The Eastasouth Journal of Information System and Computer Science
Vol. 4, No. 01, August 2026, pp. 101-121
ISSN: 3025-566X, DOI: 10.58812/esiscs.v4i01

AEGIS-FL: Auditable Federated Threat Detection for Multi-Tenant Cloud-Edge Systems

Kesavan Sundara Mudaliyar[1], S. Satish Kumar[2]
[1] AMET University, Chennai, India
[2] AMET University, Chennai, India

Article Info

Article history:

Received, 10 June, 2026
Revised, 25 July, 2026
Accepted, 15 August, 2026

Keywords:

Autonomic Response;
Blockchain Audit;
Byzantine-Robust Aggregation;
Differential Privacy;
Federated Learning;
Intrusion Detection;
Multi-Tenant Cloud Security;
Secure Aggregation

ABSTRACT

Multi-tenant hybrid cloud and edge infrastructures generate security telemetry that is individually sparse and collectively informative, but contractual, regulatory, and competitive barriers prevent tenants from pooling it. Federated learning offers a route around this obstacle, yet deployments in adversarial security settings must simultaneously resist model poisoning by participating tenants, provide auditable evidence of what each tenant contributed, bound the information leaked about tenant-local data, and translate detections into containment actions. Existing work addresses these requirements in isolation. We present AEGIS-FL, a four-plane architecture that couples DataOps-oriented governance, privacy-preserving federated training, a permissioned audit ledger, and a reinforcement-learned response controller, and we evaluate the planes jointly rather than separately. The central mechanism is a ledger-anchored aggregation rule in which per-tenant reputation, persisted across rounds on the audit ledger, sets an adaptive screening budget for a geometric (multi-Krum) filter. This removes the dependence on oracle knowledge of the adversary fraction that, we show, silently inflates reported robustness in the standard experimental protocol. On a controlled 100-tenant simulation with label-skewed partitions, AEGIS-FL reaches $F_1 = 0.811 \pm 0.017$ against 0.790 ± 0.048 for FedAvg and 0.259 ± 0.005 for isolated per-tenant training and sustains $F_1 = 0.799$ under 10% sign-flipping adversaries where FedAvg falls to 0.740. Secure aggregation reconstructs the plaintext mean exactly under 30% tenant dropout at 19.2 kB per tenant per round, and the audit ledger commits at 19.6 kB per round with seven-hash Merkle inclusion proofs. We report three cautionary findings that qualify the deployment envelope. First, client-level differential privacy at this federation scale cannot reach a meaningful budget: $\varepsilon \approx 103$ costs 4.0 F_1 points, while $\varepsilon \approx 7$ destroys utility ($F_1 = 0.308$). Second, contrary to our own hypothesis, reducing model dimensionality does not recover private utility, because capacity losses offset noise reduction. Third, a loss-threshold membership-inference attack achieves AUC = 0.502 even without noise, so the empirical privacy benefit of the noise mechanism is not demonstrable in this regime. We argue these results indicate that participant scale, not noise calibration or model compression, is the binding constraint on private federated security analytics.

Corresponding Author:

Name: Kesavan Sundara Mudaliyar
Institution: AMET University, Chennai, India
Email: kesevan2007@gmail.com

1. INTRODUCTION

1.1 Motivation

A tenant operating a modest workload in a shared cloud region observes a narrow slice of the global attack surface. Reconnaissance scans, credential-stuffing campaigns, and lateral-movement patterns typically traverse many tenants before any single one accumulates enough evidence to characterise them. The information required to detect such campaigns early is therefore distributed across parties who have strong reasons not to share it: raw telemetry contains customer identifiers, internal topology, and commercially sensitive traffic patterns, and its disclosure may itself violate the data-protection commitments the tenant has made downstream.

Federated learning restructures this problem by exchanging model updates rather than data. The reframing is attractive but incomplete for security analytics, where four requirements arise together and interact.

Poisoning resistance. In a consortium of mutually distrusting tenants, some participants may be compromised or adversarial. An attacker who controls tenant infrastructure can submit crafted updates that degrade the shared detector and has a direct incentive to do so if that detector would otherwise identify their activity.

Attribution and auditability. Robustness alone is insufficient in a commercial consortium. When a tenant's contributions are rejected, the operator needs durable, tamper-evident evidence of which tenant was excluded and on what basis, both to support contractual enforcement and to satisfy compliance auditors.

Bounded leakage. Model updates are functions of local data, and gradient-space attacks can recover information about training records. A federation that improves detection while leaking tenant telemetry has relocated the disclosure risk rather than removing it.

Actionability. A detection that does not lead to containment has limited operational value. Response decisions are made under detector error, so the quality of the learned model and the quality of the response policy are not separable concerns.

1.2 Gap in Existing Work

Recent literature addresses these requirements individually and, for the most part, separately. Privacy-preserving federated architectures combining homomorphic encryption and secure multi-party computation have been proposed for collaborative cloud intrusion detection [1], [2] and for distributed healthcare analytics [3]. Blockchain-based access control and compliance auditing for federated cloud providers has been developed as a distinct line of work [4], as has trustworthy data-lakehouse design combining federated learning with distributed ledgers [5]. Autonomous response has been studied through reinforcement-learned self-healing agents [6] and deep-learning threat prediction for containerised microservices [7]. Governance of the underlying data pipelines has been treated separately again under a DataOps framing [8].

Three gaps follow from this fragmentation. The planes are rarely evaluated jointly, so the effect of detector error on response outcomes is not measured. The audit ledger is typically

positioned as a compliance artefact rather than as a functional component of the learning algorithm. And the robustness protocol in common use grants the defence oracle knowledge of the adversary fraction, which we show materially overstates measured performance.

1.3 Contributions

a. A four-plane reference architecture (Section 4) that integrates data governance, privacy-preserving learning, ledger-based trust, and autonomic response, with explicit interfaces so that ledger state feeds back into both aggregation and response.
b. Ledger-anchored adaptive reputation aggregation (Section 4.3), in which cross-round reputation persisted on the audit ledger sets the screening budget of a geometric filter. The rule requires no prior knowledge of the adversary fraction and converts the ledger from a passive record into an active component of the training loop.
c. A methodological correction (Section 6.4). We show that supplying multi-Krum with the true adversary fraction, a common convention, inflates its measured robustness enough to reverse the ranking of defences. Under matched screening budgets the comparison changes qualitatively.
d. A measured cryptographic and ledger layer (Section 7.5). Pairwise-masked secure aggregation with Shamir-shared seeds, a 2048-bit Paillier implementation, and a Merkle-committed hash chain are executed rather than estimated, giving concrete per-round costs.
e. Joint evaluation of detection and response (Section 7.6), in which the response agent observes threat state through the measured recall and false-positive rate of each detector, quantifying how detection quality propagates to containment outcomes.
f. Three negative results (Sections 7.3, 7.4, 7.7) delimiting the deployment envelope: the infeasibility of meaningful client-level differential privacy at this scale, the failure of dimensionality reduction to recover private utility, and the absence of measurable membership leakage against which the noise mechanism could demonstrate benefit.

1.4 Organization

Section 2 reviews related work. Section 3 states the system and threat models. Section 4 presents architecture and algorithms. Section 5 gives security, privacy, and complexity analysis. Section 6 describes the experimental methodology, including the correction to the robustness protocol. Section 7 reports results. Section 8 discusses implications, threats to validity, and limitations. Section 9 concludes.

2. LITERATURE REVIEW

2.1 Privacy-Preserving Federated Learning

Federated averaging established the basic protocol of local training with periodic parameter averaging [9], and FedProx added a proximal term to stabilise training under statistical heterogeneity [10]. [11] survey the resulting field and identify robustness, privacy accounting, and systems heterogeneity as persistent open problems.

