How to Implement Cryptographic Agility

Abstracting crypto for algorithm rotation and post-quantum readiness

Step 1: Abstract Cryptographic Operations

Never call crypto primitives directly. Create a crypto service layer that wraps all operations behind a configuration-driven interface.

- Define a CryptoProvider interface with encrypt/decrypt/sign/verify/hash methods
- Store algorithm selection in configuration (not code)
- Include algorithm identifier in encrypted output (envelope pattern)
- Support multiple active algorithms during rotation periods

Step 2: Implement the Envelope Pattern

Every encrypted value should include metadata about how it was encrypted. This enables decryption even after algorithm rotation.

- Format: {algorithm_id}:{key_version}:{iv}:{ciphertext}:{tag}
- Algorithm ID maps to configuration (e.g., 'aes-256-gcm-v2')
- Key version tracks which key was used (for key rotation)

Step 3: Plan for Post-Quantum Migration

NIST has finalized post-quantum standards (ML-KEM, ML-DSA, SLH-DSA). Start preparing now:

- Inventory all cryptographic usage in your systems
- Identify highest-risk data (long-lived secrets, regulatory data)
- Test hybrid mode: classical + post-quantum algorithms simultaneously
- Monitor NIST PQC timeline and update migration plan annually