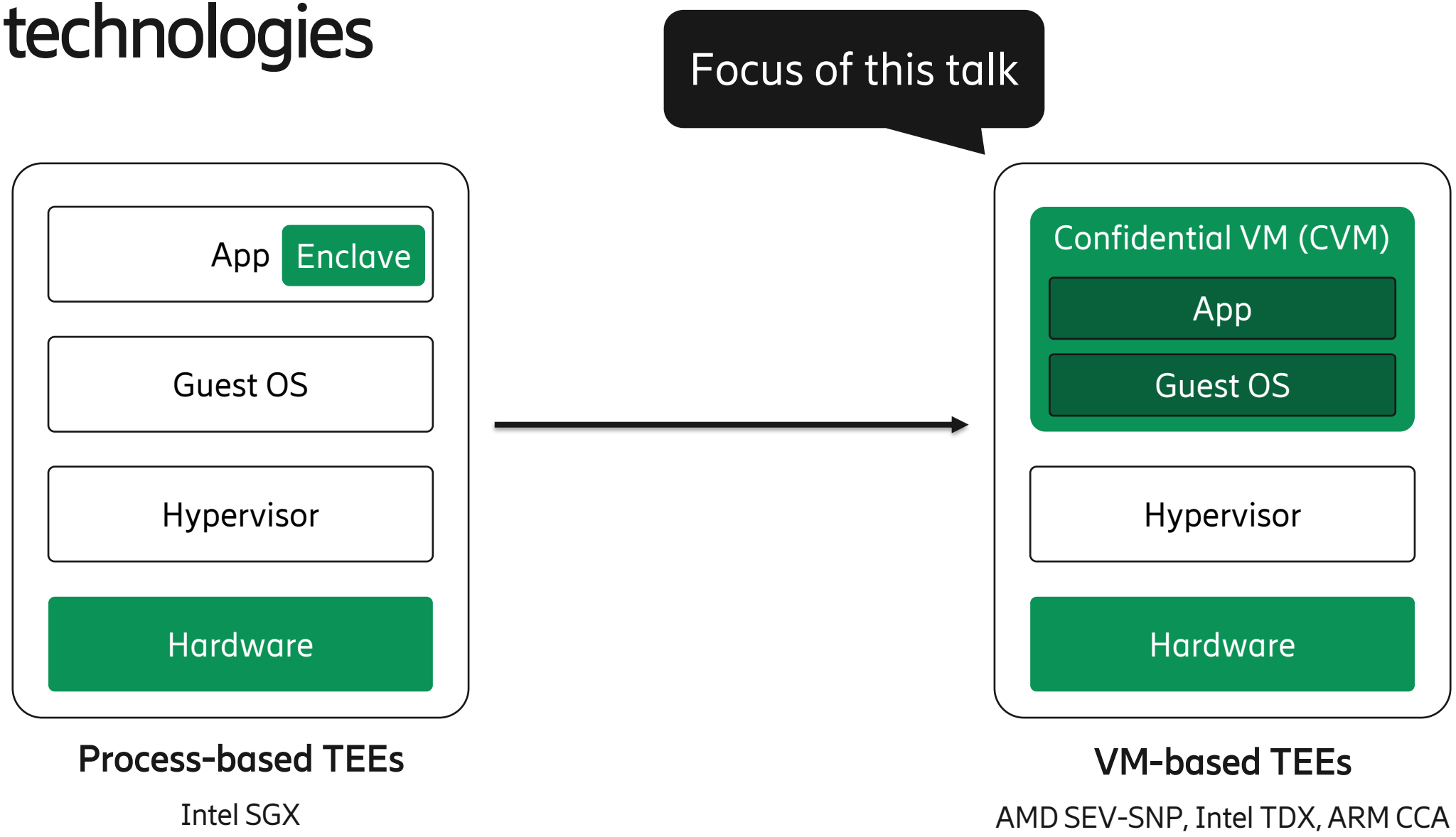
## CC Workload (TCB)

The customer workload, encapsulated inside a Trusted Execution Environment (TEE) inc
that are fully under control and trusted by the customer. The confidential computing workload is opaque to everything outside of the TCB using encryption.



ERICSSON KU LEUVEN ULB

# CC technologies




**Process-based TEEs**

Intel SGX

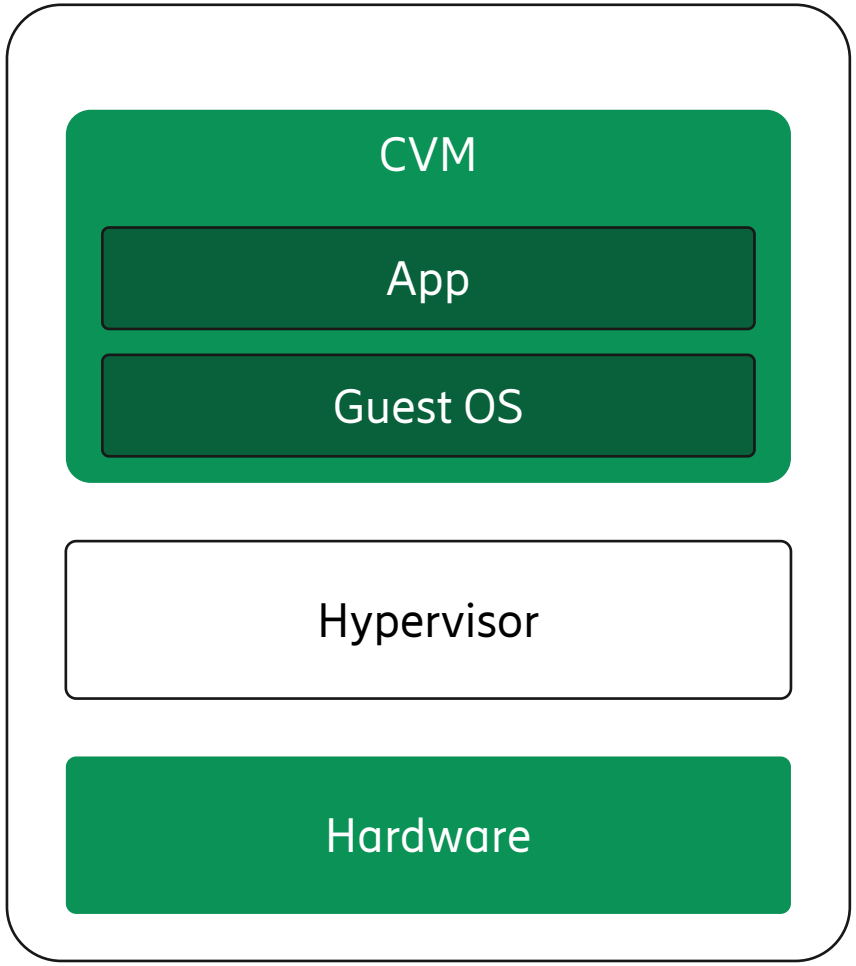
**VM-based TEEs**

AMD SEV-SNP, Intel TDX, ARM CCA

# Why CVMs are becoming so popular?

- “Easy” to use
- “Lift-and-shift”
- Cloud-native (e.g., Kata)
- Performance benefits