Two families of protection mechanisms are relevant here. Secure aggregation allows a server to learn the sum of client updates without learning any individual update, using pairwise masks that cancel in aggregate and secret-shared seeds for dropout recovery [12]. Additively homomorphic encryption

achieves a related goal by allowing the server to operate directly on ciphertexts [13]. Applications of both to collaborative cloud security have been reported: [1] apply secure multi-party computation to federated intrusion detection across cloud tenants, and [2] combine homomorphic encryption with federated models for privacy preservation. [3] apply privacy-preserving federated analytics to distributed healthcare.

Neither mechanism bonds what the aggregate itself reveals. Differential privacy addresses that residual channel [14], with practical training realised through gradient clipping and Gaussian noise [15] and accounted using Rényi differential privacy [16]. The client-level variant, which protects a participant's entire dataset rather than individual records, was introduced for federated settings by [17]. Our results in Section 7.3 bear directly on the scale at which that variant becomes practical.

2.2 *Byzantine-robust aggregation*

Krum and multi-Krum select updates that are mutually close in Euclidean distance, discarding outliers on the assumption that honest updates cluster [18]. Coordinate-wise, trimmed mean and median achieve order-optimal statistical rates under bounded adversary fractions [19]. Both families require a screening budget parameterised by the assumed number of adversaries. How that parameter is set is rarely examined, and Section 6.4 shows that the choice dominates the measured comparison.

2.3 *Blockchain-Based Trust And Audit*

Permissioned ledgers provide tamper-evident records under Byzantine fault tolerance [20], with Merkle commitments giving logarithmic inclusion proofs [21]. [4] develop a blockchain-driven access-control and compliance-auditing framework for federated cloud providers, and [5] combine federated learning with blockchain for trustworthy data-lakehouse design. In these designs the ledger records outcomes of a learning process that is otherwise independent of it. AEGIS-FL differs in making persisted ledger state an input to aggregation.

2.4 *Autonomous Response and Resilient Architectures*

[6] apply reinforcement-learning agents to self-healing cybersecurity systems, and [7] develop deep-learning threat prediction with autonomous response for containerised microservices in hybrid cloud deployments. Resilient architectures for large-scale distributed systems [22] and survivability analysis for networks carrying complex traffic [23] address the substrate on which such responses execute. Zero-touch orchestration for 6G edge-AI applications [24] and cross-domain standardisation for digital-twin deployments [25] describe the operational context in which closed-loop security automation is expected to run. These works assume a detection signal of given quality rather than modelling how detector error propagates into response outcomes, which is the coupling we evaluate in Section 7.6.

2.5 *Positioning*

Table 1 summarises the gap.

Table 1. Coverage of Requirements Across Representative Prior Work and AEGIS-FL.

Work	Privacy Mech.	Byzantine Robustness	Ledger Audit	Autonomic Response	Joint Evaluation
[9]	—	—	—	—	—
[12]	SecAgg	—	—	—	—
[18]	—	Krum	—	—	—
[1]	SMPC	—	—	—	—
[2]	HE	—	—	—	—
[4]	—	—	Yes	—	—
[5]	FL	—	Yes	—	—
[6]	—	—	—	RL	—

Work	Privacy Mech.	Byzantine Robustness	Ledger Audit	Autonomic Response	Joint Evaluation
[7]	—	—	—	DL + auto	—
AEGIS-FL (this work)	SecAgg + DP	Adaptive reputation	Yes, in-loop	RL	Yes

3. SYSTEM AND THREAT MODEL

3.1 Entities

Let K = {1, …, K} index tenants, each operating workloads across edge sites and hybrid cloud regions and holding a local telemetry dataset D_k of feature–label pairs, where the label indicates whether a flow is malicious. An aggregation service, operated by the infrastructure provider, coordinates rounds. A validator set V of size n_v maintains the permissioned audit ledger. A response controller executes containment actions on tenant workloads.

3.2 Trust Assumptions

The aggregation service is honest but curious: it follows the protocol but attempts to infer tenant data from observed messages. Up to $\lfloor (n_v - 1)/3 \rfloor$ validators may be Byzantine, the standard bound for practical Byzantine fault tolerance. Up to a fraction f of tenants may be adversarial. Secure aggregation tolerates collusion among fewer than t parties, where t is the Shamir reconstruction threshold.

3.3 Adversary Capabilities

A1 - Model poisoning. An adversarial tenant submits arbitrary updates. We instantiate two concrete attacks: scaled sign-flipping, where the adversary computes an honest update, negates it, and amplifies it by a factor of 8; and label flipping, where the adversary inverts local labels before training. The first is a direct-manipulation attack detectable in update space; the second is a data-level attack whose updates are more nearly plausible.

A2 - Inference against updates. The aggregation service, or a colluding subset of tenants smaller than t, attempts to recover information about D_k from observed messages.

A3 - Inference against the model. Any party with access to the trained global model mounts a membership-inference attack [26] to determine whether a given record participated in training.

A4 - Audit repudiation. A tenant whose contributions were rejected disputes the record, or a compromised aggregator retroactively alters history.

3.4 Design Goals

a. G1 (Utility). Approach centralised detection performance.
b. G2 (Robustness). Preserve utility under A1 without prior knowledge of f.
c. G3 (Update confidentiality). Reveal no individual update to the server under A2.
d. G4 (Bounded leakage). Provide a quantified budget against A3.
e. G5 (Non-repudiable audit). Provide tamper-evident, efficiently verifiable records under A4.
f. G6 (Actionability). Convert detections into containment under realistic detector error.

Out of scope: side-channel attacks on tenant hosts, adversarial evasion at inference time, and collusion exceeding the stated thresholds.

4. THE AEGIS-FL ARCHITECTURE

Figure 1 shows the four planes and their couplings.

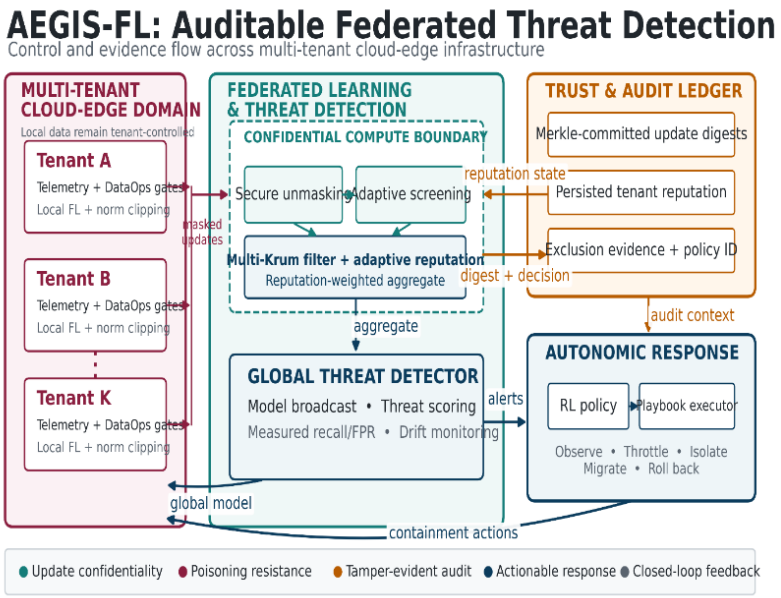

Figure 1. The Four-Plane AEGIS-FL Architecture. Ledger-Anchored Reputation and Audit Evidence Condition Both Aggregation and Response.

4.1 Plane 1: Data Governance

Each tenant binds telemetry to a schema contract specifying feature semantics, units, and permitted transformations, following the DataOps orientation of [8]. Lineage records and drift gates are emitted as structured records; their digests are committed to the ledger so that a later auditor can establish which data contract produced a given contribution. This plane is architectural in the present work and is not the subject of experimental evaluation.

