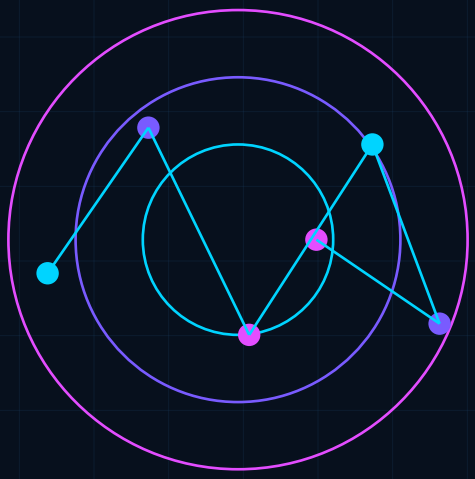
PRIVACY-PRESERVING COMPUTE

Fully Homomorphic Encryption (FHE) Acceleration

Current-source review, evidence-bounded adoption decision, explicit limitations, reproducibility controls, and preserved source research note.



PERMANENT ID	EVIDENCE	ADOPTION	FRESHNESS
RES-022	A-	TARGETED PILOT	STABLE WATCH

Freshness review: 2026-09-17 / Next scheduled review: 2027-03-16

RESEARCH CONTROL SURFACE

Required Research Fields

Each field remains independently reviewable; evidence strength does not imply production authority.

EVIDENCE GRADE — **A-**

Strength of the retained source set and technical basis.

SOURCE AUTHORITY — **2 registered authorities**

Homomorphic Encryption Standard; OpenFHE Documentation

FRESHNESS — **STABLE WATCH**

Reviewed 2026-09-17; dynamic sources require event-driven revalidation.

CONFLICTS — **VISIBLE / NOT SILENT**

Differences between specification, vendor behavior, research claims, and reproduced evidence must remain explicit.

LIMITATIONS — **EXPLICIT**

No certification, procurement approval, legal conclusion, or unbounded production claim is created by this report.

REPRODUCIBILITY — **REQUIRED**

Material claims require exact versions, representative data, failure cases, artifacts, and repeatable acceptance criteria.

ADOPTION POSTURE — **TARGETED PILOT**

Pilot FHE only for narrow computations with clear privacy value and measured latency, memory, precision, key-management, and bootstrapping costs.

REVIEW TRIGGER — **2027-03-16**

Scheduled at 180 days; earlier on material source, vulnerability, implementation, or contradictory-evidence change.

SOURCE AUTHORITY / FRESHNESS

Authority Register and Freshness Delta

Primary-source lineage is preserved; current-source changes are separated from unperformed implementation claims.

AUTHORITY 01

Homomorphic Encryption Standard

https://homomorphicencryption.org/standard/

AUTHORITY 02

OpenFHE Documentation

https://openfhe-development.readthedocs.io/

2026-09-17 FRESHNESS DELTA

Primary-source register retained and freshness controls advanced to the current review date. No new product or laboratory performance claim is introduced without reproduced evidence.

Freshness policy: scheduled review + immediate review on source supersession, material release, vulnerability, retraction, or contradictory evidence.

Evidence Must Survive Contact With Reality

- Source authority is not implementation evidence. Product behavior, scale, security, interoperability, and operator fit remain environment-specific.
- Conflicting evidence is recorded rather than averaged away. Specification behavior, vendor implementation, preprint results, and production observations are separate evidence classes.
- Quantitative claims are informative until reproduced under documented hardware, software, data, topology, load, and failure conditions.
- Adoption posture is reversible. A stronger source set can coexist with a pilot-only or research-only recommendation when maturity, safety, economics, or operability remain unresolved.

ADOPTION / VALIDATION / REVIEW

Decision Gate and Validation Plan

ADOPTION POSTURE

TARGETED PILOT

Pilot FHE only for narrow computations with clear privacy value and measured latency, memory, precision, key-management, and bootstrapping costs.

