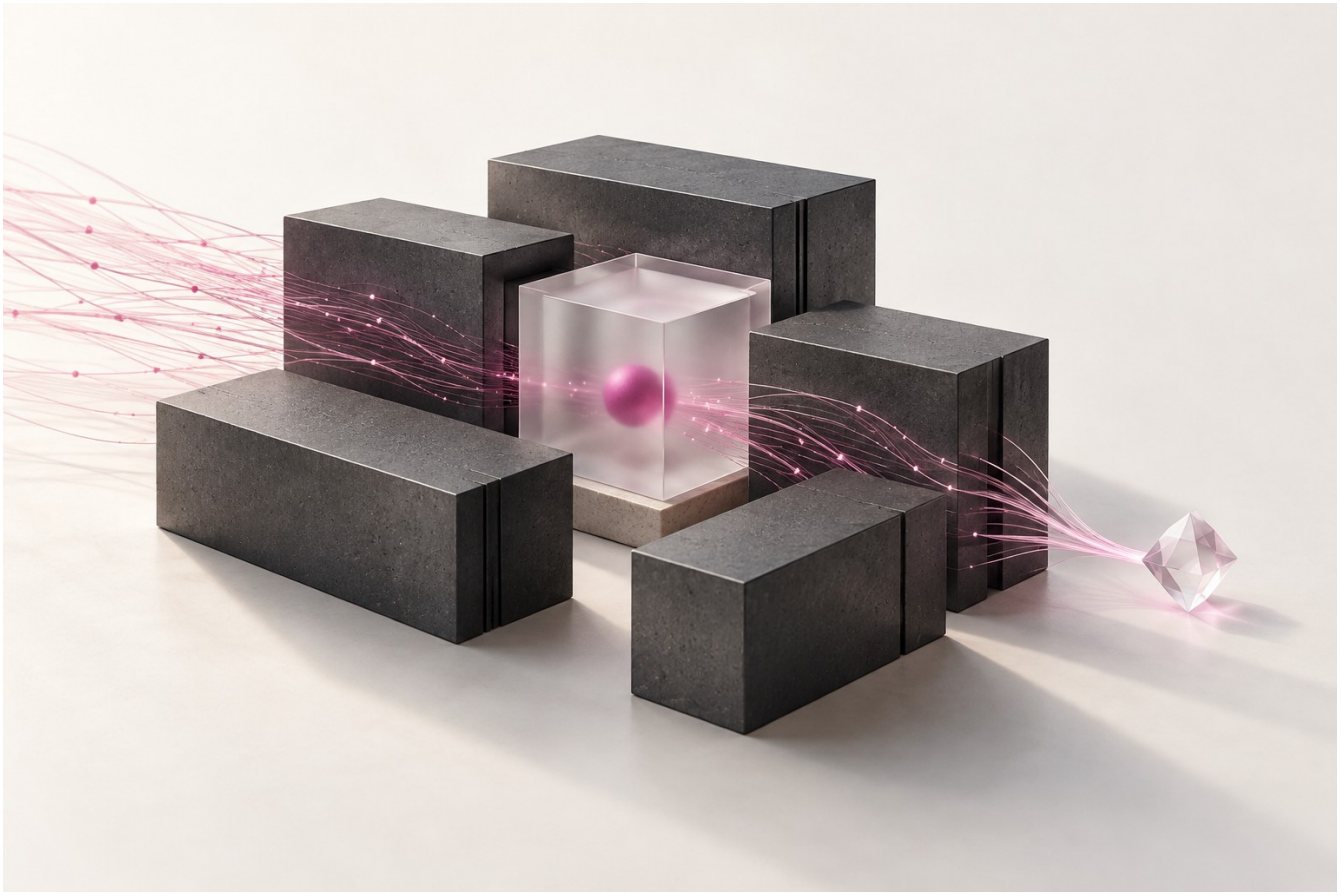
Confidential compute on non-confidential silicon

Architecture and security boundaries



Protected information and controlled answers

An application can need an answer without needing the records behind it. SCSA describes how an approved computation can use protected information on external infrastructure, with separate controls over the calculation and the disclosure of its result.