Where is the catch? ☺

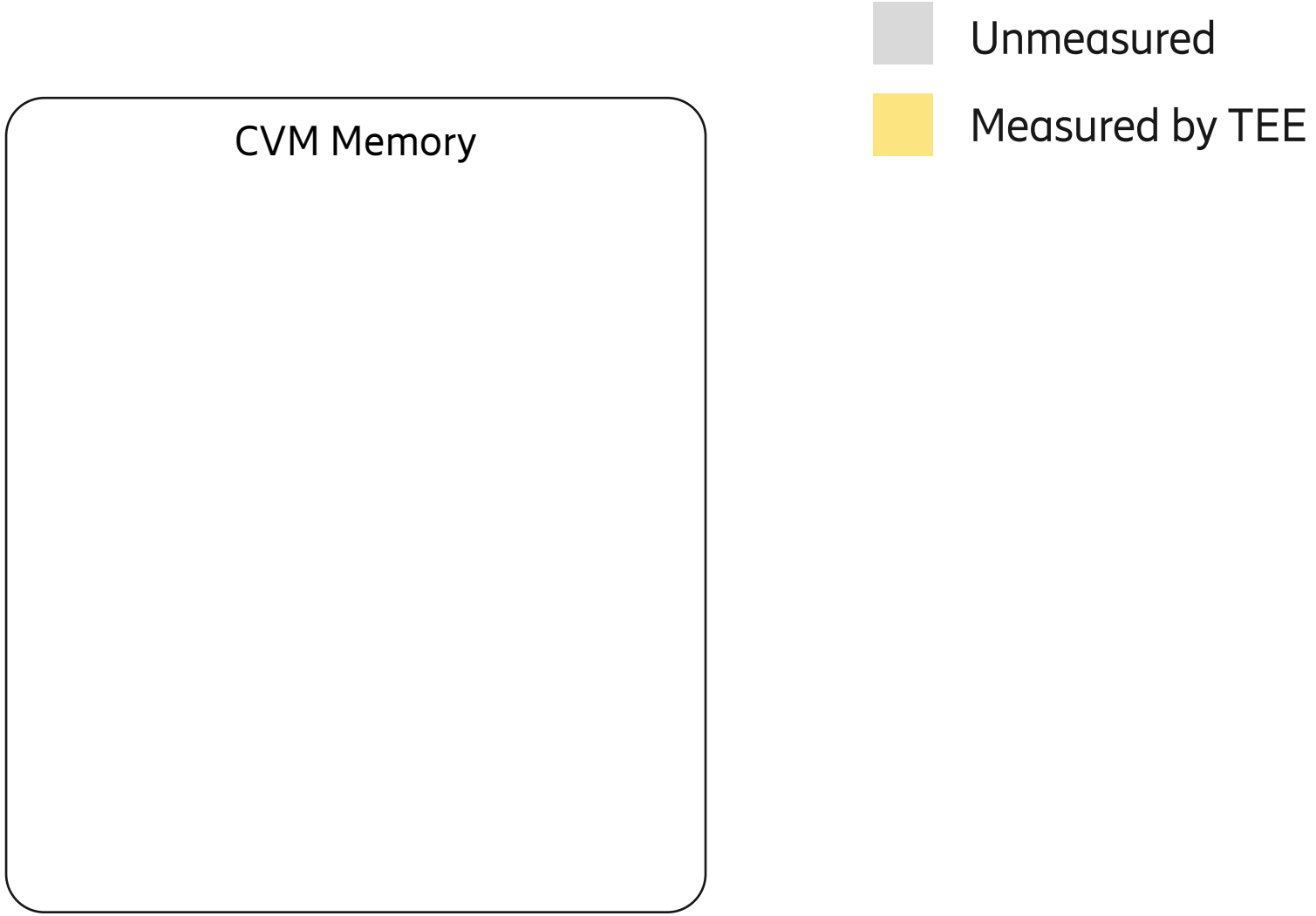




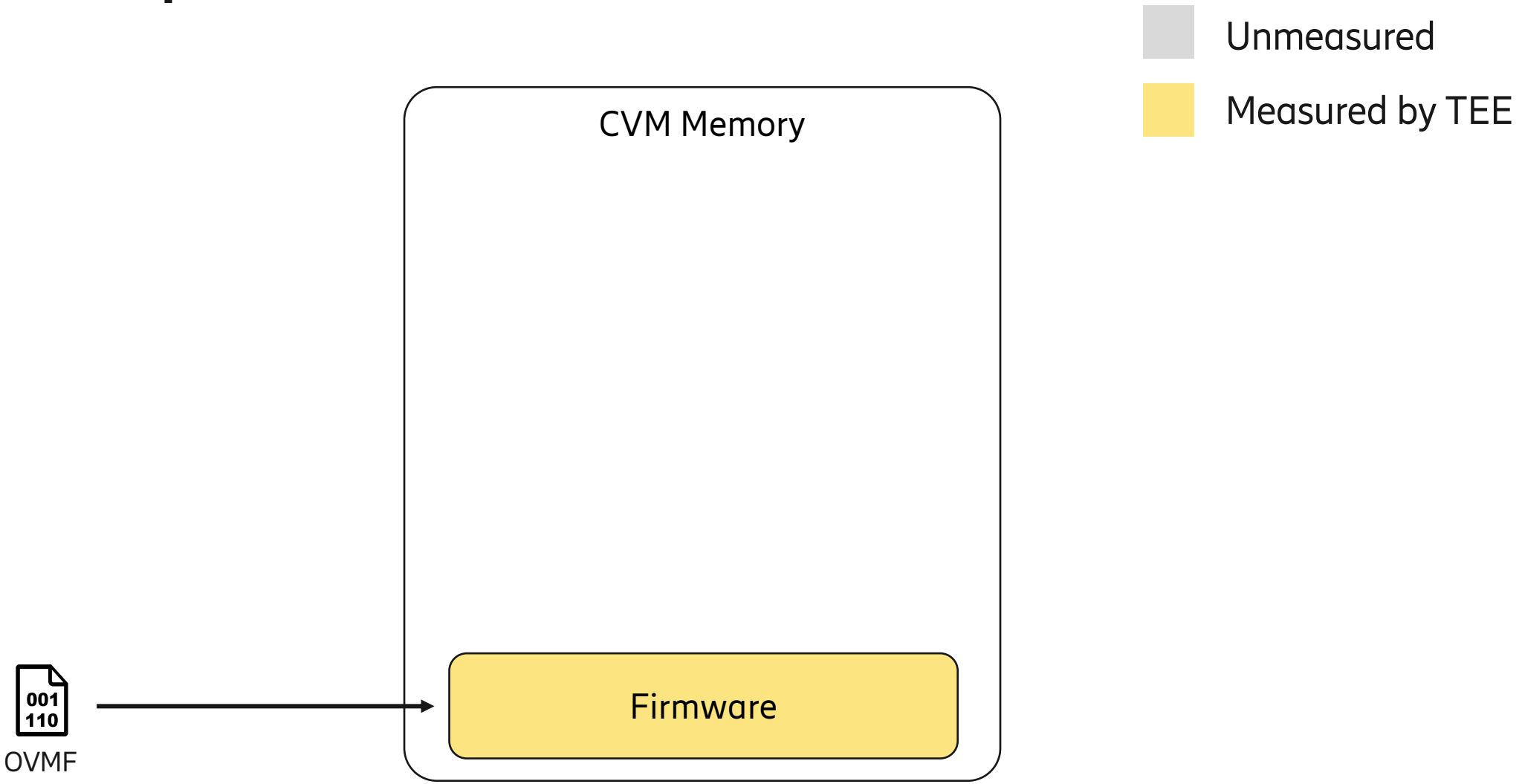

ERICSSON KU LEUVEN ULB

# When using CVMs in the cloud, how is trust established?

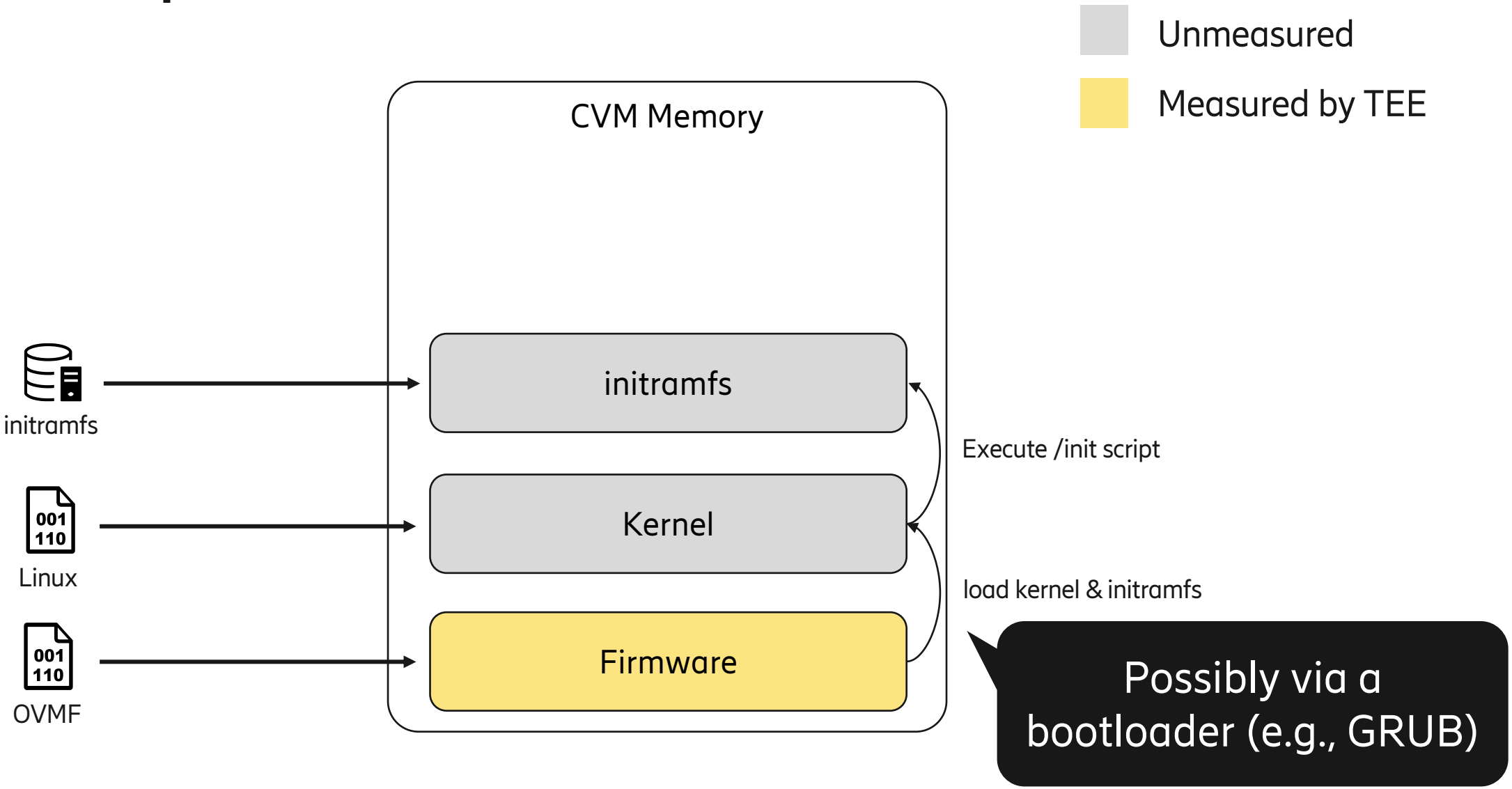
