Post-Quantum

Cryptography Conference

PKI Agility and the Difference to Cryptographic Agility: Lessons from the Past and Present

Michael Osborne

CTO IBM Quantum Safe at IBM Research

KEYFACTOR CRYPTO4A SSL.com ENTRUST HID

October 28 - 30, 2025 - Kuala Lumpur, Malaysia

PKI Consortium

Cryptographic Agility / PKI Agility

Michael Osborne
IBM Research Europe - Zurich

PKI Consortium / PQC Conference
Kuala Lumpur, Malaysia

October 30, 2025

Presentation Goal

To understand why managing distributed trust is harder than changing algorithms in products

Agility

Let’s look at desired outcome rather than yet another definition

- Prepare organizations to adapt to cryptographic change without disruption
- Maintain resilience, trust, and compliance through evolving standards and threats
- Contribute to and not hamper cybersecurity and organizational agility
- Reduce the costs of all enterprise interactions with cryptography

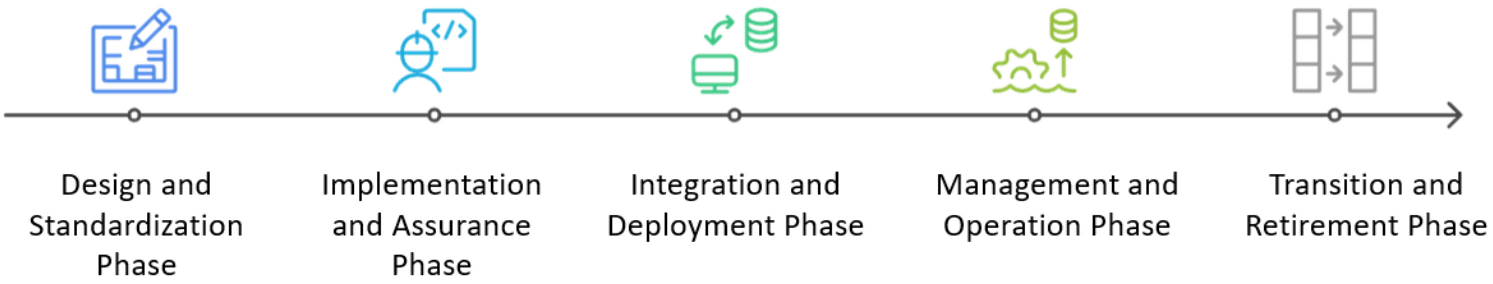
Agility

Cryptographic agility touchpoints

- Algorithmic – ability to switch algorithms
- Operational – rollout processes and deployment pipelines
- Automation, Architecture and configuration
- Governance – policies, rules, and compliance frameworks
- Ecosystem – coordination with vendors, suppliers, regulators

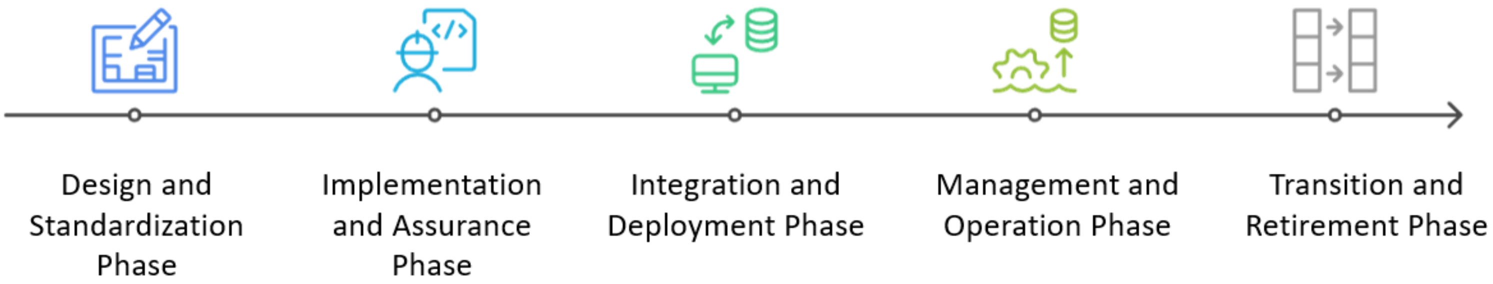
Agility

Alignment across development and deployment life cycle?