This paper covers the execution, authority and custody boundaries, the security requirements for each protection profile, and the evidence used to assess a deployment.

1 An answer without unrestricted access

A purchasing agent receives a request to order replacement equipment. It needs to establish whether the purchase fits the company's financial constraints. Giving it access to bank statements, outstanding liabilities and every supplier contract would reveal far more than the task requires.

A confidential tool endpoint exposes a narrower operation. The agent submits the purchase parameters. The information owner supplies the protected inputs needed by an approved calculation. The endpoint returns a permitted decision, with evidence that identifies the operation and its authorisation. The agent never needs the source records in its language-model context.

The purchasing example follows one operation from protected inputs to controlled disclosure, with computation outside the information owner's infrastructure.

What ordinary hardware changes

Conventional encryption protects stored data and network traffic. During ordinary application execution, plaintext and keys can enter memory accessible to a sufficiently privileged operator. Hardware-backed confidential computing limits that exposure through a trusted execution environment, or TEE.

The Confidential Computing Consortium's definition specifically involves a hardware-based, attested TEE. We use the title of this paper to describe a broader architectural question: how can processors without that facility still participate in protected computation? Cryptographic execution does not turn them into general-purpose hardware enclaves. [1]

An encrypted-computation worker receives ciphertext and permitted evaluation material. An MPC participant receives the inputs or shares defined by a multiparty protocol. Protection follows from the selected construction and its assumptions, rather than a claim that an administrator cannot inspect the worker.

The product boundary

The SCSA architecture combines a defined computation with control over its authorised output. It separates execution, authority and custody. A workload can use ordinary workers, an appropriate hardware-protected route, or a hybrid arrangement, subject to an explicit security contract.

An owner-hosted API may already solve the problem. Protected outsourcing becomes relevant when the owner cannot or does not want to host the material computation, several organisations contribute private assets, or the recipient needs evidence that crosses an administrative boundary.

The first deployment question should therefore be specific: which operation is blocked today, who needs its answer, and who must not receive the information used to calculate it?