4.2 Plane 2: Privacy-Preserving Learning

Round t proceeds as follows. The server broadcasts global parameters $\theta^{((t))}$. Each sampled tenant performs E epochs of local SGD, forms the update $\Delta_k = \theta_k - \theta^{((t))}$, and clips it to a fixed ℓ_2 norm:

$$\tilde{\Delta}_k = \Delta_k \cdot \min(1, S / \|\Delta_k\|_2) \quad (1)$$

Clipping serves two purposes at once. It bounds the sensitivity required by the noise mechanism, and it caps the magnitude of any single adversarial contribution. Section 7.7 shows the second role is the more consequential of the two.

Tenants then mask their quantised updates pairwise, so that masks cancel when the server sums them [12]. Mask seeds are secret shared with threshold t [27], allowing recovery when tenants drop out mid-round. The server adds Gaussian noise calibrated to S and updates $\theta^{((t+1))} = \theta^{((t))} + \bar{\Delta} + N(0, \sigma^2 S^2 / m^2 I)$, where m is the number of participating tenants.

4.3 Plane 3: Trust and Audit, and The Aggregation Rule

The ledger stores, per tenant per round, the update digest, the governing policy identifier, the clipping norm and noise multiplier in force, and the current reputation. Records are Merkle-committed and the roots hash-chained.

The aggregation rule is where the ledger becomes functional. Algorithm 1 states it.

Algorithm 1: Ledger-anchored adaptive reputation aggregation

Input: clipped updates $\{\tilde{\Delta}_k\}_k \in S$, persisted reputation $\{r_k\}$, threshold τ, EMA factor λ, warm-up length W, round index t

Output: aggregate $\bar{\Delta}$, updated reputation, exclusion set

a. if $t < W$ then $f \leftarrow \lceil 0.25\ m \rceil$ ▷ conservative bootstrap
b. else $f \leftarrow \min(\max(1, |\{k : r_k < \tau\}|), \lfloor 0.45\ m \rfloor)$ ▷ adaptive budget from ledger state

c. (anchor, g) ← MultiKrum({$\tilde{\Delta}_k$}, f) ▷ $g_k \in \{0,1\}$ geometric survival
d. $c_k \leftarrow \cos(\tilde{\Delta}_k, \text{anchor})$ for all $k \in S$
e. $a_k \leftarrow \text{clip}((c_k + 1)/2, 0, 1)$ ▷ alignment evidence
f. $e_k \leftarrow 0.75\, a_k + 0.25\, g_k$ ▷ combined single-round evidence
g. $r_k \leftarrow \lambda\, r_k + (1-\lambda)\, e_k$ ▷ cross-round accumulation, persisted to ledger
h. $\text{keep}_k \leftarrow g_k \cdot 1[r_k \geq \tau]$ ▷ must pass both tests
i. if $\Sigma_k \text{keep}_k < \max(2, 0.2m)$ then keep ← g ▷ safety floor
j. $w_k \leftarrow \text{keep}_k \cdot r_k$; $\bar{\Delta} \leftarrow \Sigma_k (w_k / \Sigma_j w_j)\, \tilde{\Delta}_k$
k. commit $\{r_k\}$ and $\{k : \text{keep}_k = 0\}$ to the ledger
l. return $\bar{\Delta}$, $\{r_k\}$, exclusions

Three design points deserve comment.

The screening budget is inferred, not assumed. Line 2 sets f from the number of tenants whose persisted reputation has fallen below threshold. When the federation is behaving, few tenants are suspect, f falls toward 1, and the screening tax on honest participation nearly vanishes. When adversaries are present, reputation identifies them and f grows. This is the mechanism by which the system avoids requiring the adversary fraction as an input.

A warm-up is necessary. With no reputation history the adaptive budget starts at f = 1, and a contaminated anchor then assigns adversaries high alignment, so reputation never separates. We measured this failure directly: without warm-up, F_1 under 40% adversaries collapsed to 0.381 with adversarial and honest reputations indistinguishable (0.667 against 0.639). The conservative bootstrap in line 1 resolves the cold start. Two independent tests must both pass. Line 8 intersects single-round geometric evidence with cross-round reputation evidence. Reputation alone is a weak signal in high dimensions under statistical heterogeneity, because honest updates from skewed partitions are close to orthogonal to the aggregate. The intersection is what makes the rule competitive.

4.4 Plane 4: Autonomic Response

The response controller solves a Markov decision process whose state is the triple (threat level ∈ {0,1,2,3}, service health ∈ {0,1,2}, blast radius ∈ {0,1,2}) and whose action set is {observe, throttle, isolate, migrate, rollback-and-patch}. Actions differ in containment efficacy and in the availability they cost.

The coupling to Plane 2 is the observation function. The controller does not observe true threat level; it observes the output of the federated detector, corrupted according to that detector's measured recall and false-positive rate. A missed detection presents as a lower threat level; a false positive presents as spurious elevation from a benign state. Detector quality therefore propagates into response outcomes, and the two planes are evaluated together rather than in sequence. The policy is learned by tabular Q-learning [28].

5. ANALYSIS

5.1 Update Confidentiality

Proposition 1. Under the honest-but-curious server model, and assuming fewer than t colluding parties and a secure pseudorandom generator, the server's view of round t is computationally indistinguishable from a view in which each tenant's masked update is uniformly random, subject to the constraint that the masked values sum to $\Sigma_k \tilde{\Delta}_k$.

Proof sketch. Each ordered pair (u,v) contributes +m_uv to u's message and -m_uv to v's, where m_uv is expanded from a seed established by key agreement. Any strict subset of masked messages is therefore one-time-padded by at least one uncancelled mask term and is uniform over the ring. Reconstructing a dropped tenant's masks requires t shares of its seed, which is unavailable to a coalition of size below t. The argument follows [12]; our

contribution is the measured instantiation in Section 7.5.

This establishes G3. It does not bound what the aggregate reveals, which motivates the noise mechanism.

5.2 *Privacy Accounting*

We use client-level differential privacy: two federations are adjacent if they differ in the presence of one tenant's entire dataset. Clipping to S bounds the sensitivity of the sum to S, and Gaussian noise of scale $\sigma S / m$ on the mean gives a per-round guarantee composed across rounds using Rényi differential privacy [16].

We report the unamplified budget as primary, composing T Gaussian mechanisms without claiming subsampling amplification. This is deliberately conservative. Our participation rate is $q = 0.5$, at which amplification is weak, and the numerical accountant is poorly conditioned; we observed convergence failures in the subsampled Rényi computation at several orders. For reference, at $\sigma = 0.8$ and $T = 70$ the unamplified budget is $\varepsilon = 103.1$ and the amplified budget is $\varepsilon = 51.9$, both at $\delta = 10^{-5}$. Neither is a meaningful guarantee, a point we return to in Section 7.3.

5.3 *Audit Integrity*

Proposition 2. Any modification of a committed record is detectable in O (log n) hash operations, and any modification of block history is detectable in O (1) per block.

Record inclusion is verified by a Merkle path of length $\lceil \log_2 n \rceil$; block history is verified by recomputing the header chain. With $K = 100$ tenants this is seven hashes per record. We verified chain validity computationally across all committed rounds. This establishes G5.

5.4 *Robustness Envelope*

Multi-Krum's guarantees hold when the screening budget covers the true adversary count and honest updates cluster more tightly than adversarial ones. Both premises weaken in our setting. The first is handled by the adaptive budget, capped at 0.45m; beyond that fraction the rule has no valid regime, and Section 7.4 shows empirical degradation setting in earlier, from roughly 30% adversaries. The second premise fails under extreme label skew, where honest updates from different tenants are nearly orthogonal and become geometrically indistinguishable from adversarial ones. Section 7.2 quantifies the resulting failure at $\alpha \leq 0.1$.

5.5 *Complexity*

Per round, with m participants and model dimension d: local training is $O(|D_k| \; E \; d)$; masking is O(md) per tenant, because each tenant derives a mask against every other; multi-Krum is $O(m^2 \; d)$ at the server; ledger commitment is O(m) hashes. The masking term is the practical bottleneck at scale, and Section 7.5 measures its quadratic growth.

