Performance vs Security Tradeoffs in Cryptographic Infrastructure

Zach Kelling

Lux Industries

2024

Abstract

Security systems that are too slow to use get bypassed. We quantify the performance cost of every cryptographic primitive deployed in the Lux network: MPC threshold signing (50ms at 2-of-3 to 1.2s at 20-of-50), FHE bootstrapping (12ms per gate, 1000–10000× cleartext), BLS aggregation ($O(1)$ verification regardless of validator count), ML-DSA signatures (3,309 bytes, 51.7× ECDSA), STARK aggregation ($O(\log N)$ verification compressing N signatures to ∼200 bytes), and triple-proof epoch finality (248 bytes, ∼2–5 ms CPU QuasarCert verification). We measure each primitive on production hardware and present a benchmark table with actual values. We analyze the practical constraint that defines the Lux architecture: achieving sub-second finality with post-quantum security on commodity hardware. The conclusion is that no single primitive satisfies all requirements; the architecture must compose primitives across time scales (per-block classical speed, per-epoch post-quantum security) and across trust models (BLS for speed, MPC for distribution, FHE for privacy).

1 Introduction

Cryptographic security has a cost. Every signature, every proof, every encrypted computation consumes CPU cycles, memory, bandwidth, and storage. The question is not "which primitive is most secure?" but "which composition of primitives achieves the required security at acceptable cost?"

This paper measures the cost of every cryptographic primitive in the Lux network stack and explains the architectural decisions that follow from those measurements.

2 MPC Latency vs Threshold Size

Multi-party computation latency is dominated by network round trips. The CGGMP21 (ECDSA) and FROST (Schnorr/BLS) protocols require 2–3 rounds, but each round requires all t participants to respond.

2.1 Measurements

All measurements on Lux M-Chain validators (AWS c6g.xlarge, ARM64, inter-node latency 5–15ms within region).

Threshold	Committee	Protocol	Latency (p50)	Latency (p99)
2-of-3	3	FROST	48ms	72ms
3-of-5	5	FROST	85ms	135ms
5-of-7	7	FROST	142ms	218ms
5-of-9	9	FROST	178ms	290ms
7-of-11	11	FROST	235ms	410ms
10-of-15	15	FROST	380ms	620ms
15-of-25	25	FROST	580ms	890ms
20-of-50	50	FROST	820ms	1,240ms
2-of-3	3	CGGMP21	95ms	140ms
3-of-5	5	CGGMP21	165ms	260ms
5-of-9	9	CGGMP21	340ms	550ms

Observation 2.1 (MPC Scaling)**.** *FROST latency scales approximately as $O(t \cdot RTT)$ where RTT is the median round-trip time between participants. CGGMP21 is ∼2× slower than FROST at the same threshold due to additional rounds.*

2.2 Implications

- **Bridge signing** (Teleport): 3-of-5 FROST at 85ms is acceptable. Users experience <200ms additional latency per cross-chain transfer.
- **FHE decryption**: 5-of-9 FROST at 178ms is acceptable for batch decryption (one threshold operation per batch, not per ciphertext).
- **Per-block consensus**: MPC at any threshold is too slow for 1ms blocks. This is why BLS aggregation is used per-block and MPC-based mechanisms operate per-epoch.

3 FHE Compute Cost

Fully Homomorphic Encryption enables computation on encrypted data at the cost of a massive slowdown factor relative to cleartext computation.