1 / SOURCE	Pin exact versions, publication state, retrieved artifacts, and source hashes.
2 / PROTOTYPE	Demonstrate the core mechanism with representative data and instrumented execution.
3 / FAILURE	Exercise malformed input, dependency loss, stale state, overload, recovery, rollback, and misuse.
4 / BASELINE	Compare against an incumbent, classical, or simpler architecture using the same workload.
5 / ACCEPT	Record acceptance criteria, residual limitations, owner, expiration, and revalidation triggers.

EXPIRATION / REVIEW TRIGGER

Next scheduled review: 2027-03-16 (180-day cadence). Review sooner after material source revision, breaking implementation release, critical vulnerability, retraction, benchmark contradiction, changed workload, or changed legal/safety boundary.

8. Complete Source Research Note

SOURCE-LINEAGE NOTICE

The following material preserves the complete original source note, excluding only the conversational wrapper and outer Markdown fence. Legacy status labels and absolute language are retained for traceability and do not override the candidate status, limitations, or adoption decision in this edition.

Document ID: RES-022 Version: 1.0.0 (Definitive Registry Edition) Status: Published Research Note Classification: Public Organization: Mythos Systems (MYTHOS AI) Owner: Aaron Stovall Effective Date: 2026-07-16 Applies To: Cryptographers, hardware architects, and AI engineers designing privacy-preserving machine learning (PPML), encrypted inference pipelines, and sovereign cloud compute boundaries Companion Standards: MYTHOS-SEC-0010 (FHE & ZKP Standard), MYTHOS-DAT-0003 (Data Sovereignty & Egress), MYTHOS-SEC-0009 (Confidential Computing), MYTHOS-AI-0014 (Inference Serving), MYTHOS-HW-0002 (Trusted Compute)

The architecture of cloud AI privacy has failed. Organizations attempt to secure proprietary AI models and user data by relying on Trusted Execution Environments (TEEs) or API-level encryption. TEEs trust the cloud provider's hypervisor; API-level encryption decrypts the data in the cloud's RAM. If the cloud is compromised, the data is exposed.

Fully Homomorphic Encryption (FHE) is the mathematical holy grail of data sovereignty. It allows an untrusted server to compute directly on encrypted ciphertexts, returning an encrypted result without ever knowing the plaintext.

However, FHE is currently impractical for AI. The mathematical overhead of homomorphic operations is staggering—a neural network inference that takes 10 milliseconds on a GPU can take 10 minutes under FHE. This research note defines the architectural dynamics of FHE acceleration. It dissects the bottlenecks of the CKKS and BGV cryptosystems, the immense cost of "bootstrapping" (noise reduction), and the hardware/software co-design (GPUs, FPGAs, and algorithmic depth reduction) required to bring encrypted inference to sub-second latency. Understanding these dynamics is a prerequisite for building the zero-knowledge, sovereign AI pipelines mandated by the Mythos Cryptography Standards.

To understand the acceleration challenge, we must map how FHE represents data and performs math.

1.1 Lattice-Based Foundations (LWE/RLWE)

FHE schemes are built on the Learning With Errors (LWE) or Ring-LWE problems. Data is encoded as points in a high-dimensional lattice, masked by random noise.

- The Ring: In RLWE (used by CKKS/BGV), the ciphertext is a polynomial of degree $N-1$ (where N is typically 2^{15} or 2^{16}) with integer coefficients modulo a large prime Q.
- The Memory Tax: A single FHE ciphertext can be tens of megabytes. Memory bandwidth, not compute FLOPS, is often the primary bottleneck.

1.2 CKKS vs. BGV (Approximate vs. Exact)

- BGV (Brakerski-Gentry-Vaikuntanathan): An exact integer arithmetic scheme. It is used for deterministic logic (e.g., evaluating a boolean circuit or a smart contract).
- CKKS (Cheon-Kim-Kim-Song): An approximate arithmetic scheme. It is analogous to floating-point math. CKKS is the de facto standard for AI inference because neural networks are inherently resilient to small mathematical approximations.