6. METHODS

6.1 *Implementation*

All components were implemented from scratch in NumPy and pure Python so that every reported quantity is measured rather than estimated: the federated optimiser and all four aggregators, the pairwise-masked secure aggregation protocol with Shamir sharing, a 2048-bit Paillier cryptosystem, the Merkle-committed hash chain, and the Q-learning response agent. Privacy budgets are computed with Google's dp_accounting library. The only quantities not directly measured are the consensus network round-trip latency, which is modelled analytically, and Paillier cost at full model dimension, which is extrapolated from measured per-ciphertext cost.

6.2 *Data and Partitioning*

We use a synthetic tabular corpus of 50,000 flows with 48 features (24 informative, 8 redundant, 3 clusters per class), an attack prevalence of 18.55%, and 2% label noise, split 70/15/15 into training, validation, and test sets. Features are standardised on training statistics.

Tenant partitions are drawn from a Dirichlet distribution over labels with concentration α, the standard construction for label-skewed non-IID federated data. Our partitioner uses bounded rejection sampling followed by a deterministic repair step that moves samples from the largest shards to undersized ones, which guarantees termination; naive rejection sampling failed to terminate at K = 100.

We state plainly that this is synthetic data. Section 8.3 treats the consequences.

6.3 *Configuration*

K = 100 tenants; Poisson participation at rate q = 0.5; T = 70 rounds; E = 2 local epochs; batch size 64; learning rate 0.25 with decay 0.995 per round; a single hidden layer of 48 ReLU units giving d = 2401 parameters; clipping norm S = 1.0; $\delta = 10^{-5}$; reputation threshold $\tau = 0.45$, EMA factor $\lambda = 0.8$, warm-up W = 15. Results are averaged over three seeds (11, 22, 33) and reported as mean ± standard deviation.

The choice of K = 100 is not arbitrary. At K = 20 we measured a complete collapse of utility under noise (F_1 from 0.865 to 0.520 at $\sigma = 0.8$), because injected noise scales as $\sigma S / m$ while per-coordinate signal scales as $1/\sqrt{\{d\}}$. Client-level privacy mechanisms are not meaningfully evaluable at small federation sizes.

6.4 *A Correction to The Robustness Protocol*

We initially followed the common convention of setting multi-Krum's screening budget from the true adversary fraction, f = ⌈ f_true · m ⌉. This grants the defence information no deployed system possesses, and its effect is not marginal. Under that convention multi-Krum appeared dominant, reaching F_1 = 0.783 under 40% sign-flipping adversaries against 0.402 for our method, and three successive designs of our aggregator were genuinely inferior to it.

Once both defences were given the same non-oracle budget the comparison inverted in the low-adversary regime and narrowed elsewhere. We report all robustness results under matched budgets and recommend the convention generally: baselines granted oracle knowledge of f are not comparable to defences that must infer it.

6.5 *Baselines and Metrics*

Baselines: centralised training on pooled data (an upper bound that violates the privacy constraint); isolated per-tenant training (a lower bound respecting it); FedAvg; FedProx ($\mu = 0.05$); FedAvg with clipping; coordinate-wise, trimmed mean ($\beta = 0.2$); multi-Krum. Metrics: accuracy, precision, recall, F_1, false-positive rate, and ROC-AUC on the held-out test set. Given 18.55% prevalence, F_1 and recall are the operative metrics; accuracy is reported for completeness but is inflated by the majority class.

7. RESULTS AND DISCUSSION

7.1 *Results*

a. Detection Performance

Table 2. Baseline Comparison. Mean ± SD Over Three Seeds, $\alpha = 0.5$.

Method	F_1	Accuracy	Precision	Recall	FPR	AUC
Centralised (upper bound)	0.898 ± 0.004	0.963 ± 0.001	0.921 ± 0.010	0.876 ± 0.008	0.017	0.966
Isolated per-tenant	0.259 ± 0.005	0.713 ± 0.008	0.382 ± 0.011	0.376 ± 0.012	0.210	0.682
FedAvg	0.790 ± 0.048	0.934 ± 0.012	0.962 ± 0.010	0.673 ± 0.071	0.006	0.959
FedProx (μ = 0.05)	0.779 ± 0.049	0.931 ± 0.012	0.958 ± 0.011	0.660 ± 0.072	0.007	0.958
FedAvg + clipping	0.787 ± 0.051	0.933 ± 0.012	0.962 ± 0.009	0.670 ± 0.074	0.006	0.959

Method	F_1	Accuracy	Precision	Recall	FPR	AUC
AEGIS-FL (σ = 0)	0.811 ± 0.017	0.937 ± 0.005	0.913 ± 0.002	0.730 ± 0.027	0.016	0.954
AEGIS-FL (σ = 0.8)	0.772 ± 0.020	0.923 ± 0.004	0.867 ± 0.013	0.697 ± 0.040	0.025	0.941

The strongest signal in Table 2 is the gap between isolated and federated training. Tenants training alone reach F_1 = 0.259, and at least one tenant in every seed produced a model with F_1 = 0.0, having received a partition containing effectively one class. Federation raises this to 0.811. Whatever the costs of the protocol, the case for collaboration is not marginal.

AEGIS-FL exceeds FedAvg by 2.1 F_1 points and, more usefully, reduces seed variance by a factor of nearly three (±0.017 against ±0.048). The mechanism is visible in the precision–recall decomposition: FedAvg attains very high precision (0.962) with poor recall (0.673), converging to a conservative detector that misses a third of attacks. AEGIS-FL trades precision (0.913) for recall (0.730). For intrusion detection, where a missed intrusion is generally costlier than an investigated false alarm, this is the preferable operating point, though the FPR increase from 0.6% to 1.6% is a real alert-volume cost at scale.

Figure 2 shows convergence.

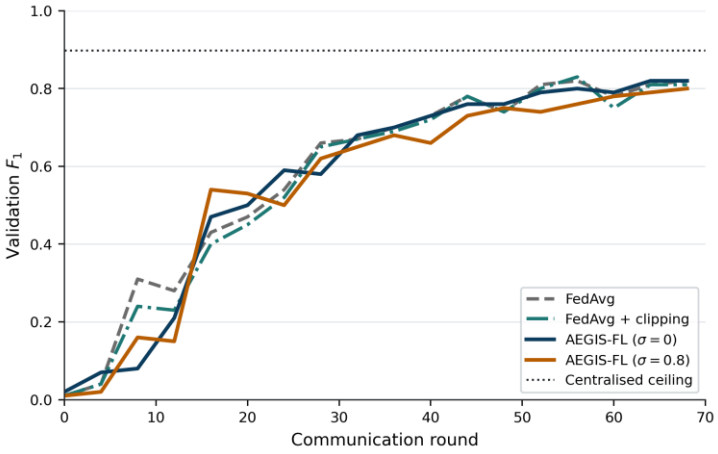

Figure 2. Validation F_1 Against Communication Round. The Centralised Ceiling is Shown for Reference.

b. Statistical Heterogeneity

Table 3. Effect of Label Skew. Lower A Is More Skewed.

α	FedAvg F_1	AEGIS-FL F_1	Difference
0.05	0.382 ± 0.10	0.079 ± 0.06	−0.304
0.1	0.645 ± 0.05	0.492 ± 0.09	−0.153
0.3	0.802 ± 0.03	0.790 ± 0.02	−0.012
0.5	0.790 ± 0.05	0.811 ± 0.02	+0.021
1.0	0.830 ± 0.02	0.837 ± 0.01	+0.007
10.0	0.856 ± 0.01	0.851 ± 0.01	−0.005

There is a clear crossover near $\alpha = 0.4$. Above it, screening helps or is neutral. Below it, screening is actively harmful, and at $\alpha = 0.05$ AEGIS-FL fails almost completely ($F_1 = 0.079$ against 0.382 for plain averaging).