Agility

Agility goals per phase – Design and Standardization



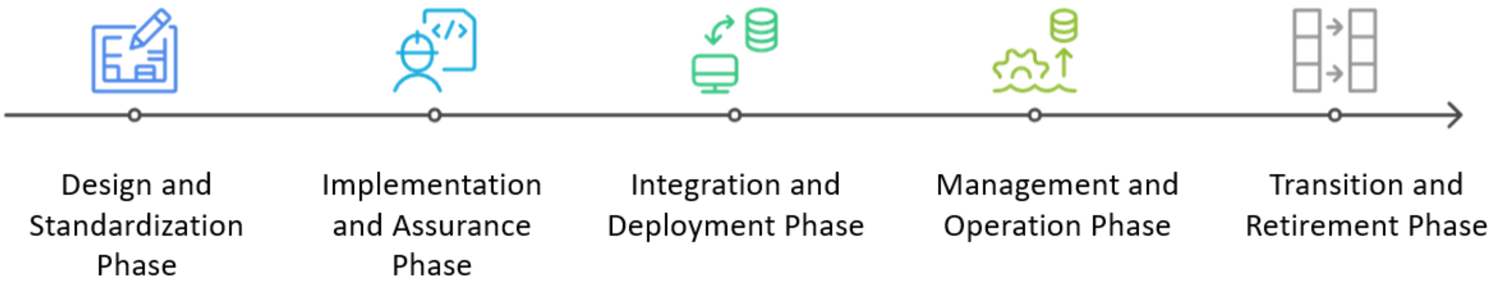
Goal: Define algorithms, protocols, and standards that can evolve without breaking ecosystems.

Algorithmic agility: specify extensible algorithm suites (e.g., hybrid modes, algorithm negotiation in protocols).

- **Standards agility:** design standards to allow future algorithm substitution (e.g., TLS cipher suite flexibility).
- **Governance agility:** coordination across standards bodies (NIST, ETSI, IETF, ISO) to prevent fragmentation.
- **Policy foresight:** include transition clauses and crypto-deprecation timelines in standards.
- **Cross-domain adaptability:** design standards that can interoperate across sectors (finance, telco, government).

Agility

Agility goals per phase – Implementation and Assurance

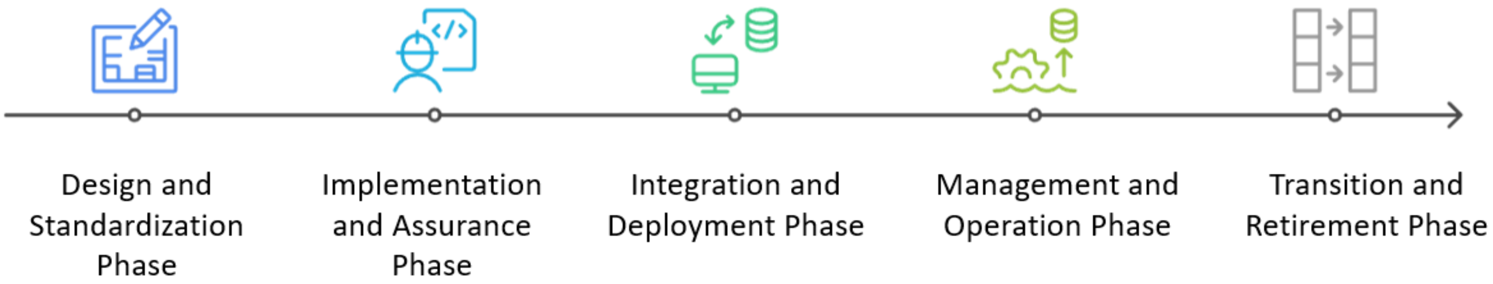


Goal: Develop cryptographic libraries, hardware modules, and assurance processes that can adapt securely.