# CVM boot process



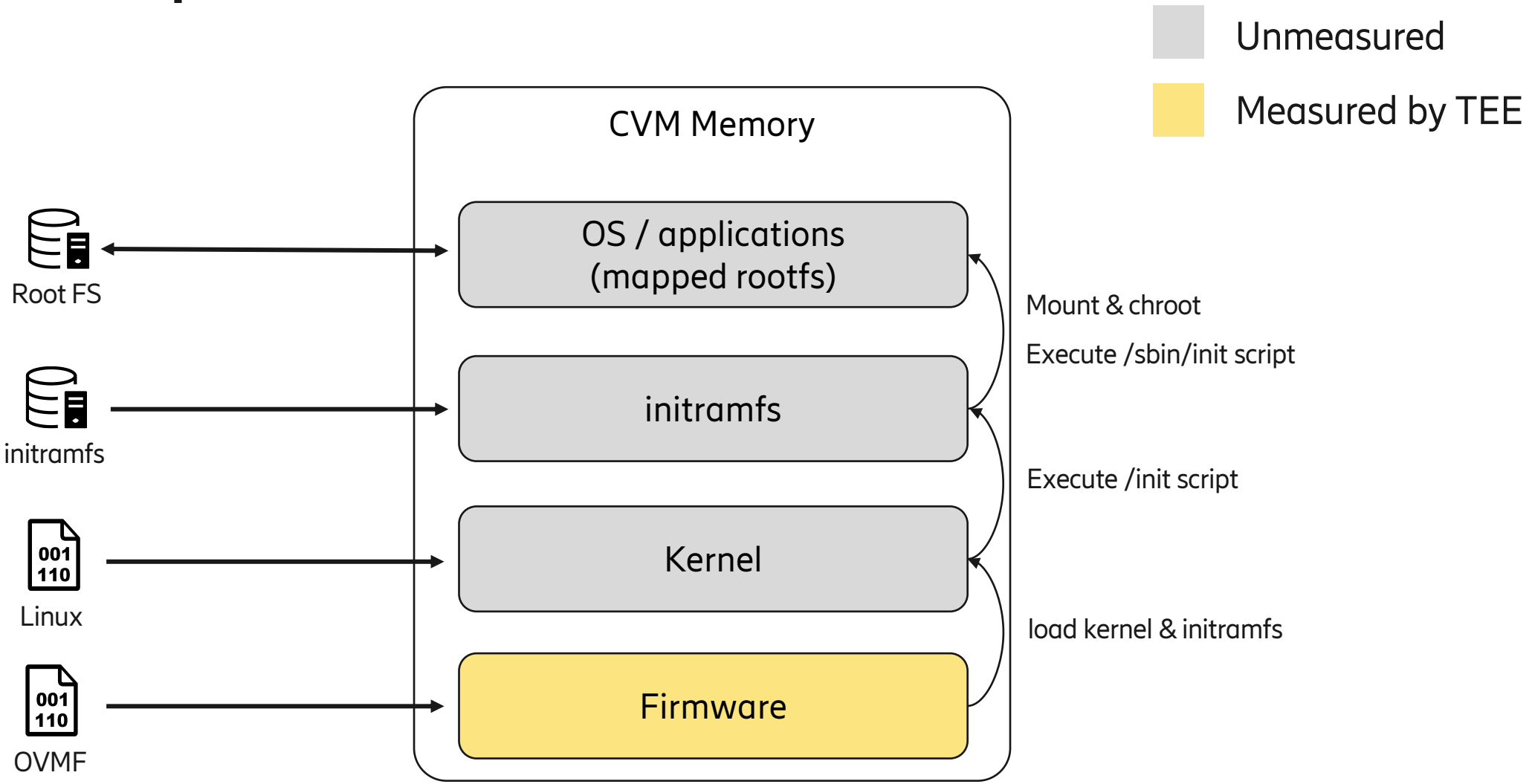
# CVM boot process



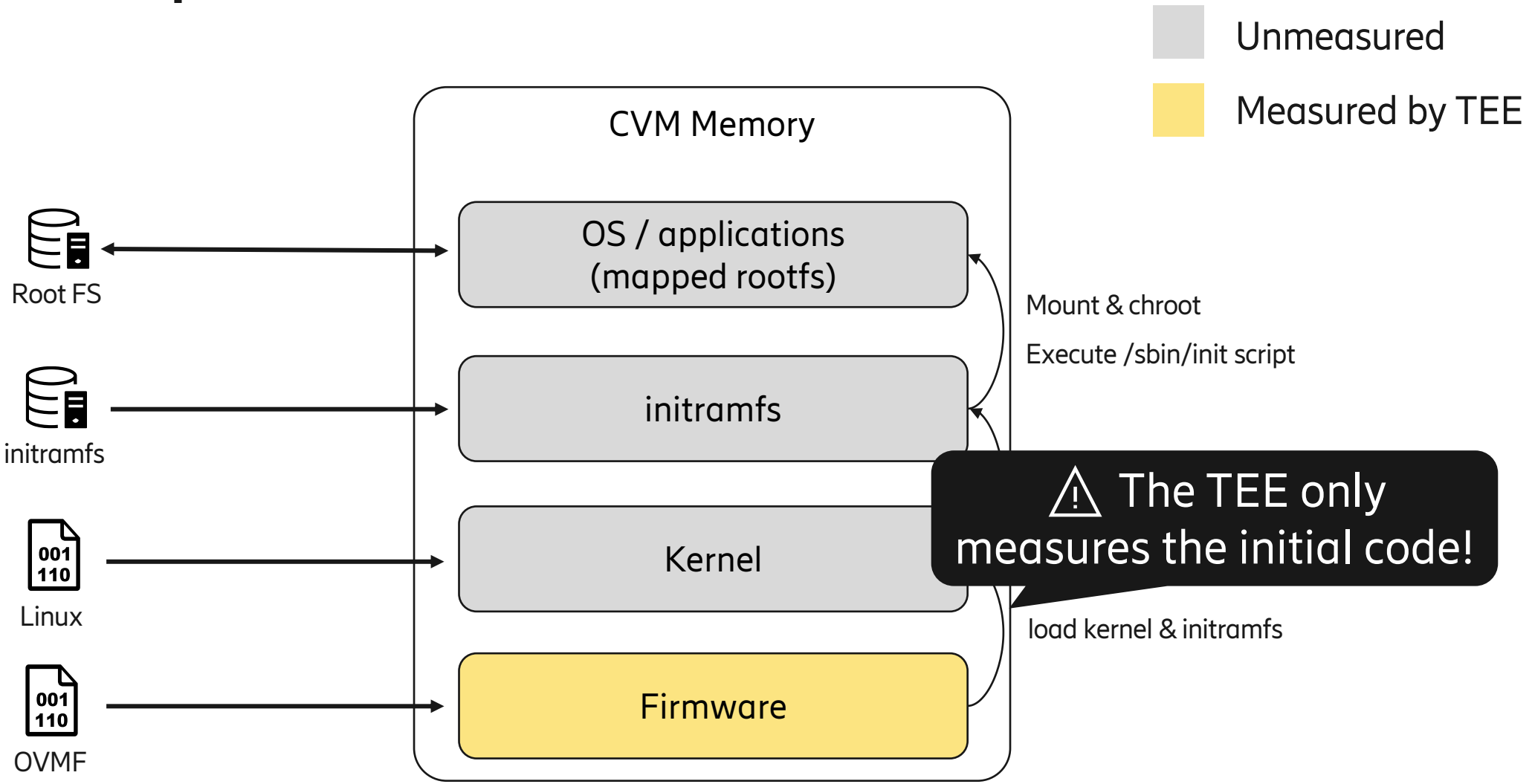
# CVM boot process



# CVM boot process

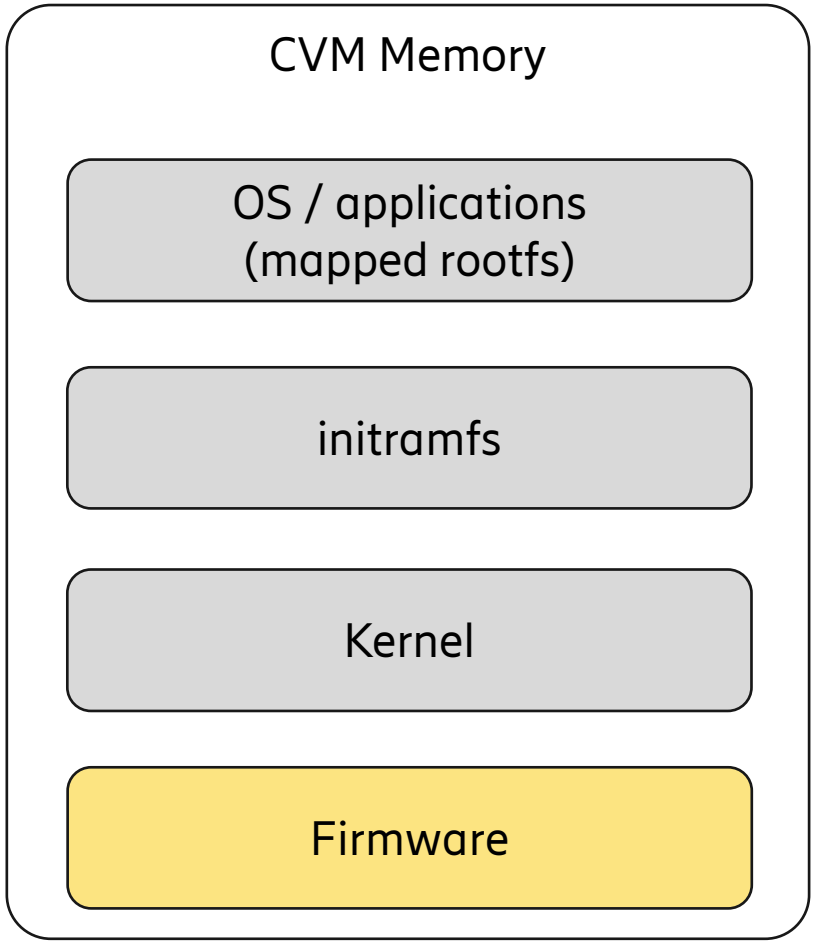


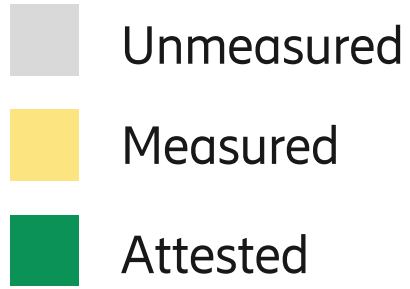