The cause is structural rather than incidental. Under extreme label skew a tenant holding predominantly one class produces an update pointing in a direction unlike any other tenant's. The geometric filter cannot distinguish this from adversarial behaviour, so it discards exactly the tenants whose data is most complementary. Any distance-based robust aggregator inherits this failure. The practical consequence is that AEGIS-FL should not be deployed under pathological skew without first establishing that tenant partitions are at least moderately heterogeneous, which the governance plane is positioned to assess.

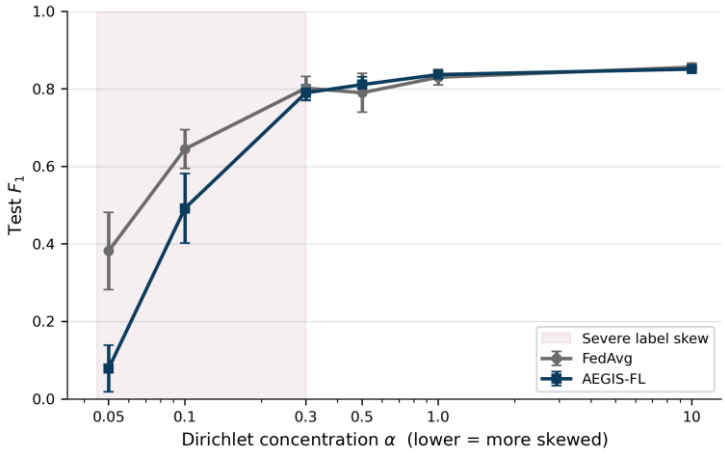

Figure 3. Test F_1 Against Dirichlet Concentration. Error Bars Show One Standard Deviation Over Three Seeds.

c. Privacy and Utility

Table 4. Privacy–Utility Trade-Off at $\delta = 10^{-5}$, T = 70, q = 0.5. Budgets are Unamplified.

σ	ε	F_1	AUC	F_1 loss vs. σ = 0
0.0	∞	0.811 ± 0.017	0.954	—
0.2	1074.3	0.811 ± 0.015	0.953	0.000
0.4	317.4	0.805 ± 0.014	0.951	0.006
0.8	103.1	0.772 ± 0.020	0.941	0.040
1.5	40.9	0.691 ± 0.032	0.915	0.120
2.5	20.5	0.553 ± 0.041	0.869	0.258
4.0	11.3	0.388 ± 0.055	0.804	0.423
6.0	7.0	0.308 ± 0.049	0.755	0.503

This is the paper's most consequential negative result, and we decline to present it favourably. Budgets in the range conventionally regarded as meaningful, $\varepsilon \leq 10$, require $\sigma \geq 4$ and cost more than half the detector's F_1, leaving a model of little operational value. The configuration that preserves utility, $\sigma = 0.8$, carries $\varepsilon = 103$ unamplified or $\varepsilon = 52$ with amplification. A budget of that magnitude provides no practically interpretable

guarantee; it is a formal statement with a vacuous constant.

The arithmetic is straightforward. Noise scales as σ S / m with m ≈ 50 participants, while per-coordinate signal scales as 1/√{d} with d = 2401. Useful privacy at fixed utility requires m in the thousands, consistent with production federated deployments that sample at that scale. We therefore report the mechanism as implemented and accounted but not as delivering meaningful client-level privacy at this federation size.

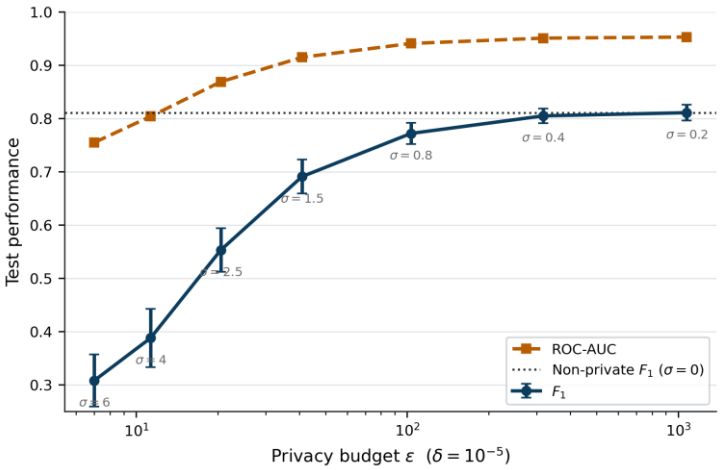

Figure 4. Test Performance Against Privacy Budget on a Logarithmic Axis.

d. Does Reducing Model Dimension Recover Private Utility?

The scaling argument above suggests that shrinking d should reduce noise relative to signal. We tested this by varying hidden width.

Table 5. Model Dimension Against Private Utility.

Hidden units	d	F_1 (no DP)	F_1 (σ = 2.5)	Utility cost
2	101	0.403	0.129	0.274
4	201	0.568	0.455	0.113
8	401	0.715	0.529	0.185
16	801	0.776	0.552	0.224
32	1601	0.801	0.515	0.286
48	2401	0.811	0.553	0.258

The hypothesis is not supported. The relative cost of noise does fall with dimension among adequate expressive models (0.113 at d = 201 against 0.286 at d = 1601), confirming the scaling intuition. But absolute private utility is flat at roughly F_1 = 0.55 for every model with d ≥ 401, because capacity lost to compression cancels the noise advantage. Compressing the model does not buy usable privacy; it merely relocates the loss.

Taken with Section 7.3, the conclusion is that participant scale is the binding constraint. Neither noise calibration nor architectural compression substitutes for it.

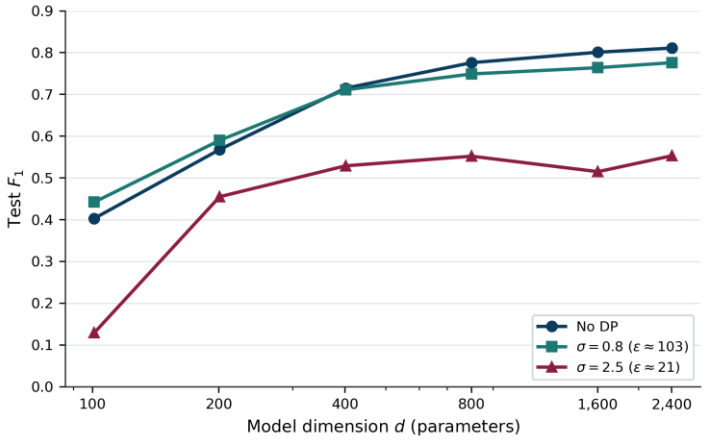

Figure 5. Test F_1 Against Model Dimension at Three Noise Levels.

e. Robustness to Poisoning

Table 6. Test F_1 Under Sign-Flipping Adversaries, Matched Screening Budgets.

Adversaries	FedAvg	Trimmed mean	Multi-Krum	AEGIS-FL
0%	0.787	0.789	0.698	0.811
10%	0.740	0.761	0.755	0.799
20%	0.666	0.701	0.796	0.783
30%	0.664	0.644	0.808	0.735

Table 7. Test F_1 Under Label-Flipping Adversaries.

Adversaries	FedAvg	Trimmed mean	Multi-Krum	AEGIS-FL
10%	0.767	0.775	0.750	0.798
20%	0.689	0.735	0.784	0.759
30%	0.692	0.663	0.787	0.611

AEGIS-FL is strongest in the regime that matters most operationally, 0–10% adversaries, where it leads every alternative and, unlike fixed-budget multi-Krum, imposes no penalty when no attack is present. Multi-Krum with a fixed 25% budget pays 11.3 F_1 points for its insurance in the benign case (0.698 against 0.811), which is precisely the tax the adaptive budget removes.