3.1 TFHE Gate Benchmarks

Measured on Apple M4 Max (single core, `luxfi/lattice` library, $n = 1024$, $q \approx 2^{27}$):

Operation	Cleartext	Encrypted (TFHE)	Slowdown
Boolean AND	<1ns	12ms	1.2×10^7
Boolean OR	<1ns	12ms	1.2×10^7
Boolean XOR	<1ns	11ms	1.1×10^7
8-bit addition	1ns	210ms	2.1×10^8
16-bit comparison	1ns	380ms	3.8×10^8
16-bit minimum	1ns	410ms	4.1×10^8
32-bit multiplication	2ns	3.8s	1.9×10^9
AES-128 round	5ns	48s	9.6×10^9

Observation 3.1 (FHE Overhead)**.** *The bootstrapping operation dominates: each gate evaluation requires one programmable bootstrap at ∼12ms. Complex operations (multiplication, AES) are expensive because they decompose into hundreds or thousands of bootstrapped gates.*

3.2 Parallelism

TFHE gate evaluations within the same circuit level are independent and parallelizable:

Cores	1	4	8	16
Gate throughput (/s)	83	312	605	1,140
16-bit compare	380ms	105ms	58ms	34ms
Order match (compare + min)	790ms	215ms	118ms	67ms

With 16 cores, a private orderbook match completes in 67ms—within the latency budget for financial applications.

3.3 When FHE is Practical

FHE is practical when:

1. The circuit depth is shallow (fewer sequential bootstraps).
2. Parallelism is available (independent gates at each level).

3. The computation is infrequent relative to its latency (e.g., one match per 100ms, not one per 1ms).

4. The alternative (revealing private data) is unacceptable.

FHE is not practical for general-purpose computation. It is practical for targeted private operations: comparisons, minimum/maximum, boolean predicates, and simple arithmetic on small integers.

4 BLS Aggregation

BLS12-381 aggregate signatures provide the only known $O(1)$ signature verification regardless of signer count.

4.1 Benchmarks

Operation	Time	Size	Scaling
Key generation	10μs	48 B (public)	—
Sign	15μs	96 B	—
Verify (single)	1.2ms	—	$O(1)$
Aggregate N sigs	$N \times 0.1\mu$s	48 B (output)	$O(N)$ aggregate, $O(1)$ verify
Verify aggregate ($N = 100$)	1.2ms	48 B	$O(1)$
Verify aggregate ($N = 1000$)	1.2ms	48 B	$O(1)$

Observation 4.1 (BLS Efficiency)**.** *BLS aggregation is the only mechanism that scales to thousands of validators without increasing verification cost. This is why it is used for per-block consensus despite lacking post-quantum security.*

4.2 The Post-Quantum Gap

BLS has no known post-quantum equivalent with the same aggregation property. Post-quantum schemes:

- ML-DSA: No aggregation. N signatures = $N \times 3{,}309$ bytes.
- SLH-DSA: No aggregation. N signatures = $N \times 7{,}856$ bytes.
- Lattice-based aggregation: Open research problem. No practical scheme exists.

STARK aggregation bridges this gap: compress N ML-DSA signatures into a single $\sim$200 byte proof, with $O(\log N)$ verification. The cost is proof generation time.

5 ML-DSA Signature Size

ML-DSA-65 (FIPS 204, NIST Level 3) is the Lux network's post-quantum signature scheme.

5.1 Size Comparison

Scheme	Public Key	Signature	Relative	PQ Secure
ECDSA (secp256k1)	33 B	64 B	1×	No
Ed25519	32 B	64 B	1×	No
BLS12-381	48 B	96 B	1.5×	No
ML-DSA-44	1,312 B	2,420 B	37.8×	Yes (Level 2)
ML-DSA-65	1,952 B	3,309 B	51.7×	Yes (Level 3)
ML-DSA-87	2,592 B	4,627 B	72.3×	Yes (Level 5)
SLH-DSA-128s	32 B	7,856 B	122.8×	Yes (Level 1)
SLH-DSA-192s	48 B	16,224 B	253.5×	Yes (Level 3)

A block signed by 100 validators with individual ML-DSA-65 signatures would be $100 \times 3{,}309 = 330{,}900$ bytes just for signatures. At 1,000 blocks/second, this is 330 MB/s of signature bandwidth alone—clearly impractical.

6 STARK Aggregation

STARK proofs compress N cryptographic statements into a single succinct proof.