# CVM boot process

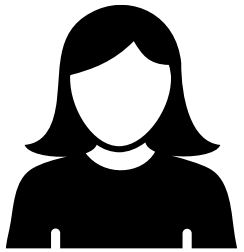


# CVM Attestation

**AL0**: No attestation

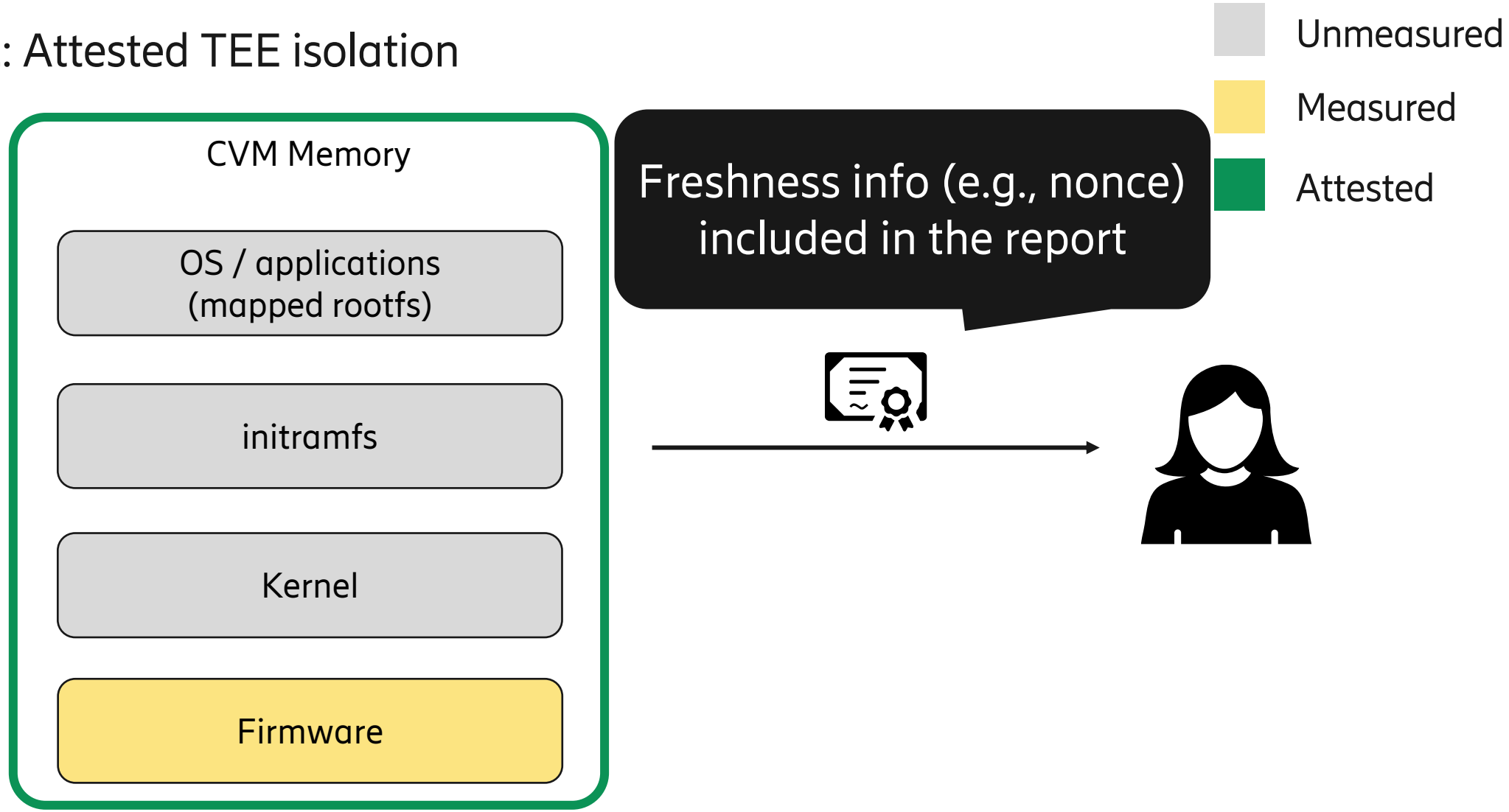


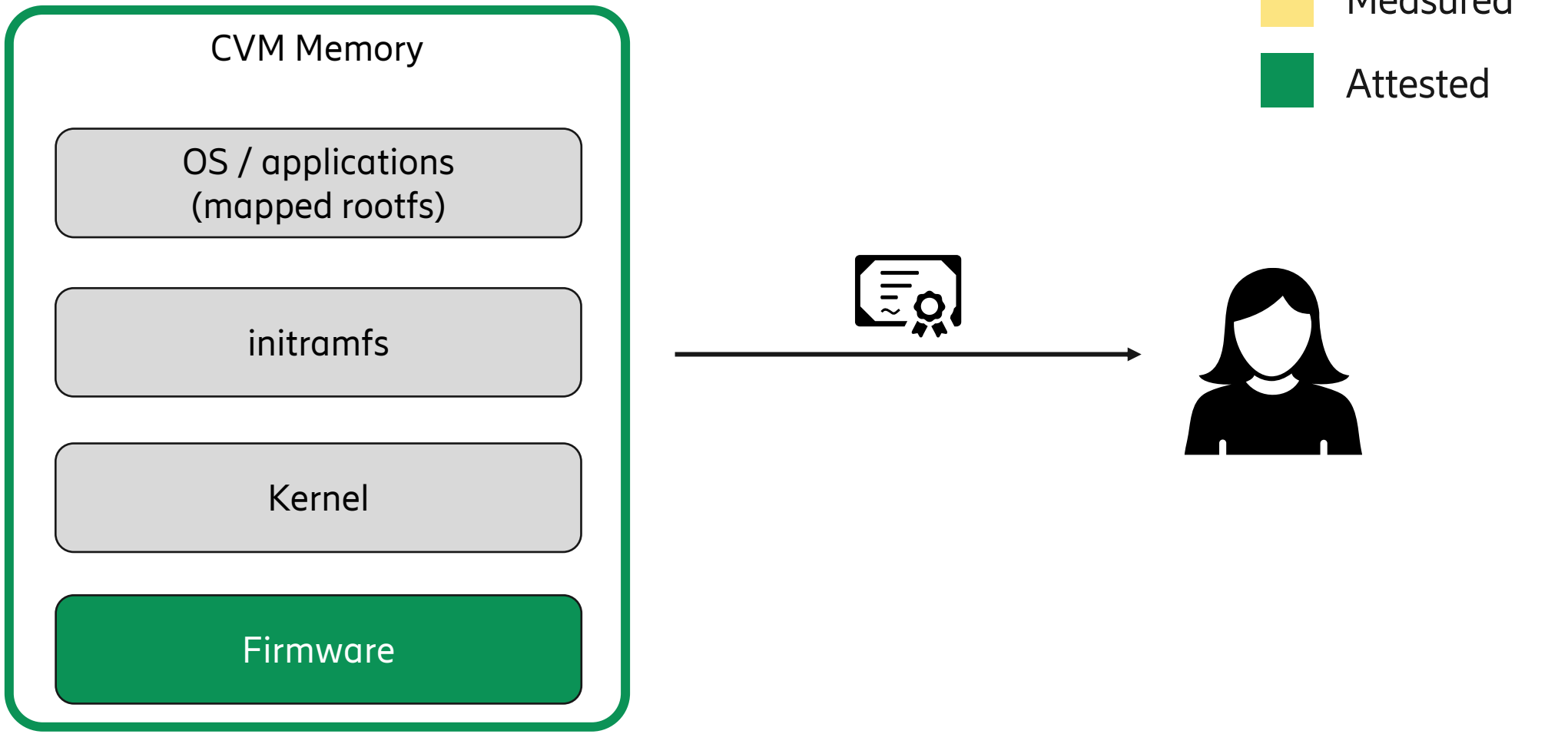
# CVM Attestation

## **AL1**: Attested TEE isolation



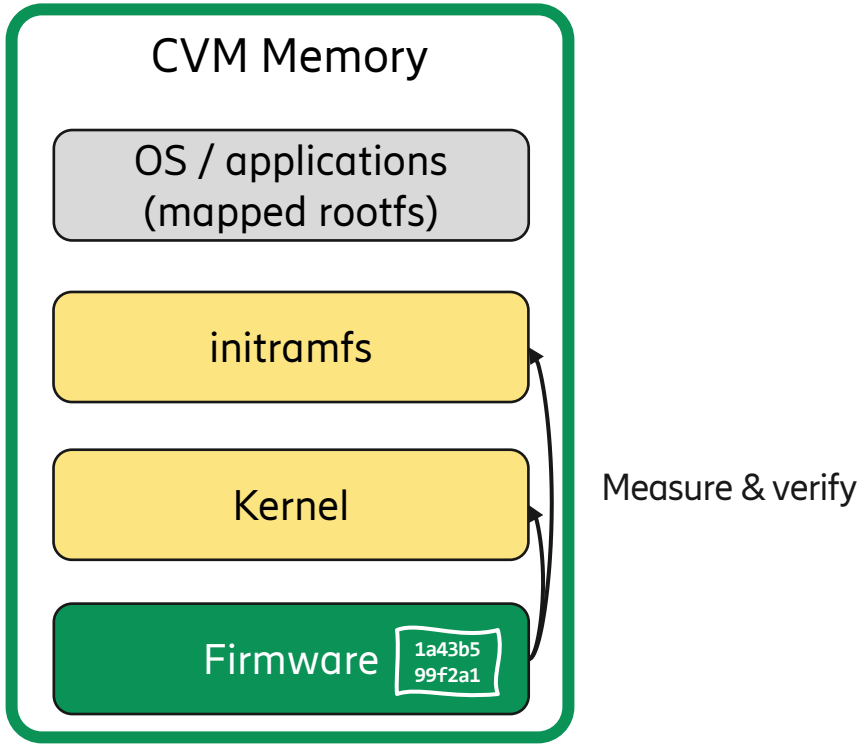