- **Modular architecture:** isolate cryptographic modules from application logic to enable future upgrades.
- **API stability:** provide abstract cryptographic interfaces (Higher than these -> PKCS#11, OpenSSL EVP, PSA Crypto).
- **Testing agility:** automated test harnesses that support rapid validation of new algorithms.
- **Assurance agility:** certification frameworks (e.g., FIPS 140-3, Common Criteria) - automation
- **Dependency observability:** maintain CBOMs and SBOMs to track algorithm usage across software supply chains

Agility

Agility goals per phase – Integration and Deployment

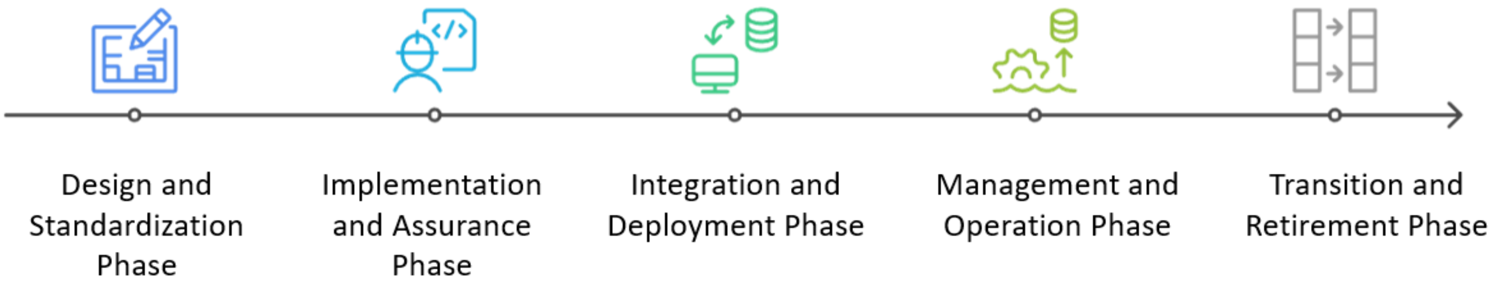


Goal: Introduce new cryptographic components into diverse system architectures without disrupting operations.

- **Configuration agility:** centralized management of crypto policies and cipher suites.
- **Interoperability:** coexistence of classical and PQC algorithms (dual stacks, hybrid signatures).
- **Dependency management:** version control and compatibility checks across heterogeneous systems.
- **Organizational coordination:** align IT operations, DevSecOps, and compliance teams for synchronized deployment.
- **Continuous integration:** incorporate crypto compliance checks in CI/CD pipelines to enforce consistent rollout.

Agility

Agility goals per phase – Management and Operation

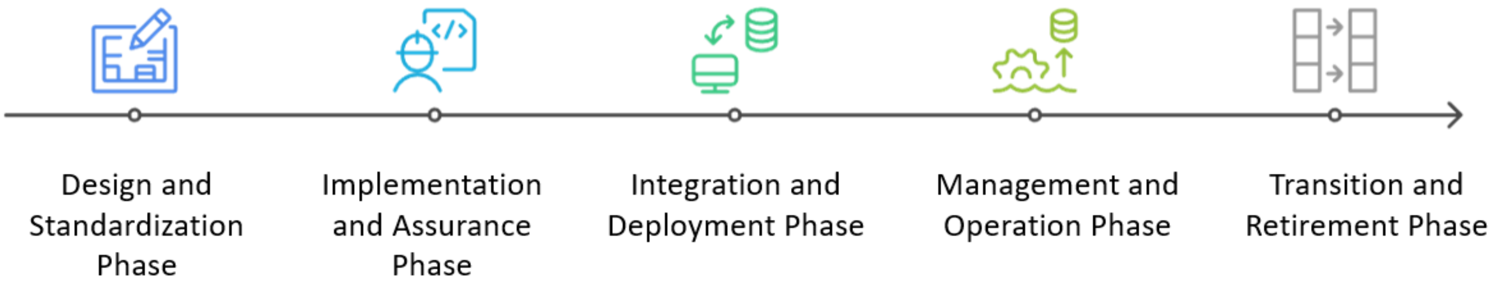


Goal: Ensure cryptographic assets are monitored, governed, and updated over time.

