Annex C
Threat Model

Informative – Not Normative

C.1 Introduction

This annex describes the threat model that informed the Framework for Continuous Remote Attestation in Confidential Computing Environments (v0.9.6). Its purpose is to make the reasoning behind the framework's requirements visible: the threats the working group considered relevant to confidential computing, why leaving each one unaddressed would be a problem, and why the framework's mitigations are expected to be effective against it. Confidential computing removes large classes of threats that affect conventional cloud workloads, but it does not remove all of them. This annex sets out the residual threats the framework is designed to address.

This annex is informative and does not impose requirements of its own. It motivates the normative requirements in the main framework and corresponds to Section 9 of the main specification, which should be read alongside it. A reader can use it in two ways: to understand why a given requirement exists, by tracing it back to the threat it mitigates, and to assess whether the framework's coverage is adequate for a particular deployment, by comparing the threats listed here against those present in that environment.

The threats considered here draw on several existing threat repositories, including EMB3D and ESTM for hardware and firmware threats and MITRE ATT&CK for ICS for industrial control system threats. These sources informed the working group's thinking and served as references for the kinds of threats worth considering. This annex does not claim to be a formal or complete mapping to any of them, and no formal threat assessment, testing, or validation has been performed against these sources. Because the framework applies attestation across several layers, no single existing repository is comprehensive enough on its own for the scope addressed here. The working group has instead selected the threats it considers most relevant to continuous remote attestation in confidential computing at this stage.

This is a public review draft. The threat set is not exhaustive and is expected to be extended as the framework matures. A complete mapping to the referenced repositories, and empirical validation of the listed mitigations against a reference implementation, are identified as future work.

Confidential computing differs from embedded environments in two critical ways:

- **The Adversary Model:** In embedded environments, the adversary often has physical access to the device. In cloud environments, the Cloud Service Provider (CSP) controls physical access, but the Hypervisor / Host OS is treated as an untrusted adversary.

- **The Lifecycle:** Cloud workloads are ephemeral and dynamic, whereas embedded firmware is persistent.

Therefore, this model applies an “Overlay”: we draw on EMB3D’s treatment of Layer 1 (Hardware) threats but exclude physical tampering (CSP responsibility) and add Cloud Orchestration and Publisher Protocol threats. When the framework is deployed to protect OT / ICS assets, such as IT/OT convergence gateways, SCADA historian databases, or edge controllers running inside TEEs, the threat model extends with an additional "ICS Overlay" informed by MITRE ATT&CK for ICS. ATT&CK for ICS provides a taxonomy of adversary techniques specific to industrial control systems, complementing EMB3D's embedded device focus and the Cloud Overlay's hypervisor / orchestration focus. Where EMB3D addresses hardware and firmware threats and the Cloud Overlay addresses the untrusted hypervisor and ephemeral workload lifecycle, the ICS Overlay addresses adversary behaviors that traverse the IT/OT convergence boundary to manipulate physical processes.

The ICS Overlay is informative and applies only to deployments where TEE-protected workloads interact with operational technology networks. Organizations deploying the framework exclusively in enterprise cloud environments may disregard this overlay without impact to their threat coverage.

This annex is published as a standalone companion document to provide additional detail and to support independent review by threat modeling practitioners.

C.2 Adversary Model

To contextualize the threats, we define the assumed adversary for a Cloud Confidential Computing environment.

C.2.1 Who

- **System Operator (The Host):** A malicious insider at the Cloud Service Provider (CSP) or an attacker with privileged access to the infrastructure.
- **Shared Platform Tenant (The Neighbor):** A malicious actor on the same physical host attempting side-channel attacks or cross-container exploits.
- **Malicious Tenant Insider (The Insider):** A user with valid credentials to deploy workloads, attempting to subvert the workload (e.g., injecting malware into a custom kernel) in violation of the organization's governance.
- **External Actor:** Traditional network-based attacker attempting exploitation via public interfaces.
- **OT-Targeted Adversary:** A state-sponsored or advanced persistent threat actor targeting industrial processes through the IT/OT convergence boundary. This adversary leverages compromised IT infrastructure to pivot into OT networks, using the TEE-protected gateway as an attack vector or a high-value target for manipulation of physical processes.
- **Provider Operating Service:** A compromised, coerced, or adversarial Cloud Service Provider verification service or hardware-vendor-operated attestation service in the trust chain (Type B or Type C trust anchors per the main framework Section 4.1.3). This adversary issues valid-looking signed tokens for the wrong Target Machine, withholds tokens, or asserts passing verdicts on evidence the silicon would have rejected. The adversary is bound to entities in the verification path that the relying party has accepted under its declared Provider Trust Stance (main framework Section 5.3). A Provider-Independent stance excludes this adversary by construction, since the relying party accepts only Type A trust anchors.