# CVM Attestation

## **AL2**: Measured firmware

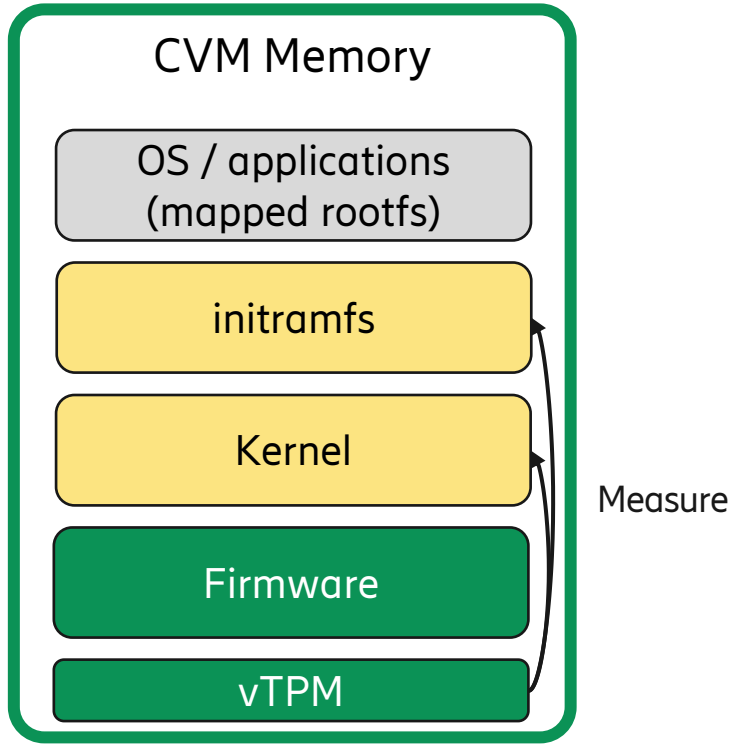


# CVM Attestation

## Extending the measurements to the kernel




**Direct Linux Boot with modified OVMF [1]**



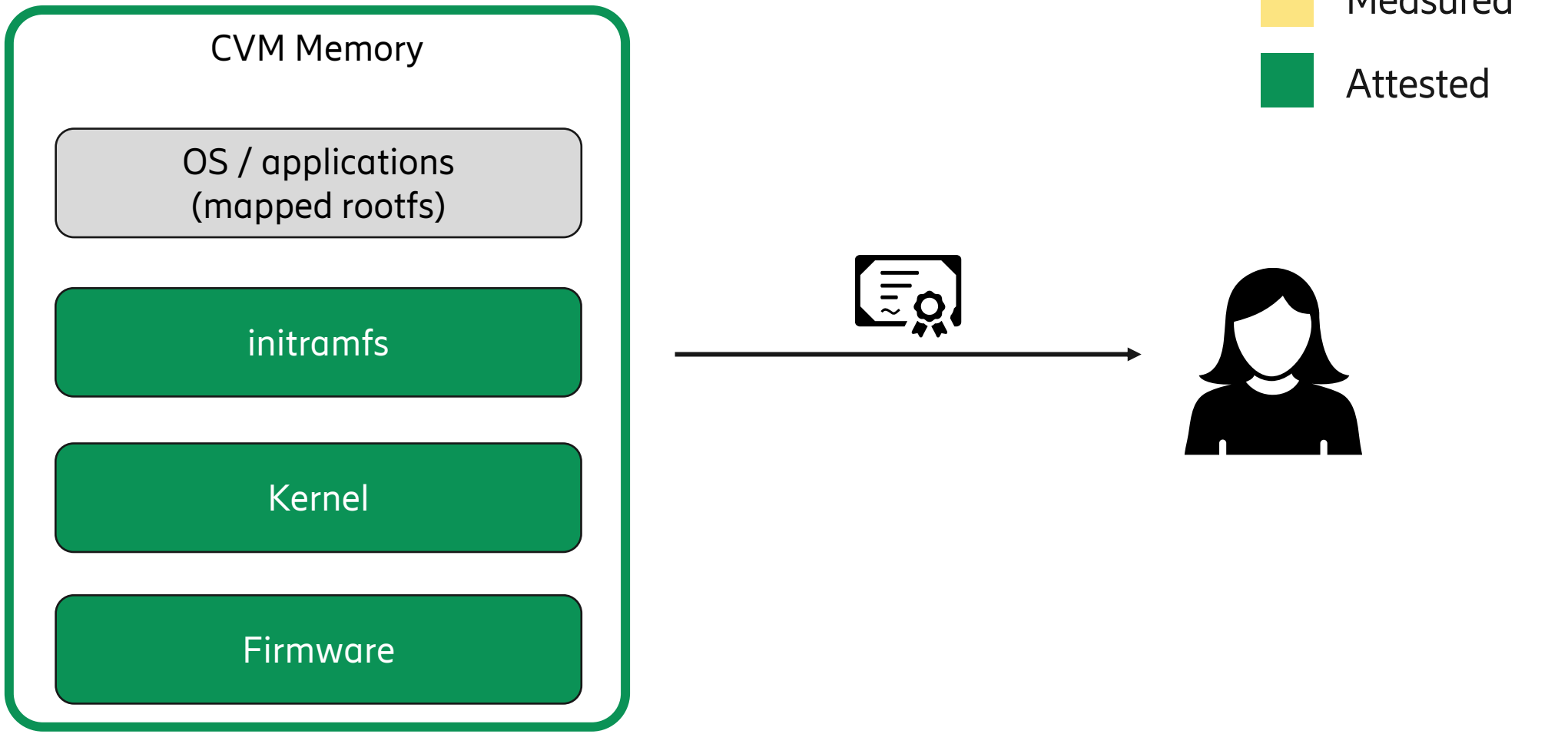

**Virtual TPM (vTPM) [2]**



[1] Dov Murik and Hubertus Franke, "*Securing Linux VM boot with AMD SEV measurement*," KVM forum 2021
[2] Narayanan et al. *"Remote attestation of confidential VMs using ephemeral vTPMs"*, ACSAC'23