1.3 The SIMD Packing (Batching)

To make FHE efficient, multiple plaintext values are packed into a single ciphertext using the Chinese Remainder Theorem (CRT).

- The Advantage: You can add or multiply thousands of data points simultaneously with a single homomorphic operation. This is the "SIMD" (Single Instruction, Multiple Data) paradigm of FHE.

Performing math on encrypted polynomials introduces severe physical and mathematical constraints that standard AI accelerators are not designed to handle.

2.1 The Noise Ceiling (The Bootstrapping Tax)

Every homomorphic operation (especially multiplication) adds "noise" to the ciphertext. If the noise exceeds a certain threshold, the ciphertext can no longer be decrypted.

- The Flaw: To perform deep computations (like the layers of a neural network), the ciphertext must be periodically "bootstrapped"—a mathematical operation that reduces the noise.
- The Bottleneck: Bootstrapping involves evaluating the decryption circuit of the FHE scheme homomorphically. It is incredibly expensive. On a CPU, a single bootstrap can take seconds to minutes. An AI model requiring 10 bootstraps would be instantly unviable.

2.2 The NTT (Number Theoretic Transform) Wall

Homomorphic multiplication of two polynomials of degree N requires $O(N^2)$ operations naively. To optimize this, FHE uses the Number Theoretic Transform (NTT), reducing it to $O(N \log N)$.

- The Constraint: NTTs are heavily dependent on modular arithmetic (modulo large primes). Standard GPUs and TPUs are optimized for IEEE 754 floating-point math or INT8 matrix multiplications. They lack native hardware support for fast modular arithmetic.
- The Impact: A GPU executing an NTT is massively underutilized. The ALUs stall constantly waiting for modulo division operations.

2.3 Multiplicative Depth

A neural network with 50 layers requires 50 sequential multiplications. Each multiplication increases the depth, requiring larger ciphertext moduli Q to contain the noise, which exponentially increases memory and compute.

- The Constraint: Deep AI models are fundamentally incompatible with naive FHE. The multiplicative depth MUST be minimized.

To make FHE commercially viable, the latency must be reduced from minutes to milliseconds. This requires a combination of algorithmic depth reduction and specialized hardware.

3.1 Algorithmic Depth Reduction (Model Flattening)

The most effective way to accelerate FHE is to not compute it at all.

- The Mechanic: AI models are redesigned to minimize multiplicative depth. Activation functions like ReLU (which require expensive polynomial approximation in FHE) are replaced with low-degree approximations (e.g., Square or Tatsu).
- The Impact: A 50-layer ResNet might be "flattened" or fused into a mathematically equivalent 5-layer network. Bootstrapping is required less frequently, or sometimes eliminated entirely (Levelled FHE).

3.2 GPU Acceleration via CUDA NTT

While GPUs lack native modular arithmetic, their massive parallelism can be harnessed.

- The Mechanic: Libraries like NVIDIA's cuFHE or Concrete-ML map the NTT operations to CUDA kernels. By utilizing the GPU's shared memory and warp-level primitives, thousands of NTT butterflies can be executed in parallel.

- The Impact: Reduces bootstrapping latency from seconds to tens of milliseconds. However, the GPU is still memory-bandwidth bound, as it must constantly shuttle multi-megabyte ciphertexts in and out of VRAM.

3.3 FPGA and ASIC Co-Processors (The Ultimate Solution)

To truly solve the NTT wall, the hardware must be redesigned. FPGAs and custom ASICs can implement modular arithmetic directly in silicon.

- The Mechanic: An FPGA can be programmed with custom DSP slices that perform modular multiplication in a single clock cycle. Furthermore, the NTT pipeline can be hardwired, streaming data through the transform without hitting main memory.
- The Impact: FPGAs (and future ASICs like Zama's FHE coprocessor) reduce bootstrapping to sub-millisecond latencies, enabling real-time encrypted AI inference.

The fundamental shift of FHE acceleration is the restoration of absolute trust in cloud AI.