C.2.2 Privileges

Full control over the Hypervisor, Host OS, Network Fabric, and Disk Storage. Ability to stop/start VMs, inspect unencrypted memory (outside the TEE), and manipulate I/O rings. Access to IT-side credentials, knowledge of OT protocols (Modbus, OPC UA, DNP3), potential for supply chain access to OT firmware or control logic.

C.2.3 Motivation

Data exfiltration (stealing keys / models), Integrity tampering (modifying workload logic), Denial of Service, or Disruption (causing negative / unintended behavior without breach), Physical Disruption, Safety System Manipulation, Industrial Espionage, or Infrastructure Sabotage.

C.2.4 Capabilities NOT Assumed

- **Physical access to the CPU die:** (e.g., focused ion beam or decapping).
- **Cryptographic Breaks:** Ability to break AES-256 encryption or forge digital signatures from the Hardware Root of Trust (AMD / Intel / Arm / NVIDIA).

C.3 Scope of Protection

C.3.1 Protection Goals

- **Integrity of Launch:** Ensuring the workload (OS, Container, Sidecars) matches the composite baseline (Tenant Policy + Vendor Trust Anchors) to prevent malicious insider attacks.
- **Runtime State Integrity:** Detecting dynamic threats including post-boot kernel modification (DKOM), unauthorized process execution, and behavioral drift.
- **Data Confidentiality (In-Use):** Protecting memory contents from the Host OS/Hypervisor via Hardware Isolation and preventing data exfiltration via Egress Filtering.
- **Session and Transport Binding:** Ensuring that the entity communicating on the network (TLS) is cryptographically bound to the attested TEE identity, as defined by the Attested Transport Binding requirements (TB-1 through TB-6) in the main framework.
- **Freshness and Anti-Replay:** Guarantees that evidence is recent and cannot be replayed by a compromised host or stale publisher.
- **Workload Identity Containment:** Limiting the use of service credentials to workloads that are currently attested and authorized. Short-lived workload credentials issued through SPIFFE / SPIRE can reduce the exposure window associated with static secrets and enable rapid revocation when attestation state degrades.

C.3.2 Assumptions

- **Hardware Root of Trust:** We assume the CPU vendor's hardware (AMD SEV-SNP / Intel TDX / Arm CCA / NVIDIA CC) functions as specified and is not compromised at the silicon level.
- **CSP Physical Security:** Physical attacks (probing, glitching) are out of scope as they are managed by the CSP's physical security controls.
- **Signing Key Integrity:** We assume attestation signing keys, CPU microcode signing keys, and the CI/CD private signing keys are not compromised.
- **Verifier Trust:** The Enterprise Validator is assumed to be running in a secure environment.
- **Policy Sovereignty:** Generally, we assume the Policy Definition provided by the tenant is the source of truth for business logic and application configuration.

- **TEE Reporting Visibility:** The attestation framework only sees what the TEE reports; we cannot detect forged hardware measurements if the Silicon Root of Trust itself is compromised.

- **Identity Plane Integrity:** If SPIFFE / SPIRE is used, the identity control plane is assumed to enforce registration and issuance policy correctly. The security benefit depends on binding identity issuance and renewal to current attestation state rather than to deployment metadata alone.

C.3.3 In-Scope Threat Categories

- **Firmware and Boot:** Malicious bootloaders, BIOS implants, and Version Rollback attacks.

- **Shared Resource Attacks:** Neighbor VMs exploiting Side-Channels (Spectre / Meltdown) or shared kernel structures.

- **Orchestration Manipulation:** Host / Hypervisor tampering with Virtio rings, disk I/O, or entropy sources.

- **Runtime Exploitation:** Container escapes, in-memory Kernel Rootkits, and unauthorized network egress.

- **Protocol Attacks:** Man-in-the-Middle (MITM), Evidence Replay, and the “Lying Publisher” scenario.

C.3.4 Out of Scope and Residual Risks

- **Availability / Denial of Service:** We cannot prevent the System Operator (Host) from terminating the VM, starving CPU resources, or dropping network packets. Confidential Computing guarantees Confidentiality and Integrity, not Availability.

- **Residual Side-Channels:** While TCB Versioning mitigates known hardware vulnerabilities, we cannot eliminate all microarchitectural side-channels (e.g., new SMT leaks) inherent to shared silicon. This is a residual risk of multi-tenant cloud computing.