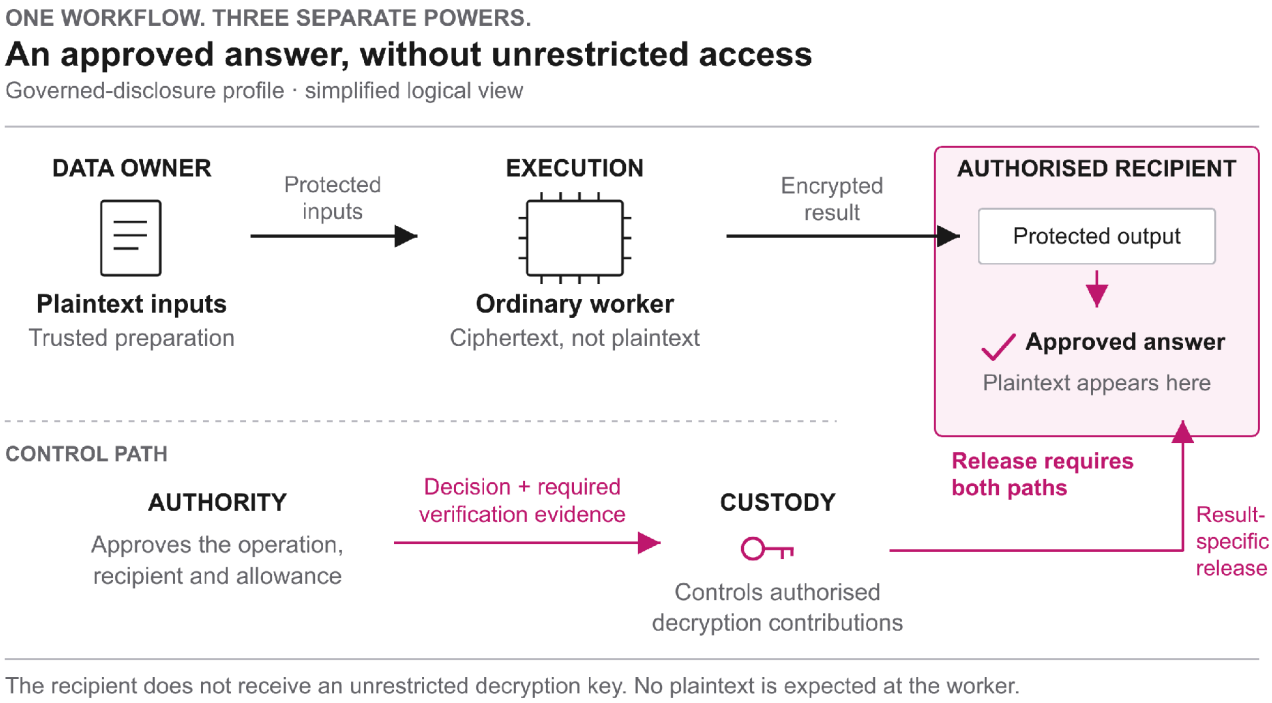
2 Execution authority and custody

Infrastructure ownership is a poor substitute for a clear allocation of powers. The operator may need to schedule workers and repair machines without obtaining the ability to decrypt customer records or approve their disclosure.

Execution performs the admitted calculation. Workers receive the representation permitted by the selected security profile. In an encrypted-input route, that representation can remain ciphertext even when every worker is controlled by the same provider.

Authority decides whether an operation and its disclosure are allowed. It tracks the current policy, recipient, delegated permissions and any shared allowance. Its decisions must survive retries and recovery.

Custody controls the cryptographic material needed to recover protected information. In a governed-disclosure profile, the recipient does not hold an unrestricted decryption key. The custodians contribute to an authorised result under a specified threshold construction.



Separation must survive administration

Different service names do not establish independence. A single administrator who can restore every custodian's backup may control enough material to defeat the threshold. Deployment credentials, update rights and recovery powers must appear in the same analysis as the running services.

The owner remains trusted before input protection, and the recipient can copy an answer after disclosure. Cryptography changes the worker's access; it does not remove those endpoint responsibilities.

3 Different promises need different profiles

Protecting a customer's input, protecting a proprietary model and restricting output disclosure require different constructions. A deployment must identify which promise it makes before it selects workers or publishes performance figures.

Profile	Intended protection	Principal remaining trust
G1 Protected input	Outsource work over customer-private inputs. All workers and relays may collude within the admitted malicious-worker construction.	Customer endpoint, cryptographic implementation and required verifier. The customer controls decryption.
G2 Governed disclosure	Release an approved answer without giving the recipient unrestricted decryption. Workers and relays may collude under the admitted construction.	Authority safety, custody threshold, trusted endpoints and cryptographic implementation.
G3 Joint private	Protect assets contributed by different parties, including model parameters where supported.	The selected MPC protocol, its corruption bound and the output-release construction.

When the customer holds the key

G1 protects the customer against the compute provider. A customer holding the decryption key and a compatible output ciphertext can decrypt it. An additional policy statement cannot reliably withhold that output from the same customer.

G2 addresses a different relationship. The information owner and the output recipient may be different organisations or services. Custody-controlled release determines whether the recipient can recover a particular answer. Compromise of the required custody threshold defeats that boundary.

When both sides contribute secrets

A model owner may wish to protect coefficients while another party protects its inputs. G3 requires an admitted protocol for that combination. One example is a three-party, honest-majority, malicious-secure profile that tolerates at most one malicious party. Two colluding parties exceed that profile's guarantee. Other protocols have different thresholds and costs. [4]

Separate accounts under one administrator do not satisfy an independence requirement merely because they look like separate parties. The scheduler must reject a placement that violates the selected corruption assumptions.

