IMPERIAL DARTMOUTH

Confidential, Attestable, and Efficient Inter-CVM Communication with Arm CCA

Sina Abdollahi, Amir Al Sadi, David Kotz, Marios Kogias, Hamed Haddadi

The Problem

Confidential Virtual Machines (CVMs) protect sensitive workloads from privileged adversaries such as the hypervisor. However, existing CVM architectures follow a disjoint memory model: a CVM's memory is either shared with the hypervisor or kept private to itself — direct CVM-to-CVM memory sharing is not possible.

As a result, all inter-CVM data must travel through hypervisor-managed channels, requiring expensive per-message encryption/decryption to preserve confidentiality and integrity.

Why This Matters. Compartmentalizing confidential services across multiple CVMs (e.g., separating the networking stack from the inference engine) mitigates the impact of exploits.

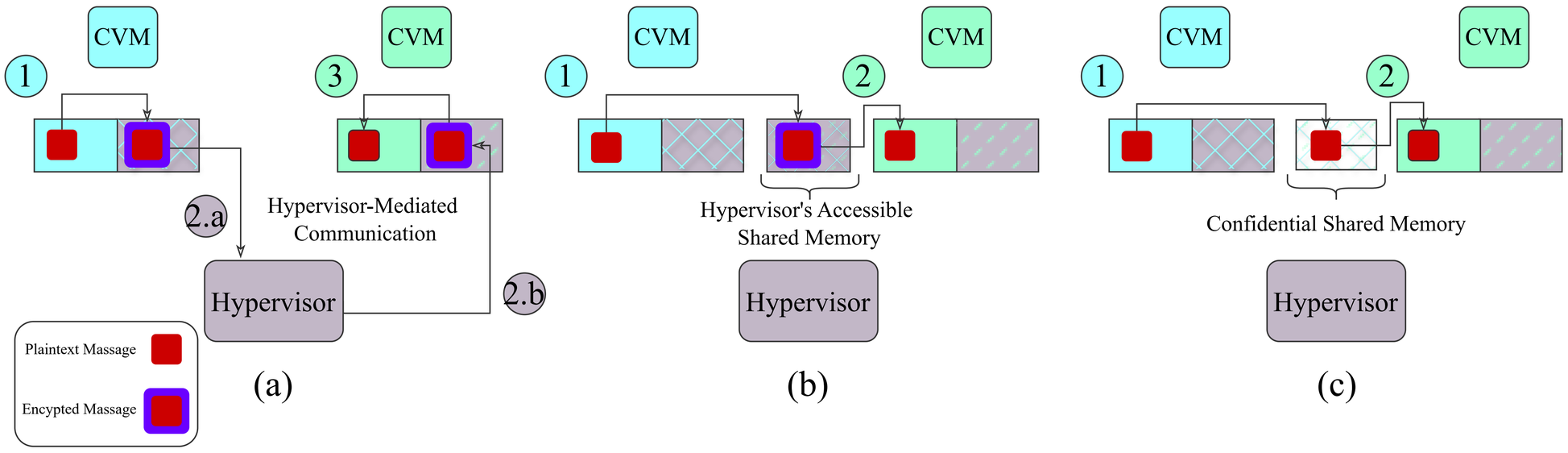

Figure 1: Communication modes between two CVMs.

Key Objectives

Objective	Requirement
Confidentiality	Shared region must be inaccessible to the hypervisor and all non-participating CVMs.
Attestation	Participants must verify each other's identity before any memory is shared.
Access Control	Fine-grained permissions (read-only / read-write) per participant.

CAEC

Trust Model. The platform boots trusted CCA firmware (RMM and Monitor) alongside an untrusted hypervisor. Realms are deployed by mutually-distrustful realm owners. All parties trust CCA hardware and firmware. Physical and side-channel attacks are out of scope.

Overview. CAEC extends Arm CCA with support for Confidential Shared Memory (CSM) — hypervisor-protected memory regions that can be shared between multiple realms while remaining inaccessible to the hypervisor and non-participating CVMs. The RMM is the trusted core: it validates operations, enforces access control, and orchestrates all CSM lifecycle events.

Sharing Lifecycle. CAEC exposes new APIs to Realms. A Provider Realm (1) contributes a portion of its protected address space to instantiate a CSM region, (2) grants or revokes access to it to/by Realms, and (3) defines per-participant permissions (read-only or read-write). Authorized Consumer Realms request to attach; upon mutual consent, the RMM securely maps the same physical pages into Consumer Realm's protected address spaces.

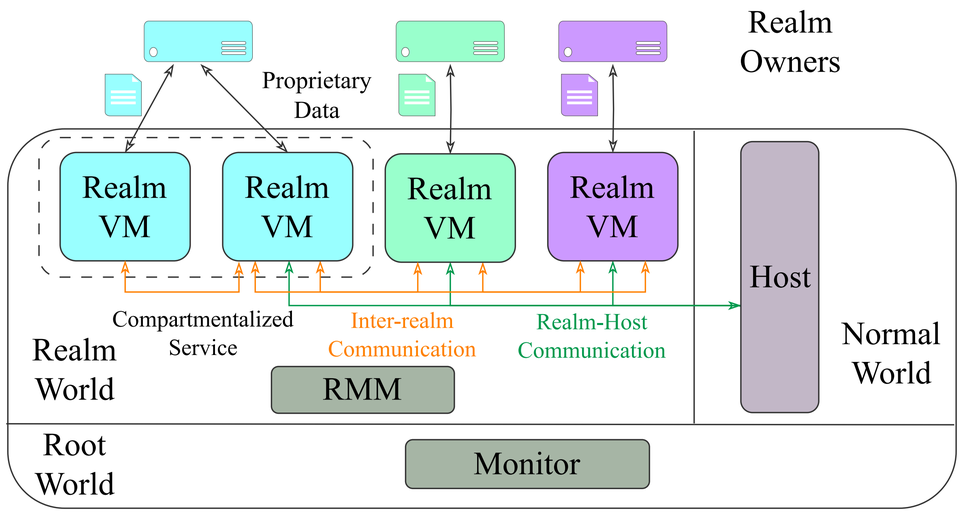

Figure 2: system model

Attestation & Protection. Each realm has a unique, cryptographically bound identifier embedded in its attestation token. Before establishing a region, realm owners mutually attest peer realms and send the verified identifiers into their realms. Sharing decisions, enforced by the RMM, are bound to these authenticated identities, preventing spoofing or unauthorized attachment by any other CVM.

Evaluation

Implemented on OpenCCA, CCA firmware code size increases by only ∼**4%**. Evaluated with an inter-CVM ping-pong benchmark in which two realms exchange messages of varying sizes.

Table 1: CAEC vs. encrypted NW channel (MbedTLS over hypervisor-shared memory) at 64 KB payload. Gains widen with message size.

Metric	Min Gain
Per-message latency	4.4×
CPU cycles per message	4.5×
Throughput	4.8×