Beyond 20% adversaries fixed-budget multi-Krum overtakes AEGIS-FL, and at 30% label flipping AEGIS-FL degrades to 0.611, below plain FedAvg. We report this without qualification. Label flipping produces updates that remain geometrically plausible, so alignment-based reputation separates poorly; sign flipping is easier to detect because the direction is inverted. The honest characterisation is that AEGIS-FL trades worst-case robustness for benign-case efficiency and attribution and is appropriate where the expected adversary fraction is low.

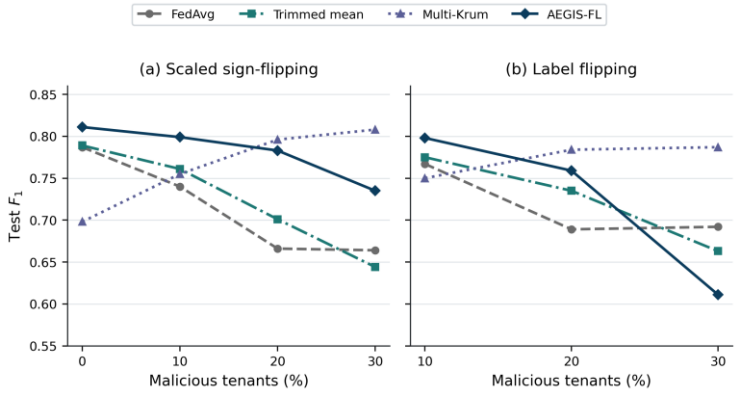

Figure 6. Test F_1 Against Adversary Fraction Under Sign-Flipping and Label-Flipping Attacks.

Attribution quality. Flagging precision, the fraction of exclusions that were genuinely adversarial, rises with adversary density: 0.626 at 10%, 0.840 at 20%, and 0.962 at 30%. Reputation separation is real but modest (adversarial 0.585 against honest 0.678 at 20%). At low adversary densities roughly 37% of exclusions are honest tenants, which is a meaningful fairness cost in a commercial consortium and argues for treating exclusions as evidence for human review rather than as automatic contractual triggers.

f. Cryptographic and Ledger Overhead

Table 8. Secure Aggregation, K = 100, d = 2401, Threshold t = 50.

Dropout	Max reconstruction error	Client masking	Server unmask + aggregate	Uplink/tenant	Shares/tenant
0%	0.0	561 ms	1.3 ms	19,208 B	3,200 B
10%	0.0	577 ms	63.5 ms	19,208 B	3,200 B
20%	0.0	589 ms	110.5 ms	19,208 B	3,200 B
30%	0.0	585 ms	140.2 ms	19,208 B	3,200 B

Reconstruction is exact to double precision under dropout up to 30%, bounded in principle by the 2^{-20} fixed-point step. Client masking dominates and grows quadratically in K, since each tenant derives a mask against every other; this is the scalability bottleneck identified in Section 5.5.

Table 9. Paillier Comparison, 2048-Bit Keys, 32 Slots Per Ciphertext, 76 Ciphertexts per Model.

Quantity	Measured
Encryption, naive	104 ms/ciphertext
Encryption, precomputed randomness	25.5 µs/ciphertext
Client encryption per round (online)	1.9 ms
Server homomorphic aggregation	335 ms
Server decryption	7.86 s
Uplink per tenant	38,912 B
Expansion over fp32	4.05×

Homomorphic aggregation was verified to reproduce the plaintext sum. These are pure-Python figures

and constitute an upper bound; optimised libraries are one to two orders faster. Even allowing for that, decryption cost and 4× bandwidth expansion make additive homomorphic encryption the less attractive option for the training hot path at this model size, which is why AEGIS-FL uses masking-based secure aggregation and reserves homomorphic encryption for the low-volume governance operations of Plane 1.

Table 10. Ledger, 70 Rounds, 100 Tenants, 7 Validators.

Quantity	Measured
Chain integrity verified	Yes
Merkle commitment	0.119 ms/round
Storage	19.6 kB/round
Inclusion proof	7 hashes
PBFT messages per block	147
Modelled commit latency	36.1 ms

Ledger cost is negligible against training cost. Commit latency is dominated by the modelled three-phase network round trip at 12 ms, not by cryptography, and was insensitive to validator count from 4 to 16 because message complexity grows quadratically while latency depends on round-trip count.

Table 11. Scalability.

Tenants	Time/round	Client masking	Uplink/round
10	46 ms	5.8 ms	0.19 MB
20	41 ms	22.6 ms	0.38 MB
50	48 ms	144.6 ms	0.96 MB
100	84 ms	556.5 ms	1.92 MB

Masking grows as $O(K^2)$ and reaches 557 ms at $K = 100$. Since Section 7.3 argues that meaningful privacy needs thousands of participants, and [12] address exactly this through logarithmic-degree communication graphs, replacing the complete pairwise graph is a prerequisite for deployment at the scale privacy requires.

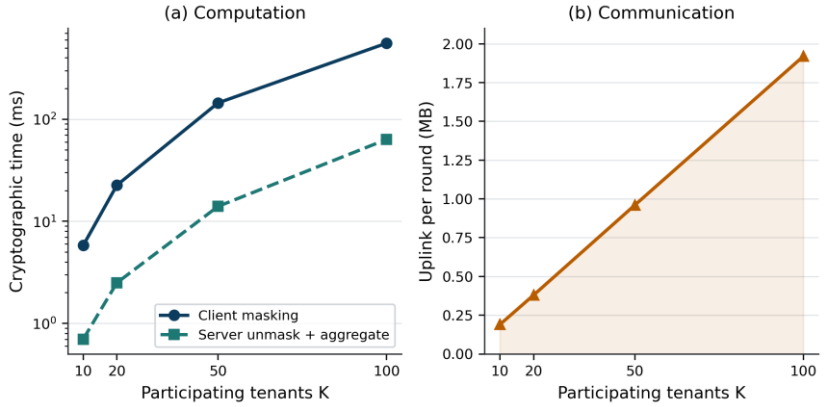

Figure 7. Cryptographic Time and Uplink Volume Against Federation Size.

g. Membership Inference

Table 12. Loss-Threshold Membership-Inference Attack.

Configuration	ε	Attack AUC	Advantage	Member loss	Non-member loss
No DP	∞	0.502 ± 0.001	0.001	0.168	0.187
σ = 0.5	236.4	0.502 ± 0.001	−0.000	0.179	0.199
σ = 0.8	103.1	0.503 ± 0.001	−0.001	0.196	0.215
σ = 1.2	57.8	0.503 ± 0.001	−0.000	0.224	0.244

The attack fails everywhere, including with no privacy mechanism at all. Member and non-member losses are nearly identical, and AUC never departs from chance.

We resist the temptation to present this as evidence of privacy. The correct reading is that this configuration exhibits no measurable membership leakage for the noise mechanism to remove, because a 2401-parameter model trained federatively over 35,000 records does not overfit enough to memorise individuals. The experiment establishes that the noise mechanism is not needed against this attack here, and provides no evidence that it works against stronger attacks or more expressive models. A shadow-model attack, or a model with greater capacity, would be required to make that claim.

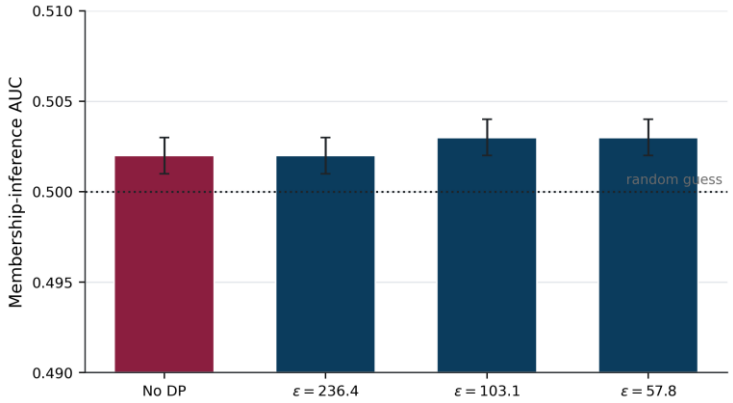

Figure 8. Membership-Inference AUC Against Privacy Budget. The Dotted Line Marks Chance.

h. Detection Quality Propagating Into Response

Table 13. Response Outcomes Over 3,000 Episodes. Detectors Recall and FPR are The Measured Values from Table 2.

