Poster: Secure Lifecycle Management of Confidential Virtual Machines in Public Clouds

João Sereno
INESC-ID & IST, University of Lisbon, Portugal
joaohsereno@tecnico.ulisboa.pt

Daniel Castro
INESC-ID & IST, University of Lisbon, Portugal
daniel.castro@tecnico.ulisboa.pt

Nuno Santos
INESC-ID & IST, University of Lisbon, Portugal
nuno.m.santos@tecnico.ulisboa.pt

Luis Rodrigues
INESC-ID & IST, University of Lisbon, Portugal
ler@tecnico.ulisboa.pt

***Abstract*—Federated Learning traditionally relies on differential privacy or cryptographic techniques such as Secure Aggregation and Homomorphic Encryption to protect data during distributed training, but these approaches incur high computational and communication costs. The emergence of hardware-based Trusted Execution Environments, particularly Confidential Virtual Machines (CVMs), offers a practical alternative by enabling secure computation on untrusted cloud infrastructures without algorithmic changes.**

However, CVM deployments by cloud providers—Google Cloud, Microsoft Azure, and AWS—remain opaque, inconsistent, and difficult to reproduce. This paper analyzes their trust models, attestation mechanisms, and deployment limitations, and introduces EVIDENT, a unified framework for transparent CVM lifecycle management. Furthermore, it supports attested interaction scenarios in which CVMs execute workloads owned by third parties—such as confidential AI inference—under cryptographically verifiable trust conditions.

***Index Terms*—Confidential Computing, Confidential Virtual Machines, Remote Attestation, Cloud Security**

I. Introduction

Cloud computing enables scalable, global services with minimal upfront investment, but requires trusting third-party Cloud Service Providers (CSPs) with sensitive data and workloads, often without full visibility into their privileged position. While encryption protects data *at-rest* and *in-transit*, securing data *in-use* remains challenging [1].

Techniques to protect data *in-use* include algorithmic solutions, such as Fully Homomorphic Encryption and Secure Multi-Party Computation, which provide strong security but incur high computational overhead [2]. An alternative solution is to use hardware-based Trusted Execution Environments (TEEs), which process data in isolated, attested environments. While algorithmic solutions minimize the Trusted Computing Base (TCB), i.e., the set of components that must be trusted for system security, TEEs trade off a larger TCB for practical performance, enabling real-world workloads that algorithmic approaches cannot efficiently support [3].

This work was developed within the scope of: proj. no.62–"Responsible AI", financed by European Funds, namely "Recovery and Resilience Plan"–Component 5: "Agendas Mobilizadoras para a Inovação Empresarial", included in the NextGenerationEU funding program; the EU's Horizon Europe research and innovation programme under Grant Agreement No 101189689; FCT under grants UID/50021/2025 and UID/PRR/50021/2025; and, IAPMEI under grant C6632206063-00466847 (PT Smart Retail)

Confidential Virtual Machines (CVMs) extend TEEs by executing entire Virtual Machines in a trusted environment, allowing sensitive workloads—including confidential AI inference and training—to run securely while ensuring only authorized components access the data [4]. Despite these advances, heterogeneity across CSP implementations and limited verification mechanisms complicate secure and reproducible deployment. This paper investigates the confidential computing guarantees offered by major CSPs and proposes mechanisms to make the secure use of CVMs more transparent and accessible to end users.

The contributions of this paper are as follows:

1) A comparative analysis of the trust models employed by three major cloud providers;
2) Methods to enhance trust in the deployment of workloads on Confidential Virtual Machines;
3) The design and implementation of a unified framework for CVM lifecycle management.

II. Background and Related Work

Intel SGX is the first major implementation of hardware-based TEEs [5], which allows the creation of isolated memory regions within user-space processes, protecting code and data from access or tampering by any surrounding software. To mitigate the trust gap inherent in executing code on third-party infrastructure, TEEs provide *remote attestation*, allowing data owners to verify the state, contents, and integrity of the TEE before provisioning sensitive data. While SGX offers a small TCB, it imposes strict memory limits and requires modifications at the application source level. These limitations complicate development and restrict applicability.

To overcome these limitations, Intel TDX and AMD SEV-SNP introduced CPUs capable of isolating and protecting an entire Virtual Machine's memory from the hypervisor. VMs with hardware-backed confidentiality and integrity guarantees—commonly called CVMs—prevent Cloud Service Providers from accessing plaintext memory contents, extending SGX's protection to the entire VM without requiring application-level modifications [6]. CVMs can leverage the available security features through, in the case of AMD, a Secure Processor within the CPU (AMD-SP) to collect a measurement of the VM firmware, which is responsible for launching the VM image. Given that this measurement

TABLE I
SUMMARY OF POSSIBLE TRUST MODELS GIVEN THE VERIFIABILITY LEVELS ON CVM DEPLOYMENTS WITHIN CSPS AND ON-PREMISES

	Cloud Providers			On-premises		
	AWS	**Azure**	**GCP**	**SNPGuard [7]**	**microCVM [8]**	**e-vTPM [9]**
AMD-SP	✓	✗	✓	✓	✓	✓
vTPM	✓	✓	✓	–	–	✓
Firmware	●	◑	◕	●	●	●
Other boot components	●	◑	◕	●	●	●
Kernel	●	◑	◕	●	●	●
Userspace and applications	●	◑	◕	○	○	●