A profile also states whether participants can prevent completion. Malicious security with abort can preserve secrecy while allowing a participant to stop the computation. That behaviour must be reflected in the service's availability expectations.

4 How the protected calculation runs

Homomorphic encryption supports evaluation over ciphertext without giving the worker the secret decryption key. The supported arithmetic and its costs depend on the scheme. Exact modular arithmetic and approximate-number arithmetic have different numerical semantics. A model's range, precision and overflow behaviour must match the chosen construction. [2]

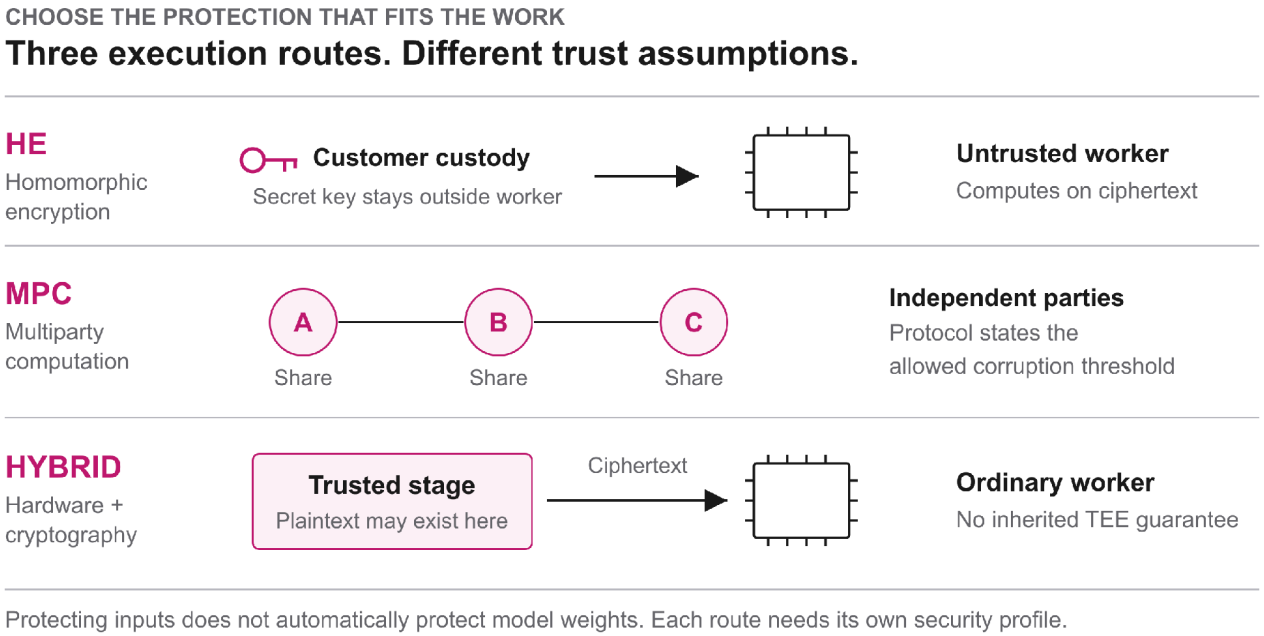


Figure 2. Execution routes place secrets and trust in different locations. Homomorphic encryption, MPC and a hybrid arrangement are mechanisms, not interchangeable assurance labels. Input protection alone does not establish model confidentiality.

Multiparty computation distributes work according to a protocol. Participants may exchange shares and consistency checks over several rounds. Latency depends on those rounds as well as local arithmetic. Nearby machines and participants in distant regions therefore present different operating conditions. [4]

A hybrid design can keep sensitive stages in trusted hardware and delegate suitable protected work to ordinary processors. Published systems such as Slalom explore verified outsourcing between trusted and untrusted devices. Slalom's results apply to its particular construction and measurement conditions. [6]

Where the boundary can break

Sending plaintext from a trusted CPU to an ordinary GPU exposes it at the GPU unless another protection mechanism applies. The same problem arises if a coordinator reconstructs an MPC result before the disclosure decision and encrypts it afterwards. Both transitions must respect the original security contract.