- **Intrinsic Workload Vulnerabilities:** The framework does not make unsafe code (e.g., C/C++) memory safe. If a workload contains a buffer overflow, the bug remains. We detect and contain the consequences (exploitation / behavioral drift), but we cannot prevent the initial memory corruption.

- **Data-Only / Logic Attacks:** Exploits that manipulate valid application memory (Heap/Stack) without altering control flow or executing new code (e.g., changing

an isAdmin flag or SQL injection). These are application-layer failures that do not trigger eBPF/Kernel violation signals and must be mitigated by software quality.

- **Forced Publisher Transmission:** We cannot cryptographically force a compromised Publisher to send a “Breach” event. If an attacker blocks the network or patches the Publisher binary to be silent, the Enterprise Validator must rely on Heartbeat Timeouts (Dead Man’s Switch) to assume compromise.

- **Residual Credential Misuse:** If a workload is compromised after receiving a valid workload identity credential, the adversary may be able to use that credential until expiration or revocation. Short credential lifetimes, attestation-gated renewal, and rapid revocation reduce this window but do not eliminate it entirely.

- **Identity Is Not Attestation:** SPIFFE / SPIRE authenticates workload identity but does not independently establish TEE integrity, runtime health, or transport binding to the End User session. These properties remain dependent on the attestation framework and Attested Transport Binding requirements.

- **Physical Attacks:** Chip decapping, probing, or glitching are managed by the CSP’s physical security controls.

C.4 The Threat Matrix

The following matrix maps specific threat scenarios to the attestation layers defined in the main framework, along with the corresponding mitigation mechanisms.

Layer / Domain	Threat Category	Specific Threat Scenario	Mitigation Mechanism
L1: Platform	Firmware Compromise	Malicious Bootloader/BIOS injects code before OS start.	Measurement: Hardware Root of Trust (TPM/SNP) verifies boot chain measurements (PCRs/Report) against a Golden Policy.
L1: Platform	Side-Channels	Neighbor VM exploits shared cache (Spectre/Meltdown) to leak keys.	Policy: TCB Version Check enforces microcode patch levels against known vulnerabilities. Residual Risk: Zero-day or unpatched side-channels are not covered.
L1: Platform	Rollback Attacks	Attacker forces a reboot into an older, vulnerable firmware/kernel version.	TCB Versioning and Anti-Replay: Policy enforces min_tcb_version (L1). The Publisher protocol uses fresh Nonces to reject stale evidence.
L1: Platform (ICS)	OT Firmware Manipulation	An adversary targets the firmware of the TEE platform hosting an OT gateway, attempting to install a persistent implant that survives reboots and undermines all higher-layer attestation.	L1 Platform Attestation: Hardware Root of Trust verifies boot chain measurements. Policy enforces min_tcb_version to reject firmware with known vulnerabilities. TCB versioning prevents rollback to compromised firmware versions. A firmware implant is detectable because it alters the boot measurement chain.
L2: Image	Supply Chain Injection	Registry serves a poisoned container image (External Actor).	Signature: Kernel (IMA) or Runtime enforces signature verification against the trusted CI/CD root key.
L2: Image	Malicious Insider	Tenant Admin injects malware into the base OS image.	Composite Baselines (Override): Runtime verification validates the loaded Kernel/OS against Vendor (3rd Party) Reference Manifests. Defense: Platform baselines explicitly override Tenant allowlists to prevent an insider from authorizing a compromised kernel.
L2: Image	TOCTOU Swap	Attacker swaps the binary after measurement but before execution.	Pinning: The Attestation Token binds the specific image digest to the released session via