Authenticity: Verifiable (✓) Non-verifiable (✗) Not used (–)
Black coloring represents trust anchoring in AMD-SP
Gray coloring represents trust anchoring in the vTPM (with no AMD-SP link)
○ : L0—Shallow or no verifiability
◔ : L1—Root of Trust Link Authenticity
◑ : L2—Reference-Based Measurement Matching
◕ : L3—Binary Measurement Reproducibility
● : L4—Source Code Measurement Reproducibility

is valid, then all subsequent VM execution is trusted, and, hence, the VM can handle confidential data.

However, trusting the CVM requires correct hypervisor configuration, which major CSPs lack, as shown in Table I. Azure disallows freshness verification of AMD-SP attestation reports, preventing any software verification. GCP only discloses the binary of the VM firmware (not its source), while a measurement can be reproduced, it lacks reliable detection of malicious replacements of the following boot components. AWS discloses the source for its VM firmware, but the firmware does not reliably measure other boot components. All three still require CSP trust.

To improve trust guarantees, SNPGuard [7], Decentriq's microCVMs [8], and e-vTPM [9], rely, respectively, on: including measurements of upcoming boot components in the VM's firmware binary, effectively binding the integrity of these components to the measurement collected by AMD-SP, and the firmware refusing to boot if any boot component presents a different measurement than expected; configuring the hypervisor to allow AMD-SP to collect measurements of components following the firmware; and deploying a software-based TPM (vTPM) that leverages the AMD SEV-SNP VM Privilege Level feature, which require management from the hypervisor, but separates the vTPM from the influence of the CSP.

III. Evident design

Trust model inconsistencies and variability across CSPs are significant barriers to adopting the security primitives that AMD SEV-SNP offers. Although CSPs do not leverage the full capabilities of AMD SEV-SNP with optimal system configurations, CVMs still benefit users with strict security requirements, despite limited guarantees. The Evident framework allows CVM deployments with usability comparable to regular VMs while automating remote attestation and making the trust model fully transparent for each deployment.

An accompanying application, Evident-server, deploys inside the CVM to bootstrap guest applications/models correctly and provide evidence for remote parties to establish trust in the instance. Remote parties verify this evidence through the Evident-client application. After successful attestation, Evident-client exposes an extensible interface enabling users to securely communicate, provision, or interact with guest applications hosted in the CVM.

In Federated Learning, the Model Owner uses Evident-client to remotely attest and provision the CVM with their model, while the Data Owner uses Evident-client to verify that only expected trusted components run on the CVM and access sensitive data. The Model Owner follows an analogous workflow to retrieve the updated model.

IV. Conclusions

This paper exposes the gap between AMD SEV-SNP's security promises and real-world CVM deployments in public clouds. Despite offering hardware-backed protection, current implementations by AWS, Azure, and GCP suffer from opaque attestation, inconsistent trust models, and limited transparency. We analyze these shortcomings and propose a unified framework for secure CVM lifecycle management, enabling practical confidential AI deployments on third-party servers. Our transparency-driven approach aims to foster broader CVM adoption and more trustworthy confidential computing in Federated Learning.

References

[1] Confidential Computing Consortium, "Confidential computing: Hardware-based trusted execution for applications and data," 2022.
[2] Podschwadt, R., et al. "A Survey of Deep Learning Architectures for Privacy-Preserving Machine Learning With Fully Homomorphic Encryption," in IEEE Access, vol. 10, pp. 117477–117500, 2022.
[3] Akram, A., et al, "Performance Analysis of Scientific Computing Workloads on General Purpose TEEs," in 2021 IEEE International Parallel and Distributed Processing Symposium (IPDPS), 2021, pp. 1066–1076.
[4] F. Mo, Z. Tarkhani, H. Haddadi. "Machine Learning with Confidential Computing: A Systematization of Knowledge," in ACM Comput. Surv., vol. 56, no. 11, 2024.
[5] Chakrabarti, S., et al, "Intel® Software Guard Extensions (Intel® SGX) Architecture for Oversubscription of Secure Memory in a Virtualized Environment," HASP, 2017.
[6] Misono, M., et al. "Confidential VMs Explained: An Empirical Analysis of AMD SEV-SNP and Intel TDX," in Proc. ACM Meas. Anal. Comput. Syst., vol. 8, no. 3, 2024.
[7] L. Wilke, G. Scopelliti, "SNPGuard: Remote Attestation of SEV-SNP VMs Using Open Source Tools," in 2024 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW), 2024, pp. 193–198.
[8] Decentriq, "Swiss cheese to cheddar: securing AMD SEV-SNP early boot," https://www.decentriq.com/article/swiss-cheese-to-cheddar-securing-amd-sev-snp-early-boot (Accessed Mar. 27, 2025).
[9] Narayanan, V., et al, "Remote attestation of confidential VMs using ephemeral vTPMs," ACSAC, 2023, pp. 732–743.