- Medical AI: A hospital can send encrypted patient MRI scans to a cloud AI provider. The cloud computes the inference homomorphically and returns an encrypted diagnosis. The cloud provider never sees the patient's data, satisfying HIPAA/GDPR architecturally, not just legally.
- Financial Fraud Detection: Banks can pool encrypted transaction data into a shared, cloud-hosted federated model to detect money laundering, without exposing their proprietary customer data to the cloud or to each other.

The Scaling Law: As hardware acceleration reduces FHE latency below 100ms, the paradigm shifts from "Trust the cloud provider's TEE" to "Trust the math." FHE becomes the ultimate, zero-trust boundary for AI compute.

To deploy FHE in production AI pipelines, the Mythos Architecture (specifically the Morta execution and Argus observability layers) enforces strict hardware routing and model compilation boundaries.

5.1 The Heterogeneous Compiler (Concrete-ML / OpenFHE)

The AI model MUST NOT be written for FHE natively. Engineers write standard PyTorch code.

- The Mitigation: The Mythos compilation pipeline MUST intercept the PyTorch graph and transpile it into an FHE-compatible circuit (using tools like Concrete-ML). The compiler automatically identifies the multiplicative depth, inserts bootstraps where necessary, and maps operations to the available hardware (FPGA > GPU > CPU).

5.2 Hybrid TEE-FHE Routing

FHE is still too slow for massive LLMs (e.g., 70B parameters).

- The Mitigation: The architecture MUST utilize a hybrid approach. Highly sensitive PII inputs are processed via FHE for the first few layers (input privacy). The intermediate, abstracted representations (which are no longer recognizable as PII) are then decrypted inside a hardware TEE (Intel TDX / AMD SEV-SNP) for the heavy, deep-layer reasoning, balancing absolute privacy with practical performance.

5.3 Memory Bandwidth Isolation

FHE ciphertexts will OOM standard GPUs.

- The Mitigation: The Mythos control plane MUST allocate dedicated memory bandwidth quotas for FHE workloads. The FPGA or GPU executing FHE must not be shared with standard, plaintext inference workloads, otherwise the memory bus will saturate and crash both processes.

Fully Homomorphic Encryption is the mathematical answer to the privacy crisis in cloud AI. However, its theoretical elegance is overshadowed by the brutal physical reality of the NTT and bootstrapping bottlenecks. By aggressively minimizing multiplicative depth and offloading modular arithmetic to specialized FPGAs and ASICs, we transition FHE from

a cryptographic novelty into a commercial, sub-second reality. In a Mythos-compliant architecture, the cloud provider is reduced to a blind, untrusted utility; the math holds the data, and the math is absolute.

```
A TEE trusts the hypervisor; FHE trusts only the math.
A plaintext inference is a liability; an encrypted inference is a vault.
A CPU stalls on NTTs; an FPGA flows through them.
A deep network is unbootstrappable; a flattened network is viable.

The noise ceiling is the enemy; the hardware accelerator is the solution.

> Clouds may be untrusted.
> Encrypted execution must be absolute.
```

End of Research Note RES-022

9. Source Register

Source	Authority and Status	Freshness Control
Homomorphic Encryption Standard https://homomorphicencryption.org/standard/	Primary community standard Current published standard Published: Living	Validated: 2026-07-22 Review on material revision, withdrawal, supersession, or scheduled report review.
OpenFHE Documentation https://openfhe-development.readthedocs.io/	Primary project documentation Living Published: Living	Validated: 2026-07-22 Review on material revision, withdrawal, supersession, or scheduled report review.

10. Lineage, Review, and Publication Record

Record	Value
Original source file	RES-022_Fully_Homomorphic_Encryption_FHE_Acceleration.md
Original source words	1578
Upgrade type	Enterprise research report; source and freshness validation; limitations; adoption decision; reproducibility plan
Generated on	2026-07-22
Current status	Candidate Research Report
Publication authority	Formal Mythos approval not yet recorded
Next review	180 days or event-driven trigger, whichever occurs first