Layer / Domain	Threat Category	Specific Threat Scenario	Mitigation Mechanism
			transport binding key commitment (TB-1).
L2: Image (ICS)	Unauthorized Control Logic	An adversary with valid credentials deploys modified control logic or unauthorized OT application code to a TEE-protected gateway, bypassing credential-based access controls.	Composite Baseline (L2): All container images and executables are measured against the Golden Baseline at boot and runtime. Vendor-defined platform baselines prevent tenant credential holders from authorizing compromised control logic. IMA measurements detect unauthorized code regardless of deployment credentials.
L3: Runtime	Kernel Modification	Runtime exploit injects a rootkit or modifies syscall tables.	Semantic Verification: Runtime Integrity Service (e.g., Invary KIM) validates kernel memory layout against the Golden Image.
L3: Runtime	Application RCE in the Guest	Workload is exploited (e.g., buffer overflow) to run unexpected code / shells.	Behavioral Containment: eBPF sensors and policy help detect or contain post-exploit behavior (e.g., unexpected shell spawning) but do not prevent the initial bug.
L3: Runtime	Network Exfiltration from Compromised Guest	Compromised workload attempts to send secrets to an unknown C2 server.	Egress Filtering: eBPF network hooks block connections to IPs/Domains not defined in the Allowlist. *NOTE: Cannot prevent exfiltration via allowed side-channels without deep inspection.*
L3: Runtime (ICS)	Manipulation of Control	A compromised workload within the TEE manipulates OT protocol commands (Modbus writes, OPC UA method calls) to cause unsafe physical outcomes such as valve misoperation or setpoint manipulation.	Behavioral Containment (L3.2): eBPF sensors monitor process behavior including network connections and syscall patterns. Connections to OT network endpoints not in the allowlist are blocked. Anomalous command patterns (e.g., unexpected Modbus function codes) can be detected through behavioral profile deviation. *NOTE: Deep OT protocol inspection (e.g., validating Modbus register values) requires application-layer extensions beyond L3.2.*

Layer / Domain	Threat Category	Specific Threat Scenario	Mitigation Mechanism
Orchestration	Hypervisor Tampering	Host manipulates Virtio ring to spoof network packets or corrupt disk reads.	Encryption and Sanity Checks: All I/O is TLS-encrypted (protects payload). eBPF validates packet headers at the guest network stack. Out of Scope: Host-initiated DoS (dropping packets, starving CPU/IO, terminating VM) cannot be prevented by guest software.
Identity / Protocol	Service Impersonation or Credential Reuse	Attacker reuses static service credentials or impersonates a workload inside the deployment.	Short-lived SPIFFE / SPIRE-issued workload credentials, automated rotation, registration policy, and attestation-gated issuance / renewal reduce the lifetime and portability of stolen credentials. Residual risk remains until credential expiry or revocation if a compromised workload still possesses an active credential.
Protocol	The “Lying” Publisher	Publisher modifies evidence, replays stale “Good” states during an active breach, or intentionally suppresses “Revocation” notifications.	1. Integrity (Modification): Publisher passes TEE-signed artifacts. It cannot forge signatures. 2. Anti-Silence (Withholding): The Validator enforces a strict Heartbeat Policy, checking that the State Epoch increments or remains consistent. If the Publisher fails to send a signed Keep-Alive or update within window T, the Validator defaults to Untrusted. 3. Anti-Replay: Push: Enforced via TEE-signed monotonic sequence numbers. Query: Enforced via Validator-supplied nonces bound to the response.
Protocol	Replay Attack	Attacker presents an old, valid token for a now-compromised VM.	Freshness: Protocol uses Challenge-Response Nonces (preferred) or short-lived Epoch IDs.
Protocol	Relay / MITM	Attacker forwards traffic to a valid TEE but inspects it in transit.	Attested Transport Binding: The TLS session is cryptographically bound to the TEE’s hardware attestation evidence via key commitment (TB-1) and freshness (TB-2). Traffic cannot be decrypted outside the specific TEE instance. Formal analysis has identified relay vectors in

Layer / Domain	Threat Category	Specific Threat Scenario	Mitigation Mechanism
			certain intra-handshake approaches; see TB-6 disclosure requirements.
Protocol (ICS)	OT Remote Service Exploitation	An adversary exploits remote access services (OPC UA, RDP, SSH, VPN) to reach the TEE-protected OT gateway, potentially relaying traffic through a compromised intermediate host to present valid-appearing connections.	Attested Transport Binding: TB-1 key commitment and TB-2 freshness guarantee prevent relay and MITM attacks on transport sessions to the TEE. The TEE's hardware identity is cryptographically bound to the TLS session, ensuring the client communicates with the genuine TEE gateway and not a relay. eBPF egress filtering (L3.2) restricts outbound OT protocol connections to authorized endpoints.
Trust Anchor / Provider	Compromised or Coerced Verification Service	A CSP-operated verification service or a hardware-vendor-operated attestation service in the trust path is compromised, coerced, or adversarial, and issues a passing token for evidence the silicon would have rejected.	Provider Trust Stance Disclosure (main framework Section 5.3): the relying party declares which trust anchor types it accepts. A Provider-Independent stance excludes this threat at the architectural level by accepting only Type A (silicon-rooted) anchors. Where Type B or Type C is accepted, the relying party has incorporated the operating service into its TCB as an explicit policy choice.
Trust Anchor / Provider	Verification Service Withholding	A CSP-operated verification service or vendor-operated attestation service refuses to issue valid tokens, becomes unreachable, or selectively delays evidence delivery, denying the relying party fresh attestation.	Dead Man's Switch (main framework Section 7.4): if fresh evidence is not received within the configured window, the Publisher publishes a FAIL status and trust is revoked. Where the relying party also requires a Provider-Independent fallback, a Type A path can be configured to take over from a degraded Type B service.