- **Key lifecycle control:** support key rotation, reissuance, and revocation under multiple algorithms.
- **Crypto observability:** real-time inventory of algorithms, key lengths, and certificate dependencies (via CBOMs).
- **Policy enforcement:** dynamic crypto policies that adapt to threat levels and compliance mandates.
- **Incident response agility:** the ability to reconfigure cryptographic rapidly after a vulnerability disclosure.
- **Organizational alignment:** defined ownership for crypto governance between operations, security, and compliance.

Agility

Agility goals per phase – Transition and Retirement



Goal: Replace or deprecate algorithms, standards, and systems while maintaining operational continuity.

- **Dual-stack coexistence:** phased operation of legacy and quantum-safe systems (dual PKI roots, hybrid signatures).
- **Data migration:** re-encryption or re-signing of data and archives under new primitives.
- **Transition governance:** clear decision criteria for cutover, rollback, and legacy system handling.
- **Stakeholder communication:** coordinated deprecation notifications across suppliers and partners.
- **Lifecycle closure:** formal decommissioning of legacy algorithms with audit evidence and policy updates.

Agility

Mismatch between phases – Provider vs Consumer

High Agility

Implementation and Assurance Phase

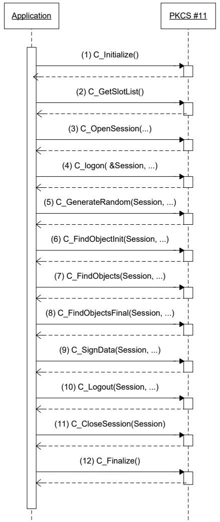
PKCS#11

Library APIs (OpenSSL)

Proprietary APIs

Platform APIs

Rich Configuration options



Integration and Deployment Phase

Low Agility

Custom provider extensions

API breakage - OpenSSL 0.x to 1.x Series, OpenSSL 2. X to 3.x Series

Sensitivity to program language support

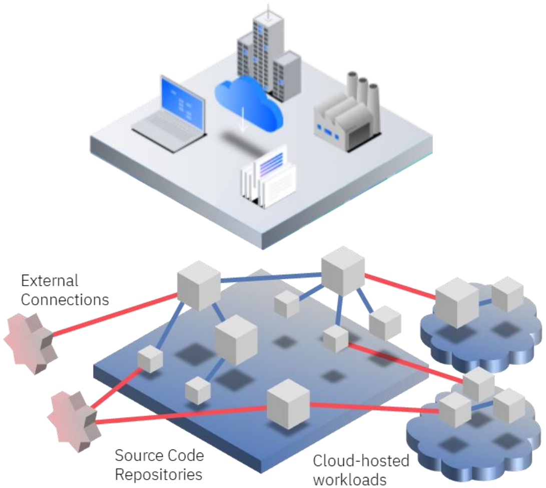
Policy encoded in the application through API usage

Low configuration management maturity

Provide secure building blocks for any scenario	→	Need to program building blocks together to do something useful
Provide a repository for cryptographic objects	→	Need to manage cryptographic objects within the application
Support multiple users	→	Need to manage logins/sessions and other state

Operational Agility:

Operational Plane –other ancillary authentication schemes



What would be the impact of somebody changing market transaction time, bid validation time, time based transaction prioritization ...?

Trusted Time

What would be the impact of somebody manipulating FX rates, Market Process, Sanction Lists, Blockchain Oracles, Credit Rating ?

Trusted Data

What would be the impact of somebody manipulating cargo routes, transaction location, geofencing ?

Trusted Location

PKI Agility: Ancillary dependencies

Trusted Time: One of many ancillary standards needing an update

Threats

- Systemic financial risk if trusted timestamp authorities (TSAs) are spoofed, undermining confidence in CBDCs, tokenized securities, and DeFi settlements.
- Cross-border disruption from large-scale time desynchronization attacks on GNSS or secure NTP services, affecting international payment finality.
- Legal uncertainty if timestamped proofs become disputable, threatening digital contracts, property registries, and regulatory reporting.
- Fragmentation risk if different jurisdictions a… interoperability across blockchains and pay…
- Future threat: Quantum-era vulnerabilities… order of transactions and financial fairness…

