

Mind the Gap: Where TEE Attestations Fall Short and Why We Need Proof of Cloud

Filip Rezabek

Technical University of Munich & Flashbots
Munich, Germany
filip.rezabek@tum.de

Jonathan Passerat-Palmbach
Flashbots & Imperial College London
London, UK
jonathan@flashbots.net

Abstract

Confidential VMs carry a fundamental blind spot: they certify *what* code is running but not *where*. TEE vendors exclude physical-access adversaries from their threat model, implicitly assuming the machine resides in a trustworthy data center. We argue this implicit trust must be made explicit and verifiable through *Proof of Cloud*: a generic concept for a cryptographic proof that the attesting hardware resides in a facility operated by an identifiable provider. We survey two recent protocols: Flashbots’ DCEA and Intel’s POE. Neither alone suffices: POE provides lean hardware provenance but leaves host stack integrity out of scope, while DCEA solves it at a deeper integration cost. We identify the residual trust supply chain problem that both approaches inherit.

Keywords: TEE, Attestation, Proof of Cloud

1 Introduction

Confidential VMs (CVMs) running inside Trusted Execution Environments (TEEs) like Intel Trust Domain Extensions (TDX) have gained significant adoption across various industries spanning from confidential AI to Decentralized Finance (DeFi) scenarios [3, 4]. This wide adoption of TEEs could be hindered due to the latest series of attacks against the range of modern TEEs powering CVMs [5, 12]. While we cannot ignore these attacks, it is worth examining the conditions required to successfully mount them in the real world.

An attacker with physical access can closely monitor and control the hardware interacting with the TEE, aiming to extract the attestation keys that underpin its integrity guarantees. A successful attacker would then be able to forge attestations outside of a TEE and masquerade a transparent environment as a CVM. This is possible because a TEE attestation report certifies a CPU model, microcode revision, and measured launch state, but provides no evidence of the processor’s physical location. This gap in attestations enables a determined operator to produce a valid attestation while hosting the workload on hardware in an uncontrolled environment rather than an established data center [11].

While these attacks are undeniably practical and cost-effective, they are considered out of scope by TEE vendors. The shift towards CVMs essentially morphed TEEs into a cloud-only technology, thus conveniently ruling out attackers with physical access from the threat model, at the cost of

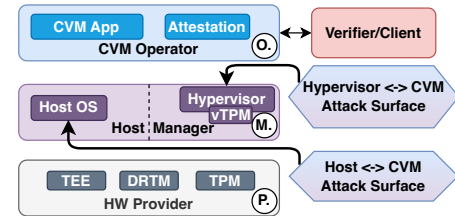


Figure 1. Roles in cloud deployments. The host-to-CVM interface is an adversarial attack surface: the hypervisor, vTPM, and host-guest communication channels are controlled by an entity the workload owner has no reason to trust.

trusting the cloud provider. Despite this significant change, Remote Attestations (RAs) have remained largely unchanged. RAs let a verifier confirm *what* code is running, rooting the authenticity of the firmware version, the guest kernel, the CVM image in hardware, but they provide no evidence of *where* that code is running [11]. Recent proposals such as Intel Platform Ownership Endorsement (POE) [7] and Data Center Execution Assurance (DCEA) [10] tackle this issue by offering what we define as *Proof of Cloud* guarantees.

We define Proof of Cloud as a cryptographic proof that a confidential workload runs on hardware within a known infrastructure environment, binding guest attestation to the provider’s root of trust so that remote verifiers can confirm execution on trusted data center hardware. In the Proof of Cloud’s threat model, we assume the host software stack (OS, hypervisor, Virtual TPM (vTPM)) is adversarial, but CPU/TPM hardware roots and their supply chains are trusted for attestation evidence, while broader supply-chain guarantees remain out of scope. The cloud provider is trusted for physical integrity and correct certificate issuance, but not for the confidentiality of any software-visible state. Figure 1 covers the individual roles, HW Provider (P) and potential attacks between Host Manager (M), and CVM operator (O).

In a conventional cloud deployment, the roles of the (O), (M), and (P) typically collapse into the same entity, or into parties bound by commercial agreements backed by legal recourse. The Host-to-CVM interface is an internal boundary in this setting, not an adversarial one. Some cloud providers already expose partial provenance signals within their network: Microsoft Azure Attestation (MAA) validates TEE evidence and issues attestation tokens, while on Google Cloud Platform

(GCP), vTPM Endorsement Key (EK) certificates are signed by Google’s subordinate Certificate Authority (CA).