Each accepted execution plan therefore needs a complete account of its intermediates. Unsupported operations must cause refusal or a newly approved plan. They must not trigger an undisclosed plaintext fallback or a change in model precision.

5 Confidentiality and correctness

An encrypted answer can be wrong. A malicious worker might substitute another computation, corrupt a result or replay an earlier response. A service that promises confidentiality against that worker must also analyse how it handles malicious outputs.

Homomorphic encryption deliberately permits transformations of ciphertext. Ordinary authenticated-encryption expectations cannot simply be transferred to it. Research on verifiable FHE shows why integrity gaps and decryption feedback can also become confidentiality problems. The selected construction must validate the required evidence before useful decryption feedback reaches an untrusted evaluator. [2, 3]

Three questions about a model result

Execution correctness asks whether the result matches the specified arithmetic. Model fidelity asks whether that arithmetic faithfully represents the intended reference model. Application quality asks whether the model produces useful decisions. A proof addressing the first question does not settle the other two.

For example, replacing an activation with a bounded polynomial changes the numerical contract. The revised model needs approval and quality evaluation. A correct proof of the revised arithmetic cannot establish that its predictions remain acceptable.

Evidence	What it can support	What it cannot establish alone
Computation proof or check	A specified relation between protected inputs, the approved calculation and output.	Truth of the source records or business value of the model.
Authority certificate	A committed decision under the stated authority protocol.	Correct execution of the calculation.
Custody contribution evidence	Correctness of the specified contribution to output recovery.	Integrity of every custodian or administrator.
Hardware attestation	Appraised claims rooted in a particular hardware and software environment.	Universal application correctness or organisational independence.

What a receipt means

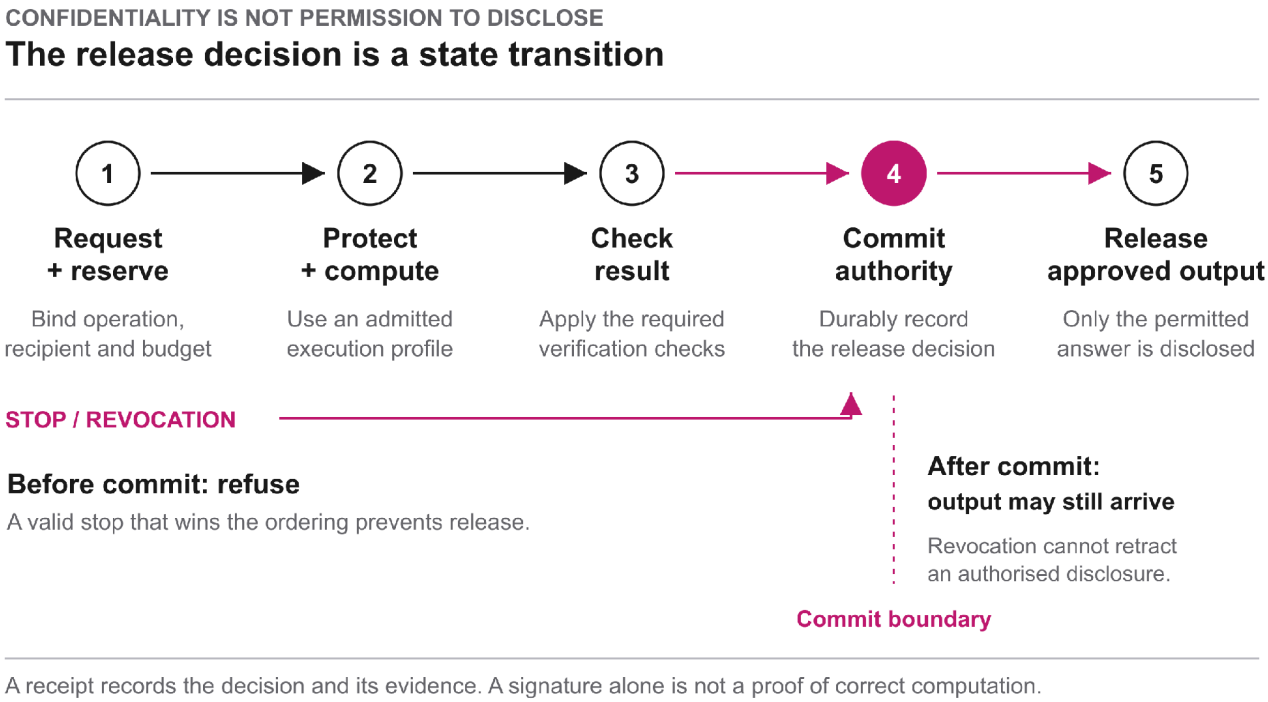
A receipt binds the operation to its evidence, policy and release decision. A verifier must distinguish the kinds of evidence it contains and report missing or unsupported claims. A signature identifies an authenticated statement; it does not make every assertion in that statement true.