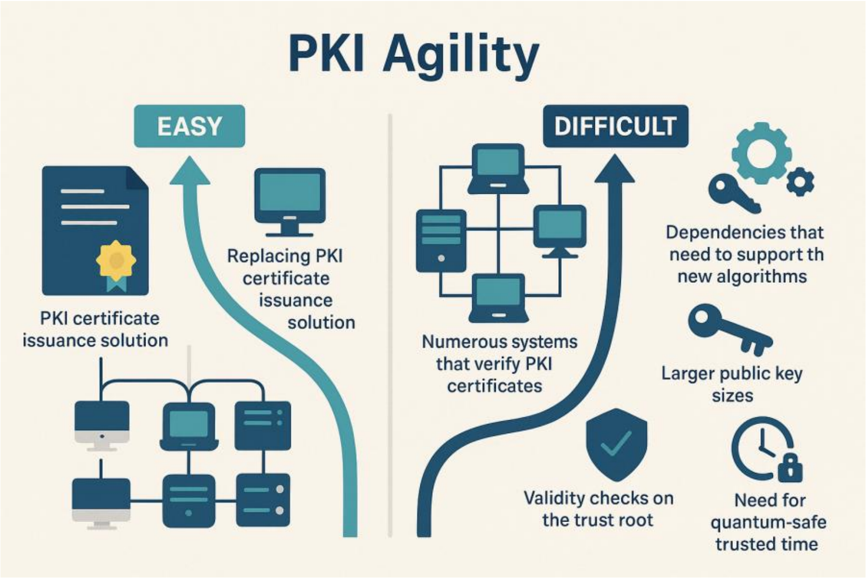
https://datatracker.ietf.org/doc/html/rfc5905
https://datatracker.ietf.org/doc/html/rfc5906
https://datatracker.ietf.org/doc/html/rfc8915
Work has begun in the IETF on NTPv5

Rfc5906 - Network Time Protocol Version 4: Autokey Specification: Identity Schemes:
(1) (2) Not recommended for production
(3) a modified Schnorr algorithm (IFF - Identify Friend or Foe),
(4) a modified Guillou-Quisquater (GQ) algorithm, and
(5) a modified Mu-Varadharajan (MV) algorithm.

Precision Time Protocol (aka PTP) which in its latest version of the standard is IEEE P1588V2.1.

PKI Agility

The Asymmetry issue?



Issuers (CAs) can migrate centrally; verifiers are distributed across many systems.

Verification dependencies are seldom captured, especially external verifiers.

Banks, governments, and telcos depend on third-party devices and software they do not control.

Fragmented update control across software vendors and user devices.

Achieving agility requires stakeholder mapping, communication plans, and rollout governance.

Hybrid composite schemes complicate agility.

Dual Root Schemes support agility

PKI Agility

Hybrid vs Dual

Composite and Hybrid Certificates

Composite designs amplify the need for **coordinated policy updates** and **testing across organizations**.

Multiple algorithm paths increase the risk of **inconsistent validation behaviors**.

Operationally harder to maintain — who is responsible for failures across hybrid chains?

Demonstrates the governance fragility of distributed PKI ecosystems.

Will require multiple transitions

Dual PKI Strategy

Parallel roots (classical + PQC) allow **phased migration** without breaking existing clients.

