OpenSSL CONFERENCE

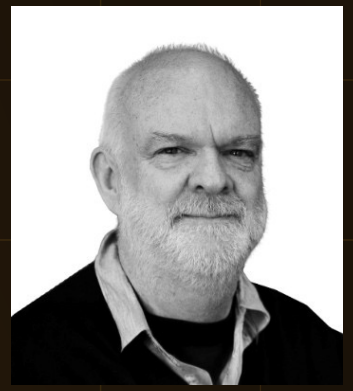
Architecting the Agility Layer: Forging a Common Crypto Agility API for the Post-Quantum Era

§ PRIVACY AS A FUNDAMENTAL HUMAN RIGHT

13.10.2026



Victor Duchovni
OpenSSL Corporation
Staff Engineer



David Hook
VP Software Engineering
Bouncy Castle Inc.



Mike Kushner
Senior Product Architect
Keyfactor



Vladimir Soukharev
VP of Cryptographic
Technology
KeyFactor



Murugiah Souppaya
Distinguished Technologist
HP Security Lab

NIST Cybersecurity White Paper
NIST CSWP 39

Considerations for Achieving Crypto Agility

Strategies and Practices

Elaine Barker*
Lily Chen
David Cooper*
Dustin Moody
Andrew Regenscheid
Murugiah Souppaya*
Computer Security Division
Information Technology Laboratory

Bill Newhouse
Applied Cybersecurity Division
Information Technology Laboratory

Russ Housley
Vigil Security

Sean Turner
sn3rd

William Barker†
Strativia LLC

Karen Kent
Trusted Cyber Annex

**Former employee; all work for this publication was done while at NIST.*

†Former employee; all work for this publication was done while at Strativia.

This publication is available free of charge from:
https://doi.org/10.6028/NIST.CSWP.39

December 19, 2025

Table of Contents

Executive Summary 1
1. Introduction 2
2. Historic Transitions and Challenges 4
2.1. Long Period for a Transition 4
2.2. Backward Compatibility and Interoperability Challenges 4
2.3. Constant Needs of Transition 5
2.4. Resource and Performance Challenges 5
3. Crypto Agility for Security Protocols 7
3.1. Algorithm Identification 7
3.1.1. Mandatory-to-Implement Algorithms 8
3.1.2. Dependent Specifications 9
3.2. Algorithm Transitions 9
3.2.1. Preserving Protocol Interoperability 10
3.2.2. Providing Notices of Expected Changes 11
3.2.3. Integrity for Algorithm Negotiation 11
3.2.4. Hybrid Cryptographic Algorithms 12
3.3. Cryptographic Key Establishment 13
3.4. Balancing Security Strength and Protocol Complexity 14
3.4.1. Balancing the Security Strength of Algorithms in a Cipher Suite 14
3.4.2. Balancing Protocol Complexity 14
4. Crypto Agility in System Implementations 16
4.1. Using an API in a Crypto Library Application 16
4.2. Using APIs in the Operating System Kernel 17
4.3. Using Service Mesh in Cloud-Native Environments 18
4.4. Embedded Systems 18
4.5. Hardware 19
4.6. Using a Crypto Gateway for Legacy Systems 20
5. Crypto Agility Strategic Plan for Managing Organizations' Crypto Risks 22
5.1. Cryptographic Standards, Regulations, and Mandates 24
5.2. Crypto Security Policy Enforcement 24
5.3. Technology Supply Chains 25
5.4. Cryptographic Architecture 25
6. Considerations for Future Works 27
6.1. Resource Considerations 27
6.2. Agility-Aware Design 27
6.3. Complexity and Security 28
6.4. Crypto Agility in the Cloud 28
6.5. Maturity Assessment for Crypto Agility 29
6.6. Common Crypto API 31
7. Conclusion 33
References 34
Appendix A. List of Symbols, Abbreviations, and Acronyms 38
Appendix B. Definition of Crypto Agility in Other Literature 41

List of Figures

Fig. 1. Using a hybrid algorithm to transition to PQC 12
Fig. 2. Functional diagram of applications using a crypto API 17
Fig. 3. Crypto agility strategic plan for managing an organization's cryptographic risks 22

Stakeholders and Use Cases

Cryptographic standard developers contribute to the common crypto API

Cryptographic library developers contribute to the common API and use it to perform functional and interoperability testing

Protocol designers contribute to and use the common API

Products and services providers contribute to, support, and use the common API across their organization

Cryptographic certification bodies and testing organizations contribute to and leverage the common API in their certification and validation programs

Cryptographic policy makers, *regulators*, and *enterprise risk managers* compose the crypto policy using the common API

Enterprise application developers use the common API to build solutions

Enterprise security operators use the common API to inventory, configure, enforce, update, log, audit, monitor, and report on cryptographic assets

Functional diagram of applications using existing crypto APIs or a common crypto agility API

OpenSSL CONFERENCE