A historical receipt also has a time boundary. Verifying it offline cannot establish that no later revocation or conflicting record exists. Trust roots must come from a trusted configuration, not merely from the bundle asking to be trusted. The RATS architecture makes a related distinction between evidence, appraisal and the relying party's decision. [5]

6 Following a purchase check

Return to the purchasing agent. Its permitted question is whether a particular purchase satisfies a defined set of private financial constraints. In this example, the calculation is public and the financial inputs are protected. Protecting model parameters as well would require a profile designed for that purpose.

Before work starts, the execution contract identifies the input owners, approved computation, numerical rules and output schema. It also fixes the intended recipient, policy, authority state and required evidence. The agent's identity alone does not grant access to the source records.



The owner prepares protected inputs inside its trusted boundary. Admission checks that the selected computation and deployment meet the contract. Workers produce an encrypted answer and the required evidence. The release path checks the calculation's bindings and the current authority conditions.

The authority service then commits the exact permitted disclosure. Custodians validate that decision and the required computation evidence before supplying ciphertext-specific contributions to the named recipient. Those contributions must not expose reusable custody shares. The recipient recovers only the output authorised by the construction.

A decision is not a purchase order

The agent may receive a bounded answer such as whether the request passes the agreed constraints. That does not authorise it to place the order. An external purchasing or payment endpoint must enforce its own permitted effect and participate in the required transaction or idempotency protocol.

Even a yes-or-no result reveals information. Repeated queries may let the agent estimate a confidential limit. Disclosure analysis must cover the allowed family of queries and their combined effect, rather than judging each answer in isolation.

7 Authority that survives failure

A copied agent should not receive a new spending allowance simply because another process starts. Delegated authority must either draw from one durable shared allowance or use disjoint allocations that the parent has already reserved.

The authority service records reservations, committed disclosures and unresolved outcomes. A timeout means the caller does not know what happened. It is not evidence that nothing happened, and it must not automatically replenish the allowance.

After a committed release, a retry should recover the same operation rather than authorise another effect. External systems need compatible transaction behaviour. An internal ledger cannot promise exactly-once execution in an unrelated payment service that does not participate.

A network partition

In online-authority mode, operations that require an unavailable quorum stop or remain unresolved. An allocated mode may continue within previously reserved limits only where the enforcement domain satisfies its anti-rollback assumptions. A counter stored on a hostile worker is insufficient: the operator could restore an old snapshot and spend the allowance again.

Unused allocations cannot safely return to the shared pool just because their holder is unreachable. Reconciliation must establish what remains available before it is reassigned.

Migration and recovery

A replacement encrypted worker need not receive decryption authority. It can restart or resume compatible protected work under a fresh attempt identity. The output remains bound to the approved operation.

Authority migration is more demanding. It carries spent allowances, outstanding reservations, committed releases and recognised epochs. The predecessor must be fenced from creating new valid authority before the successor becomes authoritative. Restarting the application alone does not accomplish this.