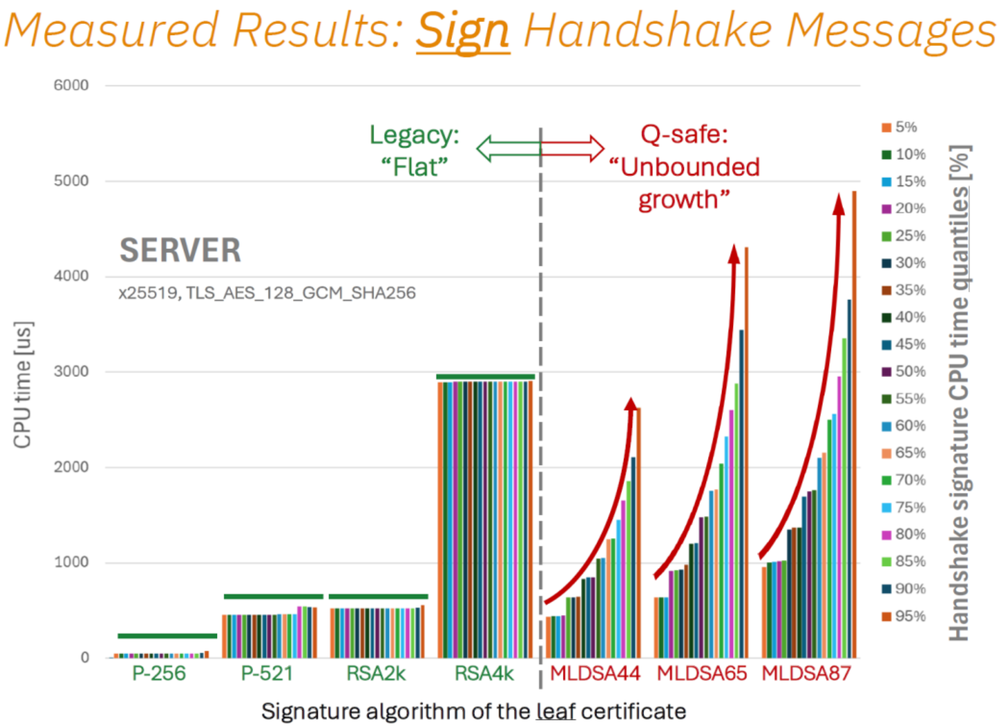
Enables clear **division of responsibilities** between legacy and PQC governance domains.

Simplifies compliance reporting and **reduces organizational coordination pressure** during rollout.

A governance-driven approach that aligns with enterprise change-management models.

PKI Agility: Case Study

ML_DSA Algorithm Impact on Operations – Rejection Sampling and Unbound Tail Latencies



Significant tail latencies for MLDSA leaf certificates

This is happening on the server-side – and the server will ALWAYS sign.

→ For a highly loaded server, ALL clients are affected by significant tail latencies

("highly loaded" = many parallel request, processed sequentially)

OpenSSL Prague 2025 *In-Situ Performance Measurements of Crypto-Algorithms in TLS v1.3* IBM, Martin Schmtz

PKI Agility: Case Study

Algorithm Impact – Server-Side Operational Perspective

Insights Summary

CIPHERS

Choice of ciphers has a small impact on overall latency (variations are within ± 50 µs)

GROUPS

Protecting against '**harvest now, decrypt later**'

- "Pure" Q-safe key agreement algorithms are virtually at par with their legacy counterparts
- MLKEM512 (128 sec bits) is even faster than legacy groups
- Hybrid X25519MLKEM768 adds ~250 µs (~=25%) to overall latency, Server load goes up ~33%

CERTIFICATES

Using Q-safe algorithms for **authentication**

- MLDSA is costly per-se AND leads to unbound tail latencies:
  - ~3x higher latency in 50% quantile,
  - BUT up to ~10x higher latency and ~20x higher server load in 99% quantile
- Having to verify a parent (e.g. intermediate) certificate virtually doubles the latencies → use 2-stage cert chains and don't send the root certificate

DATA VOLUMES

Latencies due to **data transport latencies are relatively small and hence of no big concern**: max +23 µs @ 10Gbps
The data-center egress network BW becomes the limiting factor for TLS session establishment rates
Rules of thumb:

- Protecting against **'harvest now, decrypt later'** is cutting session establishment capability by **factor of ~2**
- Adding a **2-stage Q-safe cert chain**, without sending the root certificate, cuts session rates by **~10x**
- When also sending the parent e.g. **intermediate certificate**, the cut is **~20x**

PKI Agility: Case Study

Verification dependencies

Client: Web Browser with Smart Cards, a card reader with secure PIN entry for MS Windows login and document signing, and a biometric fingerprint reader with authenticated match on the server.

Server: IBM FIPS level 4 HSMs, IBM-written application for signing land registry data mutations on a core database.

CMS based dual signature solution (RSA + ECC) to provide strength in depth.