These mechanisms suffice when the verifier has a contractual relationship with the provider. However, these approaches are vendor-specific, do not generalize across clouds, and do not cover bare-metal deployments, which is a hard requirement for latency-sensitive workloads. They break down entirely in settings where the three roles are filled by independent, mutually untrusted, and potentially pseudonymous parties, as is common in permissionless systems.

2 Proof of Cloud: Specific Solutions

Cloud TEE deployments already implicitly trust the data center operator not to mount physical attacks as part of the threat models of Intel TDX and AMD Secure Encrypted Virtualization (SEV) [1, 6]. Proof of Cloud makes this implicit assumption explicit and verifiable, turning the provider into an accountable participant in the attestation chain rather than an invisible trust assumption. As long as TEEs cannot defend against physical adversaries, a verifiable signal that a specific machine is owned and operated by a known data center provider will be part of the threat model. Proof of Cloud is thus better discussed as a concept rather than as a single protocol.

2.1 Flashbots’ DCEA

DCEA [10] is one of the possible instantiations for Proof of Cloud. It relies on two parallel roots of trust: one from the TEE manufacturer and one from the infrastructure owner via a TPM/vTPM. It binds them through overlapping measurements between the vTPM’s Platform Configuration Registers (PCRs) and the CVM’s Runtime Extendable Measurement Registers (RTMRs). For a bare-metal deployment where the host stack is operated by an untrusted party, DCEA relies on Dynamic Root of Trust Measurement (DRTM), e.g., Intel Trusted Execution Technology (TXT), to bind the TPM to vTPM and enforce the correct hypervisor behavior.

DCEA’s defenses combine: the CVM-embedded Attestation Key (AK) hash exposes cross-machine quotes, PCR-sealed keys prevent off-platform use, nonce freshness defeats replay, and timing bounds on concurrent challenges make long proxy paths observable. This layered approach allows DCEA to mitigate proxy attacks. DCEA goes beyond by offering a second strong guarantee since it enforces the hypervisor’s integrity via TXT. DCEA generalizes to any system with comparable primitives to Intel TDX and TXT.

2.2 Intel POE

Intel POE is a recent instantiation of Proof of Cloud that binds platform attestation to verified physical ownership. Here, the Cloud Provider acts as the trusted authority known as the Endorser. During the supply chain phase, the provider collects the unique identifier of their fleet. When a machine is

installed, it generates a factory-resettable Platform Instance Identity (PIID). The provider then signs a POE “token” certifying that this specific PIID is part of their inventory.

When a CVM generates an attestation quote, the hardware automatically embeds the PIID. The verifier cross-references the PIID in the quote with the provider-signed POE. If both PIIDs match, the verifier accepts the cryptographic evidence that the workload is running on hardware previously endorsed by that cloud provider. POE does not consider the Host-to-CVM interface, leaving room for potential hypervisor threats. The design is general and could thus, in theory, be adopted by other TEE manufacturers.

3 Summary and Open Challenges

Proof of Cloud guarantees that the hardware producing an attestation belongs to a named data center operator’s inventory at a specific point in time. POE and DCEA are instantiations of this concept, differing in scope. While POE is lean, DCEA extends this with integrity evidence over lower-level layers, notably the hypervisor, thereby mitigating attacks such as SNPeek [13] that instrument the hypervisor while remaining within the cloud TEE threat model. In conventional cloud deployments, the operator, host manager, and hardware provider are typically unified or contractually bound, which may make POE’s guarantees sufficient. In permissionless environments, these roles are mutually distrustful, exposing inter-domain interfaces as adversarial surfaces, where DCEA’s coverage becomes more relevant.

Regardless of instantiation, Proof of Cloud remains a point-in-time snapshot that implicitly trusts the provider’s inventory records and maintenance processes. This leaves open a supply chain gap. Cloud infrastructure is a layered stack: hardware may be manufactured by one entity, assembled in a facility operated by another, managed through software maintained by a third, and offered to customers through further intermediaries. Each link introduces a trust boundary and potential information asymmetries. Concretely, a compromised or colluding data center operator could remove an endorsed machine from its facility while the machine retains a valid PIID. Until the POE token expires or the provider actively updates its inventory, that machine can continue to produce valid Proof of Cloud attestations from an uncontrolled location. Mitigating such supply chain attacks will likely require moving toward Trustless TEEs [8, 9] that reduce the importance of hardware manufacturers in establishing hardware roots of trust. A suggestion is to rely on Physical Unclonable Functions for root key generation, focus on a verifiable hardware and software stack, and minimize trust in individual parties. A contrasting model is vertical integration, where a single entity controls hardware manufacturing, provisioning, and runtime integrity. Apple’s hardware integrity lifecycle illustrates this approach, although it does not enable independent third-party verification [2].