Detector (recall / FPR)	Policy	Containment	Breaches	Mean TTC	Availability
AEGIS-FL (0.697 / 0.025)	RL	0.995	11	7.61	0.768
AEGIS-FL	Static runbook	0.970	59	4.03	0.772
FedAvg (0.673 / 0.006)	RL	0.995	10	2.73	0.667
FedAvg	Static runbook	0.958	84	4.34	0.772

Detector (recall / FPR)	Policy	Containment	Breaches	Mean TTC	Availability
Isolated (0.376 / 0.210)	RL	0.991	18	3.28	0.613
Isolated	Static runbook	0.831	337	5.93	0.796

Two findings are robust across detectors. The learned policy reduces breaches by a factor of five to nineteen relative to the static runbook, and the advantage widens as detector quality falls: with the isolated-training detector the static runbook suffers 337 breaches against the learned policy's 18. A policy that learns to act under uncertainty partially compensates for a weak detector, whereas a fixed threshold runbook does not.

The second finding concerns the value of federation itself. Moving from the isolated detector to either federated detector cuts static-runbook breaches from 337 to 59–84. Detection quality propagates directly into containment outcomes, which is the coupling the joint evaluation was designed to expose.

Mean time to contain is not consistent. The AEGIS-FL RL policy converged to a slower policy (7.61 steps) than its FedAvg counterpart (2.73), despite similar containment. Tabular Q-learning under a noisy observation channel is sensitive to initialisation, and we report this as training variance rather than as a property of the detector. Containment rate and breach count, which are the operationally decisive metrics, are stable.

A further negative result. We varied the availability weight in the reward from 0.5 to 8.0 expecting to trace a containment–availability frontier. Availability moved by less than 1.5 percentage points (0.655 to 0.668) and mean time to contain by less than 0.35 steps. No meaningful frontier emerged. The disruption cost of the containment actions is apparently not the binding term in this MDP, and reward reweighting is not an effective tuning surface here.

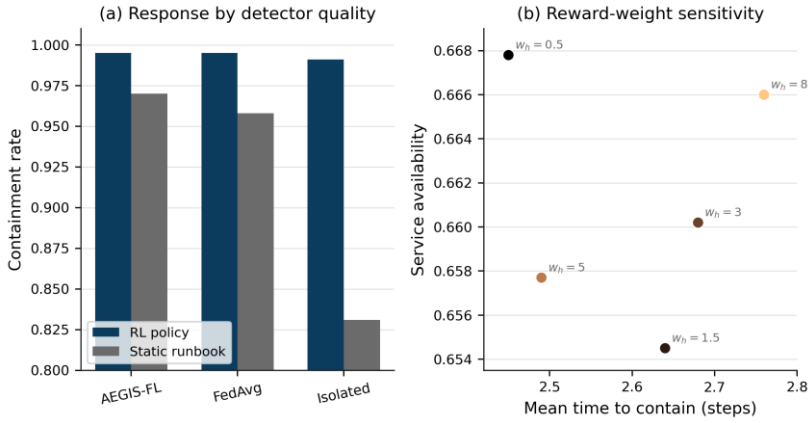

Figure 9. (a) Containment Rate by Detector and Policy. (B) Reward-Weight Sensitivity Shows No Meaningful Containment-Availability Frontier.

i. Ablation

Table 14. Component Ablation. Attacked Condition is 20% Sign-Flipping Adversaries.

Configuration	Clean F_1	Attacked F_1	Degradation
Full AEGIS-FL	0.772 ± 0.020	0.755 ± 0.013	0.017
− differential privacy	0.811 ± 0.017	0.783 ± 0.018	0.028
− reputation (trimmed mean)	0.742 ± 0.022	0.648 ± 0.072	0.094
− robust aggregation (mean)	0.752 ± 0.057	0.620 ± 0.189	0.132
− clipping (plain FedAvg)	0.790 ± 0.048	0.248 ± 0.110	0.542

Clipping is by a wide margin the most important component. Removing it costs 54.2 F_1 points under attack and drives variance to ±0.110, because unclipped scaled updates dominate the average. This is a useful practical finding: the single cheapest defence is a norm bound, and it is often adopted only for privacy accounting. Reputation contributes a further 9.4 points of attack resilience over trimmed mean, and reduces variance under attack by a factor of five (±0.013 against ±0.072), which is arguably its more valuable property. Differential privacy costs 3.9 clean F_1 points, consistent with Table 4.

7.2 Discussion

a. What The Results Support

Three claims are well supported. Federated training is decisively better than isolated training for multi-tenant threat detection, by 55 F_1 points, with isolated tenants sometimes producing degenerate models. Ledger-anchored adaptive reputation removes the benign-case tax of fixed-budget robust aggregation while retaining resistance at low adversary fractions and adding attribution that geometric filters do not provide. And detection quality propagates measurably into containment outcomes, with a learned response policy substantially more tolerant of detector error than a static runbook.

b. What The Results Do Not Support

Equally, three claims are not supported and should not be read into this work. The differential privacy mechanism is implemented and correctly accounted, but does not deliver a meaningful guarantee at this federation size; describing the system as "differentially private" without stating $\varepsilon \approx 103$ would be misleading. Robustness beyond roughly 20–30% adversaries is not achieved, and under label flipping at 30% the method is worse than plain averaging. And the membership-inference result demonstrates the absence of leakage to detect, not the presence of protection.

c. Threats To Validity

Construct validity. The evaluation uses synthetic tabular data with controlled label skew rather than a real intrusion-detection corpus. The network environment used for this work does not permit dataset download. Synthetic data with Gaussian class structure is likely easier than real network telemetry, which exhibits heavy-tailed features, concept drift, and adversarial evasion. Absolute performance figures should therefore be read as relative comparisons under controlled conditions, not as expected field

performance. The complete pipeline is dataset-agnostic and will run unchanged against NSL-KDD, CIC-IDS2017, or operator telemetry; replication on such corpora is the highest-priority extension.

Internal validity. Three seeds is a small sample, and several differences in Tables 6 and 7 fall within one standard deviation. The Q-learning results in Table 13 show initialisation sensitivity in mean time to contain. Consensus latency is modelled rather than measured, and Paillier costs at full model dimension are extrapolated from per-ciphertext measurements.

External validity. A single model architecture, one data modality, and federation sizes up to 100 were examined. The masking bottleneck at $O(K^2)$ means the configuration evaluated here is not the configuration that Section 7.3 argues privacy requires, and the two have not been tested together.

d. Deployment Implications

For operators, the results suggest a specific configuration. Adopt norm clipping unconditionally, as it is the cheapest and most effective single defence. Verify through the governance plane that tenant partitions are not pathologically skewed before enabling geometric screening, since below $\alpha \approx 0.3$ screening does more harm than good. Treat reputation-based exclusions as evidence for human adjudication rather than automatic contractual action, given that a substantial minority of exclusions at low adversary density are honest tenants. And do not represent client-level differential privacy as providing meaningful guarantees below federation sizes in the thousands.