Custody recovery follows a separately declared policy. The customer must know who can restore keys or replace participants and how those powers affect the threat model. If all legitimate recovery material is lost, the data may become permanently unavailable.

The limit of the stop button

A revocation ordered before release commitment blocks that disclosure under the specified protocol. A later revocation cannot recall the committed answer, even if delivery has not finished. Recipients can retain plaintext they already obtained.

Records of key-destruction attempts and their coverage can support an audit. They cannot prove every secret copy disappeared. Claims about cryptographic erasure need explicit assumptions about backups, prior compromise and every relevant key copy.

8 Mixed fleets and observable information

An estate may contain confidential-computing hardware alongside ordinary GPUs. A common policy interface can manage both, provided it keeps their assurance claims distinct.

Hardware evidence passes through a verifier that checks the applicable endorsements, freshness and reference values. The relying party then decides whether the appraised state meets its policy. This follows the role separation in the RATS architecture. [5]

An ordinary worker has no equivalent hardware claim. It may still be eligible for an encrypted route whose construction permits an untrusted evaluator. A software fingerprint cannot substitute for hardware-backed evidence, and several appraisers cannot repair a shared compromised root merely by agreeing.

Updates without silent downgrades

Certificate rotation changes trust material; a firmware upgrade may change measurements and security-version claims. Both require controlled evaluation. A rolling upgrade can allow several approved versions to coexist, but an unknown state must not acquire approval automatically because it is newer.

New assignments and key releases should follow the current eligibility decision. Handling a running job requires an explicit policy for re-attestation, key lifetime and interruption. Revocation cannot retrieve a key already released to a compromised environment.

An HSM can strengthen custody for operations it actually performs within its boundary. If it unwraps a key for use in host memory, that later use falls outside the HSM. Key-origin evidence does not automatically attest the application that controls release.

The information that remains visible

Protected inputs can coexist with observable job dimensions, ciphertext lengths, timing, frequency and routing. Public model identity or structure may also be visible. The leakage declaration must match the actual protocol and deployment.

Padding and batching can alter these observations, but they do not establish general traffic-analysis resistance. Trusted endpoints and custody components still need protection against side channels because they hold secrets.

Outputs deserve the same care. Enough related scores can reveal a protected feature vector. A minimum cohort size or query limit is not, by itself, differential privacy. If a deployment uses differential privacy, it needs a defined mechanism and composition budget.

Nor does cryptography establish legal sovereignty or permission to process information. Location, jurisdiction and consent remain separate claims with their own evidence. The technical architecture can help enforce an approved policy; it cannot decide whether that policy is lawful.

9 Measuring a useful service

The relevant performance unit is a completed operation that meets its security and output-quality requirements. Timing an arithmetic kernel while omitting encryption, verification or custody produces an incomplete account of the service.

Begin with a fixed workload. Define its inputs, numerical semantics and permitted output, then choose the security profile and adversary assumptions. The plaintext baseline should use the same hardware and an equivalent quality requirement. Any model approximation or quantisation change must be disclosed and evaluated separately.

What belongs inside the measurement

Measure	Required scope
End-to-end latency	Input protection, transport, computation, required verification and authorised output recovery. Report p50 and p95.
Accepted throughput	Completed, accepted operations per second while meeting the latency and quality limits.
Customer workload	Encryption, verification, result processing and any customer-hosted services.
Traffic and resources	Protocol messages, proof material, custody traffic, memory and compute across all roles.
Cost per useful result	Provisioned and usage costs within a declared boundary, including failed attempts, idle capacity and retries.

Topology changes the result

Fast local GPU interconnects help some communication patterns. They do not remove cryptographic arithmetic or the round trips between independent MPC participants. Cross-region inference may accumulate waiting time repeatedly, especially during sequential generation.

A large fleet count is therefore insufficient evidence of efficient scaling. Aggregate serving across several regional pools and one model distributed across those regions are different experiments. Each needs its own baseline and resource accounting.