C.5 Summary

This model accepts the rigor of EMB3D for Layer 1 but applies a necessary “Cloud Overlay” to address the untrusted Hypervisor, the malicious Tenant Insider, and the specific constraints of ephemeral workloads.

Key Trust Anchors:

- **Composite Supply Chain:** We do not rely solely on the Tenant’s signature. Integrity is rooted in a Composite Baseline where the Base OS/Kernel must match a Trusted Vendor (e.g., RedHat / Microsoft) reference to detect insider tampering.

- **Transport Binding:** We rely on Attested Transport Binding to extend the TEE’s identity to the network session, preventing Relay/MITM attacks that standard TLS cannot detect. The binding requirements (TB-1 through TB-6) are platform-neutral and applicable across all current TEE technologies.

- **Freshness via Nonces:** We prioritize Nonce-based verification over synchronized clocks to eliminate Time-of-Check-Time-of-Use (TOCTOU) vulnerabilities caused by clock skew or manipulation.

- **In-Boundary Appraisal:** We minimize the privacy risk and bandwidth cost of raw memory exportation by executing the heavy analysis (e.g., kernel integrity verification) inside the TEE (or a protected sidecar). The Publisher streams only the signed Appraisal Results, not the raw binary evidence.

For deployments at the IT/OT boundary, the framework's threat model extends naturally through the ATT&CK for ICS Overlay. The same layered attestation architecture that protects cloud workloads – hardware integrity (L1), code identity (L2), behavioral containment (L3), and transport binding (Protocol) – applies to OT gateway and edge controller deployments. The primary additional consideration is that OT environments may require application-layer protocol awareness (Modbus, OPC UA, DNP3) beyond the framework's current L3.2 behavioral monitoring scope, which organizations should address through deployment-specific extensions.

Where deployments require workload-to-workload mutual authentication, SPIFFE / SPIRE can complement this framework by replacing static service credentials with short-lived workload identities. This is most effective when identity issuance is driven by attestation state. In that model, a workload is not trusted merely because it was scheduled or registered; it is trusted only while it continues to satisfy the framework’s layered attestation requirements.

NOTE: This threat model is an informative companion to the Framework for Continuous Remote Attestation in Confidential Computing Environments, Version 0.9.6. It should be read in conjunction with Section 9 of the main framework specification.

C.6 Abbreviations and Acronyms

Term	Definition
ATB	Attested Transport Binding
ATT&CK for ICS	ATT&CK for Industrial Control Systems
BIOS	Basic Input/Output System
C2	Command and Control
CC	Confidential Computing
CCA	Confidential Compute Architecture
CI/CD	Continuous Integration / Continuous Delivery (or Deployment)
CPU	Central Processing Unit
CSP	Cloud Service Provider
DKOM	Direct Kernel Object Manipulation
DNP3	Distributed Network Protocol Version 3
eBPF	Extended Berkeley Packet Filter
EMB3D	Embedded Threat Model framework
ESTM	Embedded Systems Threat Matrix
Host OS	Host Operating System
ICS	Industrial Control System
IMA	Integrity Measurement Architecture
IT	Information Technology
L1	Layer 1
L2	Layer 2
L3	Layer 3
MITM	Man-in-the-Middle
MITRE ATT&CK	Adversarial Tactics, Techniques, and Common Knowledge
Modbus	Modbus Industrial Protocol
OPC UA	Open Platform Communications Unified Architecture
OS	Operating System
OT	Operational Technology
PCR	Platform Configuration Register
RCE	Remote Code Execution
RDP	Remote Desktop Protocol
SCADA	Supervisory Control and Data Acquisition
SEV-SNP	Secure Encrypted Virtualization – Secure Nested Paging
SMT	Simultaneous Multithreading
SPIFFE	Secure Production Identity Framework for Everyone
SPIRE	SPIFFE Runtime Environment
SSH	Secure Shell
TB-1 through TB-6	Transport Binding Requirements
TCB	Trusted Computing Base
TDX	Trust Domain Extensions

Term	Definition
TEE	Trusted Execution Environment
TOCTOU	Time-of-Check Time-of-Use
TPM	Trusted Platform Module
VM	Virtual Machine
VPN	Virtual Private Network