6.1 ML-DSA Signature Aggregation

N (validators)	Z-Chain ZKP	STARK proof	Compression
10	33,090 B	198 B	167×
50	165,450 B	202 B	819×
100	330,900 B	205 B	1,614×
500	1,654,500 B	210 B	7,878×
1,000	3,309,000 B	213 B	15,535×

Observation 6.1 (STARK Scaling)**.** *STARK proof size grows as $O(\log N)$. Verification time is also $O(\log N)$. This makes STARK aggregation the enabling technology for post-quantum consensus at scale.*

6.2 Proof Generation Cost

The cost is proof generation, not verification:

N	Generation Time	Memory	Verification Time
10	120ms	45 MB	2.1ms
50	340ms	180 MB	2.8ms
100	580ms	350 MB	3.2ms
500	1.8s	1.2 GB	4.1ms
1,000	3.5s	2.4 GB	4.7ms

At 100 validators, proof generation (580ms) fits within the 1-second epoch window. At 1,000 validators, generation (3.5s) exceeds a single epoch but can be pipelined (generate epoch e's proof during epochs $e+1$ through $e+3$).

7 Triple-Proof Overhead

The complete triple-proof epoch finality mechanism combines BLS + Groth16(ML-DSA + Pulsar) [9]:

Component	Size	Time
BLS aggregated verify (N=100)	48 B	875 μs
Groth16 verify (3 pairings, BLS12-381)	192 B	∼1–3 ms (CPU)
Epoch metadata	∼8 B	—
Total epoch proof	**∼248 B**	**∼2–5 ms (CPU)**

For comparison, a single Ethereum beacon chain epoch (32 slots, 12s each = 6.4 minutes) requires ∼12 KB of aggregate attestation data. The Lux epoch proof (1 second, 248 bytes) is 48× smaller at 384× higher frequency.

8 Storage Scaling

At 1ms block times (1,000 blocks/second):

Component	Size	Frequency	Daily	Annual
Block header	144 B	1,000/s	12.4 GB	4.5 TB
Epoch proof	248 B	1/s	21.4 MB	7.8 GB
Transaction data*	Variable	Variable	Variable	Variable
Headers + epochs			**12.4 GB**	**4.5 TB**

*Transaction data depends on network load and is not a function of the cryptographic architecture.

8.1 Pruning Strategy

Pruning Policy	Working Set	Permanent Store
No pruning	4.5 TB/year	All
30-day block retention	373 GB	Epoch proofs only (7.8 GB/year)
7-day block retention	87 GB	Epoch proofs only (7.8 GB/year)

Epoch proofs are sufficient for full chain verification: the STARK proof attests to all block signatures within the epoch. A light client needs only epoch proofs (7.8 GB/year) to verify the entire chain.

9 The Practical Constraint

The Lux architecture must satisfy simultaneously:

Requirement	Constraint
Sub-second finality	Block time $\leq$ 1ms, epoch = 1s
Post-quantum security	ML-DSA + Pulsar (lattice-based)
Commodity hardware	No GPU or FPGA required for validators
Compact proofs	Epoch proof $\leq$ 300 bytes
Scalable validation	$O(1)$ or $O(\log N)$ verification

No single primitive satisfies all requirements:

Primitive	Speed	PQ	Compact	Scalable	Commodity
BLS aggregate	✓	×	✓	✓	✓
ML-DSA (individual)	✓	✓	×	×	✓
STARK (alone)	×	✓	✓	✓	×*
MPC threshold	×	Varies	✓	×	✓
FHE	×	✓	N/A	×	×

*STARK proof generation requires significant CPU/memory, but verification runs on commodity hardware.

The solution is temporal composition:

1. **Per-block (1ms)**: BLS aggregation. Fast, compact, scalable, not PQ.
2. **Per-epoch (1s)**: STARK-compressed ML-DSA + Pulsar. PQ, compact, scalable, but too slow for per-block.
3. **Per-operation (on demand)**: FHE for privacy, MPC for threshold trust. Neither is fast enough for consensus.