T-TEE could propose a natural extension to inventory-based ownership proofs with continuous runtime assurance, an open direction we believe warrants further investigation by the community.

References

- [1] 2020. *SEV-SNP: Strengthening VM Isolation with Integrity Protection and More*. Technical Report. Advanced Micro Devices, Inc. White paper.
- [2] Apple Inc. [n.d.]. Hardware Integrity in Private Cloud Compute. <https://security.apple.com/documentation/private-cloud-compute/hardwareintegrity>. Accessed: 2026-02-01.
- [3] BuilderNet. 2024. *Introducing BuilderNet*. BuilderNet. <https://buildernet.org/blog/introducing-buildernet> Accessed: 2026-02-17.
- [4] Raymond Cheng, Fan Zhang, Jernej Kos, Warren He, Nicholas Hynes, Noah Johnson, Ari Juels, Andrew Miller, and Dawn Song. 2019. Ekiden: A Platform for Confidentiality-Preserving, Trustworthy, and Performant Smart Contracts. In *2019 IEEE European Symposium on Security and Privacy (EuroS&P)*. 185–200. doi:10.1109/EuroSP.2019.00023
- [5] Jalen Chuang, Alex Seto, Nicolas Berrios, Stephan van Schaik, Christina Garman, and Daniel Genkin. 2026. TEE.fail: Breaking Trusted Execution Environments via DDR5 Memory Bus Interposition. In *47th IEEE Symposium on Security and Privacy (IEEE S&P '26)*. IEEE Computer Society. <https://tee.fail>
- [6] Intel. 2024. *intel/tdx-module*. (Accessed on 05/10/2024).
- [7] Intel Corporation. [n.d.]. Platform Ownership Endorsements for Confidential Computing. <https://www.intel.com/content/www/us/en/developer/articles/technical/software-security-guidance/technical-documentation/platform-ownership-endorsements.html>. Updated Dec. 12, 2025.
- [8] Quintus Kilbourn. 2024. Zero Trust Execution Environments. <https://collective.flashbots.net/t/zero-trust-execution-environments/3966> [Online; accessed 15. Feb. 2025].
- [9] Quintus Kilbourn, Sylvain Bellemare, Bunnie, and Michael Gao. 2024. *ZTEE - Trustless Supply Chains*. Flashbots. <https://writings.flashbots.net/ZTEE2-Supply-Chains> Accessed: 2026-02-17.
- [10] Filip Rezabek, Moe Mahhouk, Andrew Miller, Stefan Genchev, Quintus Kilbourn, Georg Carle, and Jonathan Passerat-Palmbach. 2025. Proof of Cloud: Data Center Execution Assurance for Confidential VMs. arXiv:2510.12469 [cs.CR] <https://arxiv.org/abs/2510.12469>
- [11] Filip Rezabek, Jonathan Passerat-Palmbach, Moe Mahhouk, Frieder Erdmann, and Andrew Miller. 2025. Narrowing the Gap between TEEs Threat Model and Deployment Strategies. arXiv:2506.14964 [cs.CR] <https://arxiv.org/abs/2506.14964>
- [12] Alex Seto, Oytun Kaday Duran, Samy Amer, Jalen Chuang, Stephan van Schaik, Daniel Genkin, and Christina Garman. 2025. WireTap: Breaking Server SGX via DRAM Bus Interposition. In *2025 SIGSAC Conference on Computer and Communications Security (CCS '25)*. Association for Computing Machinery. <https://wiretap.fail>
- [13] Ruiyi Zhang, Albert Cheu, Adria Gascon, Daniel Moghimi, Phillipp Schoppmann, Michael Schwarz, and Octavian Suci. 2025. SNPeek: Side-Channel Analysis for Privacy Applications on Confidential VMs. arXiv:2506.15924 [cs.CR] <https://arxiv.org/abs/2506.15924>

Received 20 February 2026; revised 20 February 2026; accepted 20 February 2026