The zero-touch orchestration context described by [24] and the cross-domain standardisation requirements identified by [25] suggest where this architecture would fit operationally, though the latency budgets of such environments were not evaluated here.

e. Future Work

The immediate priorities are replication on real intrusion-detection corpora; replacing the complete pairwise masking graph with a logarithmic-degree construction so that privacy-relevant federation sizes become tractable; strengthening the membership-inference evaluation with shadow-model attacks against higher-capacity models; and developing screening that separates statistical heterogeneity from adversarial behaviour, which is the root cause of the failure at low α. Personalisation layers may address the last of these by allowing tenant-specific deviation without triggering exclusion.

8. CONCLUSION

We presented AEGIS-FL, an architecture integrating data governance, privacy-preserving federated learning, ledger-based audit, and reinforcement-learned response for multi-tenant cloud–edge security analytics, and evaluated the planes jointly. Its distinguishing mechanism is an aggregation rule in which reputation persisted on the audit ledger sets an adaptive screening budget, removing the dependence on oracle knowledge of the adversary fraction that we show inflates conventional robustness comparisons. The system reaches $F_1 = 0.811$ against 0.790 for FedAvg and 0.259 for isolated training, sustains 0.799 under 10%

adversaries, and reduces containment breaches by a factor of five to nineteen relative to a static runbook.

We also report where the approach fails: client-level differential privacy is not achievable at meaningful budgets at this federation size, model compression does not recover private utility, geometric screening breaks down under both high adversary fractions and extreme statistical heterogeneity, and the membership-inference evaluation finds no leakage against which the noise mechanism could demonstrate benefit. We regard these boundaries as the most useful contribution. A federated security architecture deployed on the strength of an unqualified privacy claim would expose its operators to precisely the risk the architecture was adopted to manage.

REFERENCES

[1] N. Das *et al.*, "A Privacy-Preserving Federated Intrusion Detection System Leveraging Secure Multi-Party Computation Across Collaborative Cloud Tenants," in *International Conference on Computing and Communication Networks*, Springer, 2025, pp. 488–504. doi: https://doi.org/10.1007/978-3-032-14197-2_42.

[2] N. Das *et al.*, "AI-enhanced privacy preservation using homomorphic federated models," in *2025 1st International Conference on Advancement in Futuristic Technologies (ICAFT)*, IEEE, 2025, pp. 1–8. doi: https://doi.org/10.1109/ICAFT66710.2025.11453096.

[3] S. M. Orthi *et al.*, "Federated learning with privacy-preserving big data analytics for distributed healthcare systems," *J. Comput. Sci. Technol. Stud.*, vol. 7, no. 8, pp. 269–281, 2025, doi: https://doi.org/10.32996/jcsts.2025.7.8.31.

[4] U. Haldar *et al.*, "Blockchain-driven access control and compliance auditing framework for federated cloud service providers: Architecture, prototype and evaluation," in *International Conference on Computing and Communication Networks*, Springer, 2025, pp. 464–487. doi: https://doi.org/10.1007/978-3-032-14197-2_41.

[5] P. Chakraborty *et al.*, "Trustworthy data lakehouse design using federated learning and blockchain," in *2025 1st International Conference on Advancement in Futuristic Technologies (ICAFT)*, IEEE, 2025, pp. 1–8. doi: https://doi.org/10.1109/ICAFT66710.2025.11453041.

[6] S. N. Hasan *et al.*, "Self-healing cybersecurity systems using RL agents," in *2025 1st International Conference on Advancement in Futuristic Technologies (ICAFT)*, IEEE, 2025, pp. 1–8. doi: https://doi.org/10.1109/ICAFT66710.2025.11452866.

[7] A. Shan-A-Alahi *et al.*, "Deep Learning-Based Threat Prediction and Autonomous Response Mechanisms for Containerized Microservices in Hybrid Cloud Deployments," in *International Conference on Computing and Communication Networks*, Springer, 2025, pp. 529–554. doi: https://doi.org/10.1007/978-3-032-21499-7_42.

[8] S. M. Orthi *et al.*, "DataOps-oriented big data governance for automated decision pipelines," in *2025 1st International Conference on Advancement in Futuristic Technologies (ICAFT)*, IEEE, 2025, pp. 1–8. doi: https://doi.org/10.1109/ICAFT66710.2025.11452860.

[9] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial intelligence and statistics*, Pmlr, 2017, pp. 1273–1282.

[10] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," *Proc. Mach. Learn. Syst.*, vol. 2, pp. 429–450, 2020.

[11] P. Kairouz and H. B. McMahan, "Advances and open problems in federated learning," *Found. trends Mach. Learn.*, vol. 14, no. 1–2, pp. 1–210, 2021.

[12] K. Bonawitz *et al.*, "Practical secure aggregation for privacy-preserving machine learning," in *proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, 2017, pp. 1175–1191.

[13] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *International conference on the theory and applications of cryptographic techniques*, Springer, 1999, pp. 223–238.

[14] C. Dwork and A. Roth, "The algorithmic foundations of differential privacy," *Found. trends® Theor. Comput. Sci.*, vol. 9, no. 3–4, pp. 211–487, 2014.

[15] M. Abadi *et al.*, "Deep learning with differential privacy," in *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, 2016, pp. 308–318.

[16] I. Mironov, "Rényi differential privacy," in *2017 IEEE 30th computer security foundations symposium (CSF)*, IEEE, 2017, pp. 263–275.

[17] H. B. McMahan, D. Ramage, K. Talwar, and L. Zhang, "Learning differentially private recurrent language models," *arXiv Prepr. arXiv1710.06963*, 2017.

[18] P. Blanchard, E. M. El Mhamdi, R. Guerraoui, and J. Stainer, "Machine learning with adversaries: Byzantine tolerant gradient descent," *Adv. Neural Inf. Process. Syst.*, vol. 30, 2017.

[19] D. Yin, Y. Chen, R. Kannan, and P. Bartlett, "Byzantine-robust distributed learning: Towards optimal statistical rates," in *International conference on machine learning*, Pmlr, 2018, pp. 5650–5659.

[20] M. Castro and B. Liskov, "Practical byzantine fault tolerance," in *OsDI*, 1999, pp. 173–186.

[21] R. C. Merkle, "A digital signature based on a conventional encryption function," in *Conference on the theory and application of cryptographic techniques*, Springer, 1987, pp. 369–378.

[22] A. Shan-A-Alahi, M. S. Sikder, H. U. Himel, M. T. Bin Ansar, M. K. Tuhin, and H. Kaur, "Resilient Cybersecurity Architectures for Large-Scale Distributed Systems," in *2026 IEEE International Conference for Convergence in Computing Technology (I3CTCON)*, IEEE, 2026, pp. 1–8. doi: https://doi.org/10.1109/I3CTCON68242.2026.11507768.

[23] K. B. Siddiqa *et al.*, "Assessment of survivability and importance analysis for networks managing intricate traffic flows," *IEEE Commun. Stand. Mag.*, 2025, doi: https://doi.org/10.1109/MCOMSTD.2025.3638981.

[24] M. R. H. Mahin *et al.*, "Secured and standardized intelligent zero-touch 6G framework for edge-AI applications," *IEEE Commun. Stand. Mag.*, 2026, doi: https://doi.org/10.1109/MCOMSTD.2026.3660159.

[25] S. R. Varanasi, S. S. S. Valiveti, M. Adnan, M. I. Faruk, M. J. Hossain, and M. M. T. G. Manik, "Cross-Domain standardization and secure edge intelligence for Real-Time digital twin deployments in Next-Generation communication systems," *IEEE Commun. Stand. Mag.*, 2026, doi: https://doi.org/10.1109/MCOMSTD.2026.3662187.

[26] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership inference attacks against machine learning models," in *2017 IEEE symposium on security and privacy (SP)*, IEEE, 2017, pp. 3–18.

[27] A. Shamir, "gHow to share a secret,• h Commun," 1979, *ACM*.

[28] C. J. C. H. Watkins and P. Dayan, "Q-learning," *Mach. Learn.*, vol. 8, no. 3, pp. 279–292, 1992.