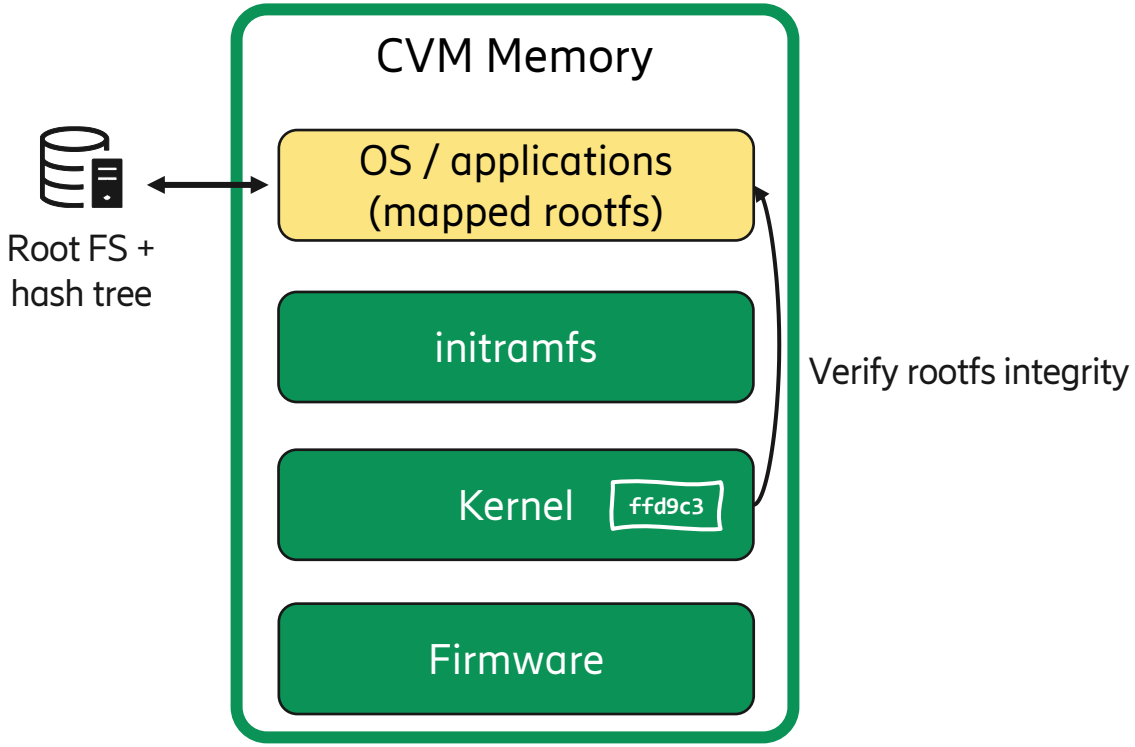
# CVM Attestation

## **AL3**: Measured Kernel

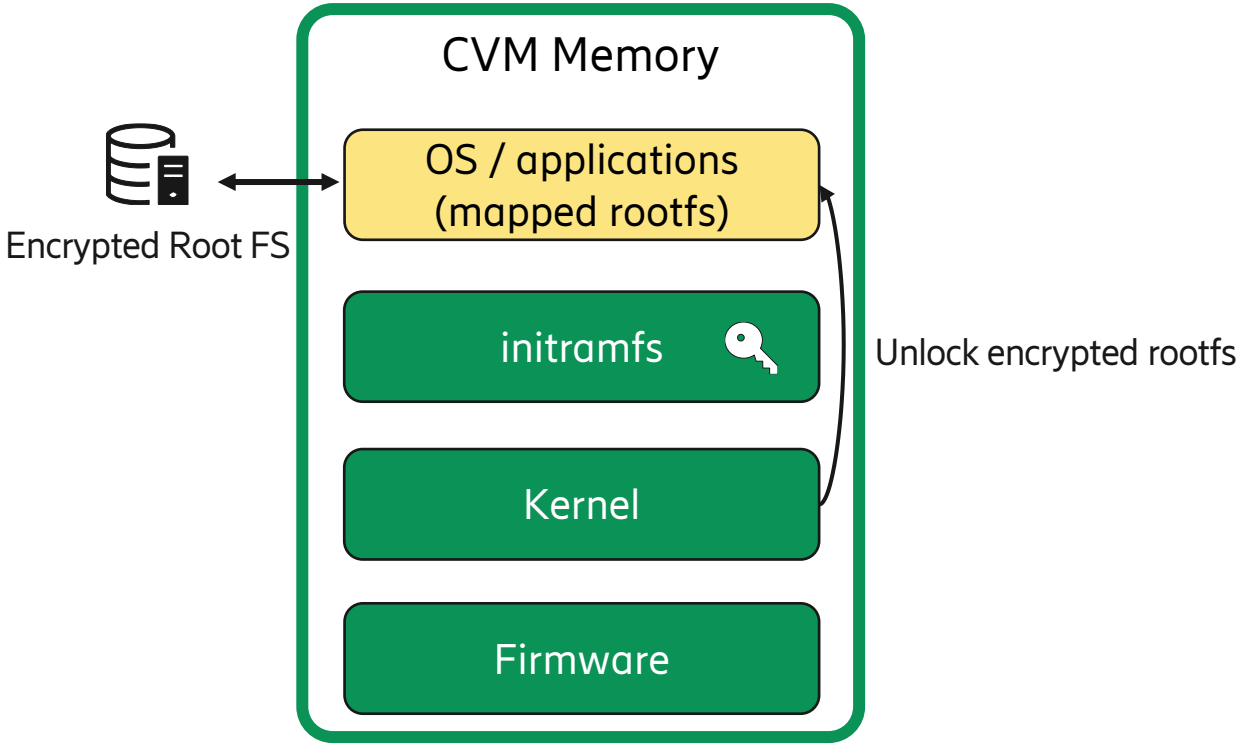


# CVM Attestation

## Protecting the root filesystem




**Integrity protection with dm-verity [3]**

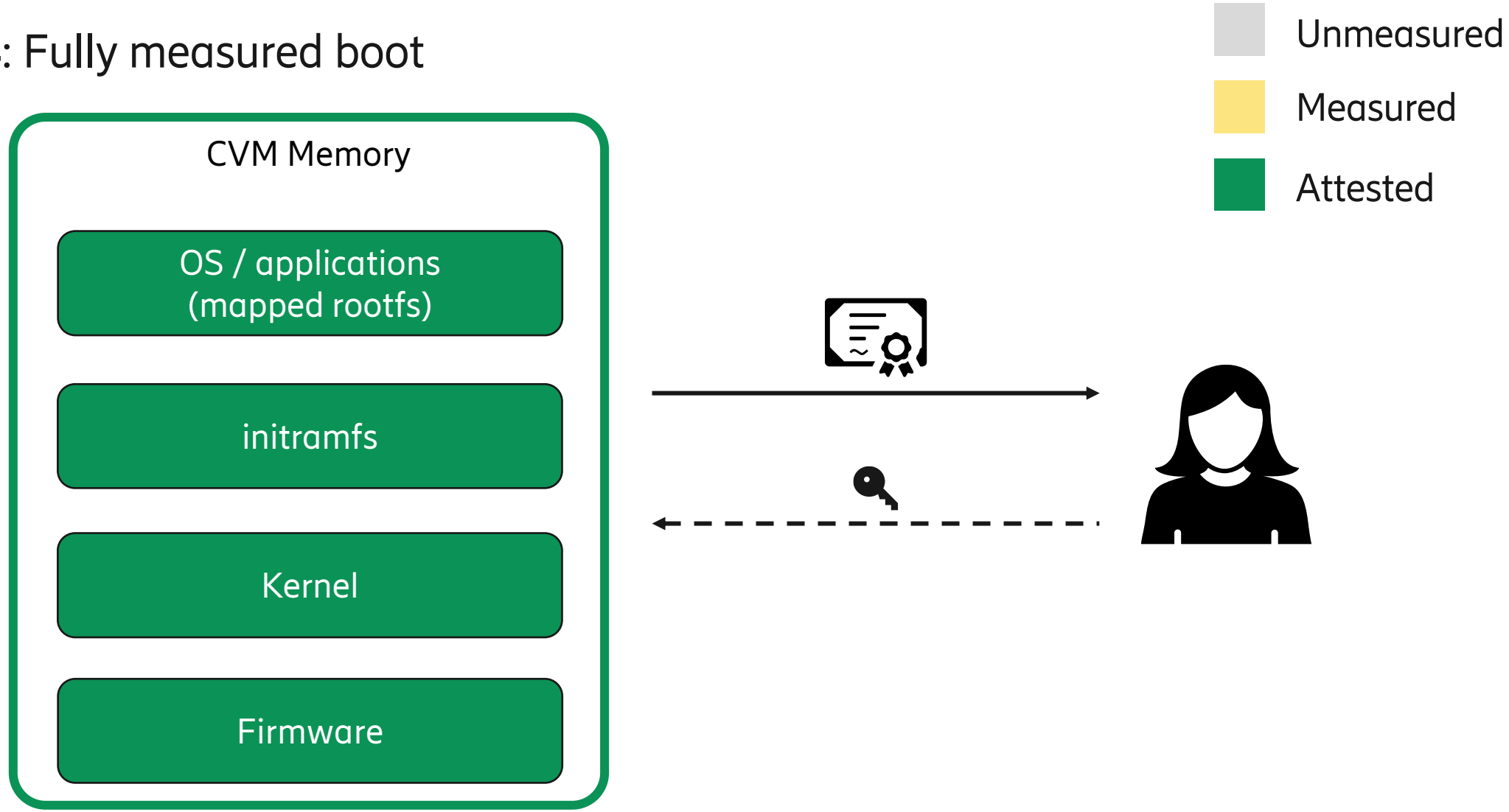



**Authenticated encryption with dm-crypt [4]**

[3] https://docs.kernel.org/admin-guide/device-mapper/verity.html
[4] https://docs.kernel.org/admin-guide/device-mapper/dm-crypt.html

# CVM Attestation

**AL4**: Fully measured boot



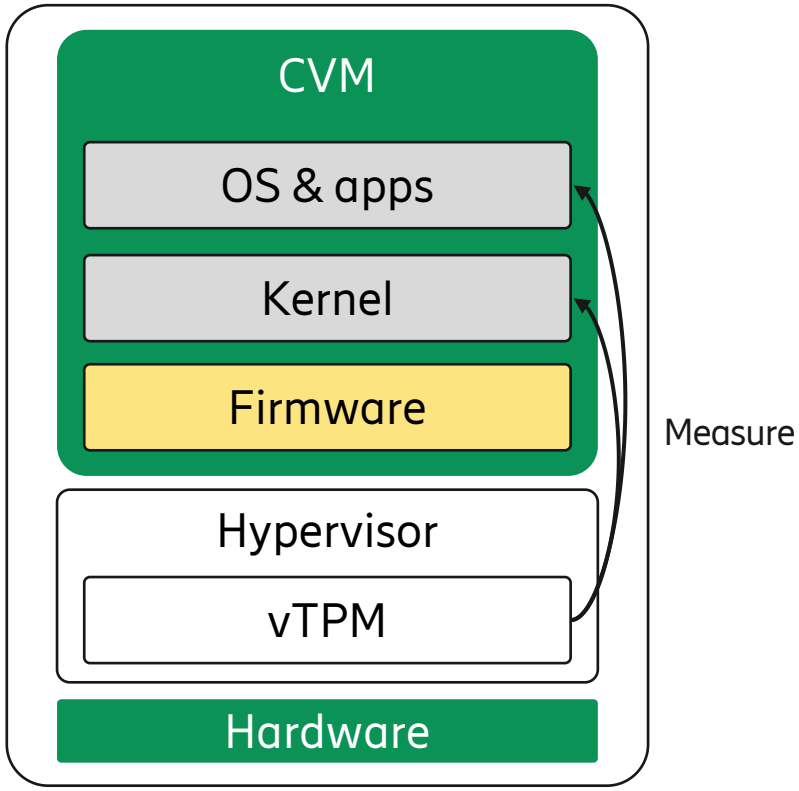
# Deploying CVMs on public clouds

## Running CVMs on demand

- CSP controls the boot process
- Lack of customization options (e.g., using custom firmware)
- Attestation is partially or fully managed by the CSP

