AEGIS AZAZEL: Adaptive Streaming Oracle for Post-Quantum Cryptographic Obfuscation on PG(11,4)

Rafael Amichis Luengo

Proyecto Estrella / Error Code Lab — tretoef@gmail.com

Abstract. We present AEGIS AZAZEL, a post-quantum cryptographic streaming oracle that wraps a static code-based obfuscation system (GORGON) with an adaptive defense layer operating over the projective geometry PG(11,4). The system exploits the 287-bit symmetry group GL(12,GF(4)) to implement seven independent defense mechanisms that mutate the oracle's response surface based on attacker query behavior. Unlike traditional defenses that prevent information leakage, AZAZEL deliberately leaks information that is mathematically consistent but computationally poisoned. The system achieves 100% authorized-user fidelity, 74.6% contradiction injection rate, and a runtime of 2.3 seconds in pure Python with zero external dependencies. Three independent AI auditors unanimously approved the system for production deployment.

Keywords: post-quantum cryptography, projective geometry, streaming oracle, adaptive defense, code-based cryptography, PG(11,4), Desarguesian spread, human-AI collaboration

1. Introduction

The transition to post-quantum cryptography has focused primarily on lattice-based, hash-based, and code-based schemes that resist known quantum algorithms. The dominant paradigm remains defensive: cryptographic systems attempt to hide information behind computational hardness assumptions.

AEGIS AZAZEL introduces an alternative paradigm: **adaptive cryptographic gaslight**. Rather than preventing the attacker from obtaining information, the system ensures that information obtained is subtly corrupted in ways that are: (1) mathematically consistent, (2) globally contradictory, (3) behaviorally adaptive, and (4) psychologically devastating.

AZAZEL is Beast 4 in the AEGIS Crystal Labyrinth series, building on GORGON v16 (Beast 3) — a static obfuscation system with 7 neurotoxic defense layers and gap 0.0008 — and KRAKEN (Beast 2) which established statistical indistinguishability on PG(11,4).

2. Mathematical Foundation

2.1 The Projective Space PG(11,4)

The ambient space is PG(11,4) containing $N = (4^{12} - 1)/3 = 5{,}592{,}405$ projective points. The full collineation group GL(12, GF(4)) provides a security parameter of approximately 287 bits. We construct a Desarguesian spread of PG(5, GF(16)) over GF(4), yielding 1,118,481 spread lines of 5 points each.

2.2 The Oracle Model

The oracle O maintains a dynamic transformation T in GL(12, GF(4)) that evolves through elementary row operations. For authorized queries (Friend path), the oracle returns the exact column $H_p[j]$. For unauthorized queries (Enemy path), the return value is T(state) * $H_p[j]$ + contamination + rain, where state evolves cryptographically via SHA-256.

3. The Seven Hells

Hell	Codename	Mechanism

1st	Rotten Planks	GL(12,GF(4)) projective mutations via row operations
2nd	Rola Bola	Phase desynchronization (10/10 unique syndromes)
3rd	Saw Traps	Contamination + Syzygy Baiting (Moebius chains as algebraic relations)
4th	Gorgon Swamp	7 neurotoxic venoms from GORGON v16 (AZAZEL Shuffle)
5th	Fractal Wind	Subspace-trajectory entropy (ungameable)
6th	Rain	Convergence-coupled chaotic friction
7th	Judas Echo	Moebius contradiction chains with Cascade Echo

The Judas Echo with Cascade Echo (new in v5) propagates contradictions to columns at offsets j+/-1 and j+/-3, creating self-amplifying poison waves. The Resonance Judas targets the attacker's leading rank pivots, ensuring Groebner basis collision in critical dimensions.

4. The False Mirror of Surrender

When desperation patterns are detected, the oracle activates a graduated degradation schedule that gives the attacker sufficient confidence to publish preliminary results. At the terminal query, the oracle deploys a Frobenius strike, mass Judas injection, and a Synthetic Valid Key — a column 85% identical to true data that passes partial validation but fails full solving.

5. Performance Optimization

Component	v4 (5.6s)	v5 (2.3s)	Improvement
GORGON heritage	2.1s	1.0s	-52%
Oracle + attacks	3.5s	1.3s	-63%
Total	5.6s	2.3s	-59%

Key optimizations: incremental rank tracking O(12 x rank), lazy T via direct row operations, XorShift128+ PRNG for non-cryptographic paths, and attack battery fusion reducing total queries by 40%.

6. Experimental Results

Test	Result	Interpretation
Friend verification	500/500	100% fidelity
Convergence	498m + 495M	Aggressive adaptive defense
Syndromes	10/10 unique	Complete phase desync
Judas Echo	0.746 rate	74.6% contradiction injection
Judas volume	17,572	x16 vs predecessor
Replay isolation	1/200	Cross-instance uncorrelatable
Runtime	2.3-2.5s	Pure Python, zero dependencies

7. Security Analysis

The classical security parameter is $|GL(12, GF(4))| \sim 2^{287}$. No known quantum algorithm provides superpolynomial advantage: Shor is inapplicable (no hidden abelian group), Grover reduces to ~143 bits (above NIST Level 1), and quantum ISD yields $>2^{200}$ effective security.

Three independent AI auditors conducted adversarial testing: Gemini (algebraic, GO, 10/10), ChatGPT (statistical, GO, 9.9/10), Grok (stress/performance, GO, 10/10). Unanimous approval for production deployment.

8. Conclusion

AEGIS AZAZEL demonstrates that post-quantum cryptographic defense can transcend computational hardness barriers through adaptive response surfaces that weaponize the attacker's own progress. The 2.3-second runtime, 100% Friend fidelity, and 74.6% contradiction rate establish AZAZEL as a practical deployable system.

The broader contribution is methodological: genuine human-AI collaborative research — with a human architect and multiple AI auditors — can produce cryptographic systems that neither party could have created independently.

References

[1] R. Amichis Luengo, 'AEGIS GORGON v16,' Proyecto Estrella, 2026.
[2] R. Amichis Luengo, 'AEGIS KRAKEN,' Proyecto Estrella, 2025.
[3] D. J. Bernstein et al., 'Attacking and Defending the McEliece Cryptosystem,' PQCrypto 2008.
[4] J.-C. Faugere, 'A New Efficient Algorithm for Computing Groebner Bases (F4),' JPAA 1999.
[5] NIST, 'Post-Quantum Cryptography Standardization,' 2024.

Project: Proyecto Estrella / Error Code Lab — github.com/tretoef-estrella