The customer alternative

Compare against the deployment the customer could actually use: an owner-hosted API, conventional provider hosting, a hardware-protected service or another protected-computation approach. Differences in trust assumptions belong beside the performance results.

The commercial case is strongest when a protected operation enables valuable work that the customer otherwise cannot outsource. Even then, the full cost must be acceptable. Performance

depends on the workload, protection profile, deployment topology and complete measurement boundary. One overhead ratio cannot describe every configuration.

10 Choosing the first workload

A useful first endpoint has a narrow output, a calculation whose semantics can be fixed and a customer who can explain why conventional hosting is unsuitable. Bounded scoring or constrained ranking may provide a more manageable evaluation than beginning with a general-purpose language model.

A manufacturer might evaluate a supplier option without disclosing all its cost assumptions. Several institutions might approve a limited joint calculation without centralising their records. A model owner and a data owner might protect different inputs under a suitable multiparty profile.

The purchasing example offers a straightforward acceptance question: can the agent obtain the agreed decision without receiving the source financial records, while the external worker remains outside the relevant confidentiality boundary?

Evidence before reliance

A reviewer needs a deployment-specific protocol specification, dependency and configuration identities, and evidence that connects the implementation to its stated guarantees. Established primitives do not by themselves prove the security of their composition.

An adversarial campaign should test malformed outputs, substitution and replay before release. It should also test concurrent requests, snapshot restoration, unavailable quorums, custody failures and the race between revocation and commitment. A memory capture without recognisable plaintext is useful evidence about that capture, not a proof that plaintext never existed.

Benchmarks, functional tests and independent cryptographic review answer different questions. Passing one does not close the others. A deployment decision needs security evidence, integration effort and economics to support the same configuration.

Questions for the deployment owner

1. Where can source plaintext, intermediate values and outputs appear, including logs and recovery?
2. Which parties can recover secrets, and what happens if their stated corruption threshold is exceeded?
3. What computation is checked, by which mechanism, and under what soundness and numerical assumptions?
4. What can an operator or recipient infer from metadata and repeated authorised queries?
5. What state survives retries, partitions, migration and lost acknowledgements?
6. What does an accepted result cost when all participating services are included?

SCSA's architectural contribution is the composition of protected execution with bounded authority and governed disclosure. Homomorphic encryption, MPC and attestation remain distinct mechanisms within that design. Their established research foundations do not imply that every combination is supported.

For a first evaluation, freeze one operation, one protection profile and one deployment topology. Publish the acceptance conditions before testing. That gives the customer a result it can evaluate and the engineering team a boundary it can defend.

References and terminology

The references support the background mechanisms and distinctions discussed here. They do not endorse Seven Layer or establish the security or performance of an SCSA implementation.

[1] Confidential Computing Consortium. Common Terminology for Confidential Computing. December 2022. Read source

[2] Microsoft. SEAL documentation and security guidance. Homomorphic arithmetic and application-level security considerations. Read source and Security guidance

[3] Alexander Viand, Christian Knabenhans and Anwar Hithnawi. Verifiable Fully Homomorphic Encryption. 2023. Read source

[4] Marcel Keller and contributors. MP-SPDZ documentation. Protocols with different security and corruption models. Read source

[5] Henk Birkholz et al. Remote ATtestation procedureS Architecture. RFC 9334. January 2023. Read source

[6] Florian Tramer and Dan Boneh. Slalom Fast Verifiable and Private Execution of Neural Networks in Trusted Hardware. ICLR 2019. Read source

Terms used in this paper

Term	Meaning
Admitted computation	A supported calculation and deployment that satisfy the approved contract.
Ciphertext	The encrypted representation on which a supported worker operates.
Custody	Control of the cryptographic material used to recover protected information.
MPC	Multiparty computation under an explicitly selected security model.
Release commitment	The durable decision authorising a particular disclosure.
TEE	A trusted execution environment with a stated hardware and software boundary.

The companion article, What is confidential compute on non confidential silicon, introduces the same mechanisms through the purchase-checking example.