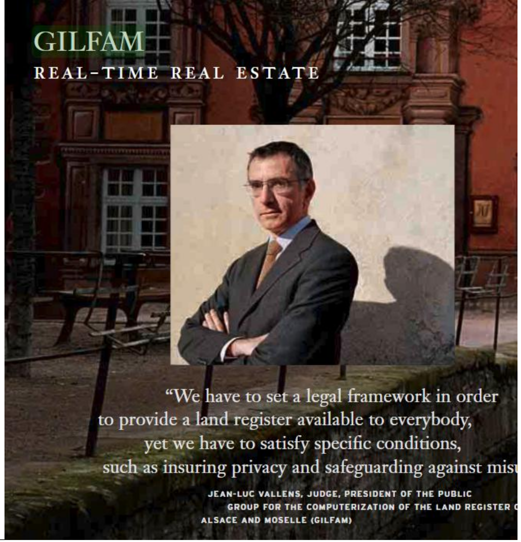
Smart Card-based authentication PKI, signing certificates for judges issued using an (m of n) admin scheme.

Challenge: One year delay to project waiting for MS to support SHA-2 on its Windows platforms

The French Ministry of Justice and the local authorities of Alsace & Moselle formed GILFAM to replace paper documents with an automated system of electronic records.

Required biometrics & Digital Signatures that are secure for 30 years

Browser based



Lessons Learned:

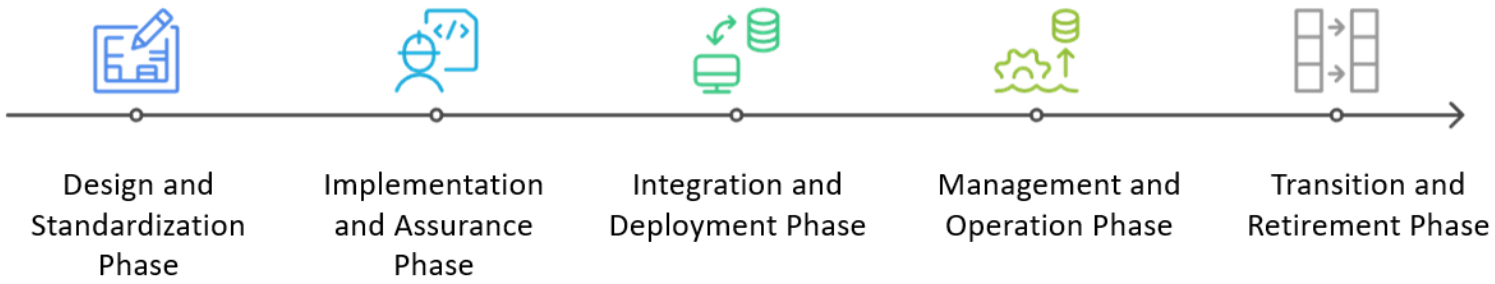
Everything that you have under your control – however complex- is goodness.

Dependencies on the verification platforms are critical.

It is not just the availability of signing algorithms, but the complete algorithm portfolio selected.

PKI Agility -

Treat PQC migration as an organizational transformation, not just a cryptography upgrade



Standardization Governance

Supply Chain Governance

Crypto Governance

Architecture and Automation

Software / Product BOMs

Deployment BOMs

Operational BOMs

Configuration Automation

Crypto Observability

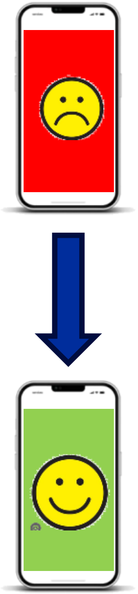
Test your device

(Direct access)

https://akamai-test-k8s-cluster.eu-de.containers.appdomain.cloud/demo

Proxy mitigation

(Via Akamai)



https://akamai.qsc-test.com/demo

Based on collaboration with Akamai. Announcement on June 18, 2025:
https://www.akamai.com/blog/trends/building-quantum-safe-internet-ietf-plan-tls

IBM **Research** Security

THANK **YOU**

Michael Osborne
CTO IBM Quantum Safe