# CSP architectures

Measured by TEE




**vTPM in hypervisor layer**

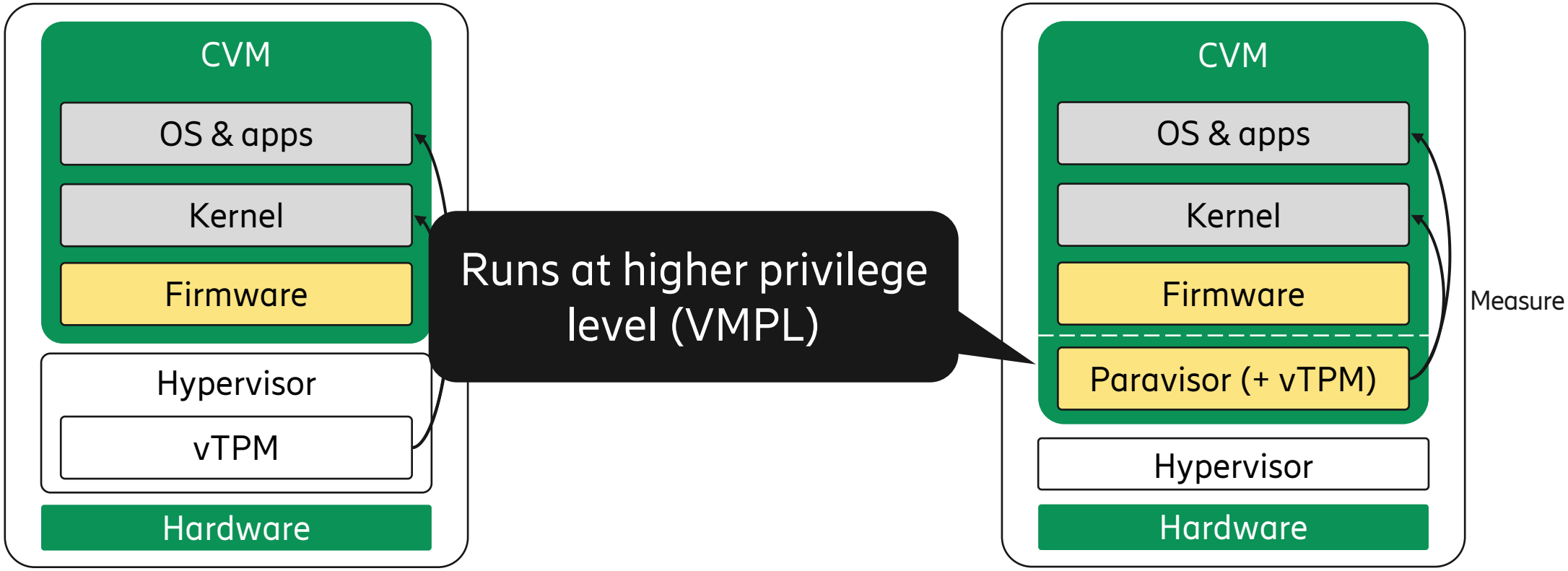
GCP, AWS



ERICSSON KU LEUVEN ULB

# CSP architectures

Measured by TEE




**vTPM in hypervisor layer**

GCP, AWS

**vTPM inside CVM**

Azure

# Cloud landscape

## AMD SEV-SNP offerings on 26 March 2024

● Verifiability without trust ◐ Verifiability with trust

| | AWS | Azure | GCP |
|---|---|---|---|
| TEE | | | |
| Firmware | | | |
| Kernel | | | |
| Root FS | | | |
| **Nominal AL** | | | |
| **Trustworthy AL** | | | |

# Cloud landscape

## AMD SEV-SNP offerings on 26 March 2024

● Verifiability without trust ◐ Verifiability with trust

| | AWS | Azure | GCP |
|---|---|---|---|
| TEE | | | |
| Firmware | | | |
| Kernel | | | |
| Root FS | | | |
| **Nominal AL** | | | |
| **Trustworthy AL** | | | |

AL that can be reached on the CSP infrastructure



ERICSSON KU LEUVEN ULB

# Cloud landscape

## AMD SEV-SNP offerings on 26 March 2024

● Verifiability without trust ◐ Verifiability with trust

| | AWS | Azure | GCP |
|---|---|---|---|
| TEE | | | |
| Firmware | | | |
| Kernel | | | |
| Root FS | | | |
| **Nominal AL** | | | |
| **Trustworthy AL** | | | |

AL that can be reached **without** trusting the CSP



ERICSSON KU LEUVEN ULB

# Cloud landscape

## AMD SEV-SNP offerings on 26 March 2024

● Verifiability without trust ◐ Verifiability with trust

| | AWS | Azure | GCP |
|---|---|---|---|
| TEE | ● | ◐ | ● |
| Firmware | | | |
| Kernel | | | |
| Root FS | | | |
| **Nominal AL** | | | |
| **Trustworthy AL** | | | |

Can only get a “static” attestation report



ERICSSON KU LEUVEN ULB

# Cloud landscape

## AMD SEV-SNP offerings on 26 March 2024

● Verifiability without trust ◐ Verifiability with trust

| | AWS | Azure | GCP |
|---|---|---|---|
| TEE | ● | ◐ | ● |
| Firmware | ● | ◐ | ◐ |
| Kernel | | | |
| Root FS | | | |
| **Nominal AL** | | | |
| **Trustworthy AL** | | | |

Open-source, reproducible builds*

* https://github.com/aws/uefi

# Cloud landscape

## AMD SEV-SNP offerings on 26 March 2024

● Verifiability without trust ◐ Verifiability with trust

| | AWS | Azure | GCP |
|---|---|---|---|
| TEE | ● | ◐ | ● |
| Firmware | ● | ◐ | ◐ |
| Kernel | ◐ | ◐ | ◐ |
| Root FS | | | |
| **Nominal AL** | | | |
| **Trustworthy AL** | | | |

Kernel measurements made by CSP-managed vTPM



ERICSSON KU LEUVEN ULB

# Cloud landscape

## AMD SEV-SNP offerings on 26 March 2024

● Verifiability without trust ◐ Verifiability with trust

| | AWS | Azure | GCP |
|---|---|---|---|
| TEE | ● | ◐ | ● |
| Firmware | ● | ◐ | ◐ |
| Kernel | ◐ | ◐ | ◐ |
| Root FS | ◐ | ◐ | ◐ |
| **Nominal AL** | | | |
| **Trustworthy AL** | | | |

# Cloud landscape

## AMD SEV-SNP offerings on 26 March 2024

● Verifiability without trust ◐ Verifiability with trust

| | AWS | Azure | GCP |
|---|---|---|---|
| TEE | ● | ◐ | ● |
| Firmware | ● | ◐ | ◐ |
| Kernel | ◐ | ◐ | ◐ |
| Root FS | ◐ | ◐ | ◐ |
| **Nominal AL** | 4 | 4 | 4 |
| **Trustworthy AL** | 2 | 0 | 1 |

# Understanding Trust Relationships in Cloud-Based Confidential Computing

- Attesting the full CVM boot chain is *challenging*
  - Just verifying the attestation report is not enough
  - Today, trust in the CSP is still required
- Trust relationships need to be properly clarified
  - Gap between TEE threat model and current offerings
- Lots of work still to be done, but we're moving in the right direction

Gianluca Scopelliti, Christoph Baumann, Jan Tobias Mühlberg

7th Workshop on System Software for Trusted Execution (SysTEX'24)

gianluca.scopelliti@ericsson.com

ERICSSON

KU LEUVEN

DistriNet

ULB CYBERSECURITY RESEARCH CENTER

# Nested virtualization on Azure

- Allows running a CVM from a normal VM (“CVM-in-VM”)
- Customer has full control over the nested hypervisor and the boot process
- Trustworthy AL4 possible!
- Today, still in preview and only usable in AKS with Confidential Containers