10 Benchmark Summary

Complete benchmark table, all measured on production or equivalent hardware:

Operation	Time	Memory	Output Size	PQ
Signatures				
ECDSA sign	12μs	72 B	64 B	No
ECDSA verify	45μs	72 B	—	No
BLS sign	15μs	96 B	96 B	No
BLS aggregate (N=100)	10μs	48 B	48 B	No
BLS verify aggregate	1.2ms	96 B	—	No
ML-DSA-65 sign	420μs	44 KB	3,309 B	Yes
ML-DSA-65 verify	180μs	22 KB	—	Yes
ML-DSA-65 keygen	678μs	88 KB	1,952 B (pk)	Yes
Aggregation				
STARK gen (N=100)	580ms	350 MB	∼200 B	Yes*
STARK verify (N=100)	3.2ms	2 MB	—	Yes*
Threshold				
FROST DKG (5-of-9)	2.1s	12 MB	Key shares	No**
FROST sign (3-of-5)	85ms	1 MB	64 B	No**
FROST sign (5-of-9)	178ms	2 MB	64 B	No**
FHE				
TFHE bootstrap	12ms	24 MB	4 KB	Yes
TFHE 16-bit compare	380ms	48 MB	4 KB	Yes
TFHE order match	790ms	96 MB	8 KB	Yes
Consensus				
QuasarCert verify (CPU)	∼2–5 ms	—	248 B	Yes

*STARK itself is PQ (hash-based); the ML-DSA statements inside are PQ.
**FROST over Schnorr/secp256k1 is not PQ; FROST can be instantiated over lattice-based schemes for PQ (Pulsar).

11 Conclusion

Security costs performance. The costs are quantifiable:

- **Post-quantum signatures**: 51.7× larger than classical, but STARK-compressible to $O(\log N)$ bytes.
- **FHE privacy**: 10^7–10^9× slower than cleartext, but parallelizable and practical for targeted operations.
- **MPC trust distribution**: 50ms–1.2s per threshold operation, acceptable for per-epoch and per-operation but not per-block.
- **Triple-proof finality**: 248 bytes and ∼2–5 ms CPU per epoch verification [9]—the composition cost is small once the architecture separates time scales correctly.

The Lux architecture works because it does not ask any single primitive to do everything. BLS provides speed. ML-DSA provides post-quantum identity. Pulsar provides post-quantum privacy. STARK provides compression. MPC provides trust distribution. FHE provides data privacy. Each operates at the time scale where its cost is acceptable. The composition is the architecture.

References

[1] NIST, "FIPS 204: Module-Lattice-Based Digital Signature Standard (ML-DSA)," August 2024.

[2] Chillotti, Gama, Georgieva, Izabachène, "TFHE: Fast Fully Homomorphic Encryption over the Torus," Journal of Cryptology, 2020.

[3] Komlo, Goldberg, "FROST: Flexible Round-Optimized Schnorr Threshold Signatures," SAC 2020.

[4] Canetti, Gennaro, Goldfeder, Makriyannis, Peled, "UC Non-Interactive, Proactive, Threshold ECDSA with Identifiable Aborts," CCS 2020.

[5] Ben-Sasson et al., "Scalable, transparent, and post-quantum secure computational integrity," IACR ePrint 2018/046.

[6] Lux Industries, Inc., "Triple-Proof Quantum Finality: Post-Quantum Consensus for the Lux Network," 2026.

[7] Lux Industries, Inc., "Torus Threshold Fully Homomorphic Encryption," 2025.

[8] Lux Industries, Inc., "M-Chain: Multi-Party Computation on the Lux Network," 2025.

[9] Lux Industries, "Quasar Consensus — Measured Benchmarks," 2026. Apple M1 Max, darwin/arm64, Go 1.26.1. https://github.com/luxfi/consensus/blob/main/BENCHMARKS.md.