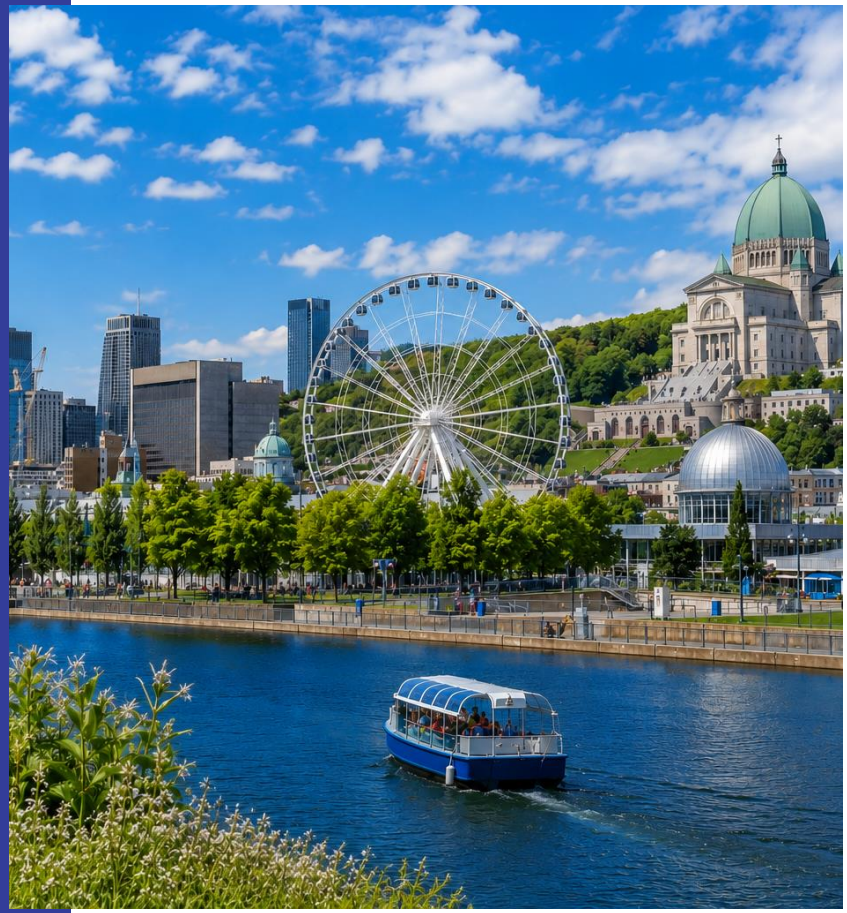
Aurora™ NETWORKS

From Integrity to Trust:
Rethinking Software Supply Chain (SSC) Security

Presenter: Xin Qiu
Authors: James Ni, Xin Qiu, Ting Yao, Lisa Yin

June 10, 2026



 © 2026 Aurora Networks. All rights reserved.

Xin Qiu, Ph.D.

Head of Security Solutions and PKI Center, Aurora Networks

Dr. Xin Qiu has over 25 years of experience in Public Key Infrastructure (PKI), device and software security, and has generated a global patent portfolio of over 100 assets.

She leads a diverse team across R&D, security operations, product marketing and management, delivering security services to global device manufacturers and network operators.

 © 2026 Aurora Networks. All rights reserved.

Agenda

- Software Supply Chain (SSC) Risk Across the Lifecycle
- Problem: Integrity vs. Trust Gap
- Limits of Artifact-Based Assurance
- Shift to Evidence-Based Trust
- Trust Signals and Correlation
- Trust in Deployment & Runtime
- Operational Impact

© 2026 Aurora Networks. All rights reserved.

Why SSC Security Matters

Growing Risk

- SSCs are a growing attack vector for Internet abuse
- Compromise enables large-scale abuse and disruption
- Trust failures propagate across ecosystems

Driving Forces

- Regulatory pressure (CRA, NIS2, etc.) and customer expectations
- Increasing complexity and attack automation

Operational Need

- Actionable trust signals, not just compliance artifacts

© 2026 Aurora Networks. All rights reserved.

Where SSC Risk Enters the SDLC

Software Development Life Cycle (SDLC)

During Development

- Malicious code injected to source and dependencies
- Tampered open-source / 3rd-party components

During Build & Test

- Code modified prior to release - build pipeline compromise
- Signing key compromise makes malware appear trusted

During Release & Deployment

- Compromised repositories / servers deliver malicious updates
- Package substitution/spoofing replace legitimate software

After Deployment - Operation

- Runtime injection enables unauthorized code execution
- Tampering of configuration update

 © 2026 Aurora Networks. All rights reserved.

What We Rely on Today

- Hash validation → detects modification
- Code signing → verifies integrity and origin
- SBOM (baseline) → lists components and dependencies
- Compliance checks → demonstrate policy adherence

Value: necessary foundation

Limit: point-in-time, not operational

© 2026 Aurora Networks. All rights reserved.

The Core Problem: Trust Gap

Why It Falls Short

- Static artifacts lack operational context
- Software and environments change after release
- New Vulnerabilities and threats emerge continuously
- Integrity ≠ trustworthiness
- Valid artifacts ≠ safe behavior

Key Question: Can this software be **trusted now**?

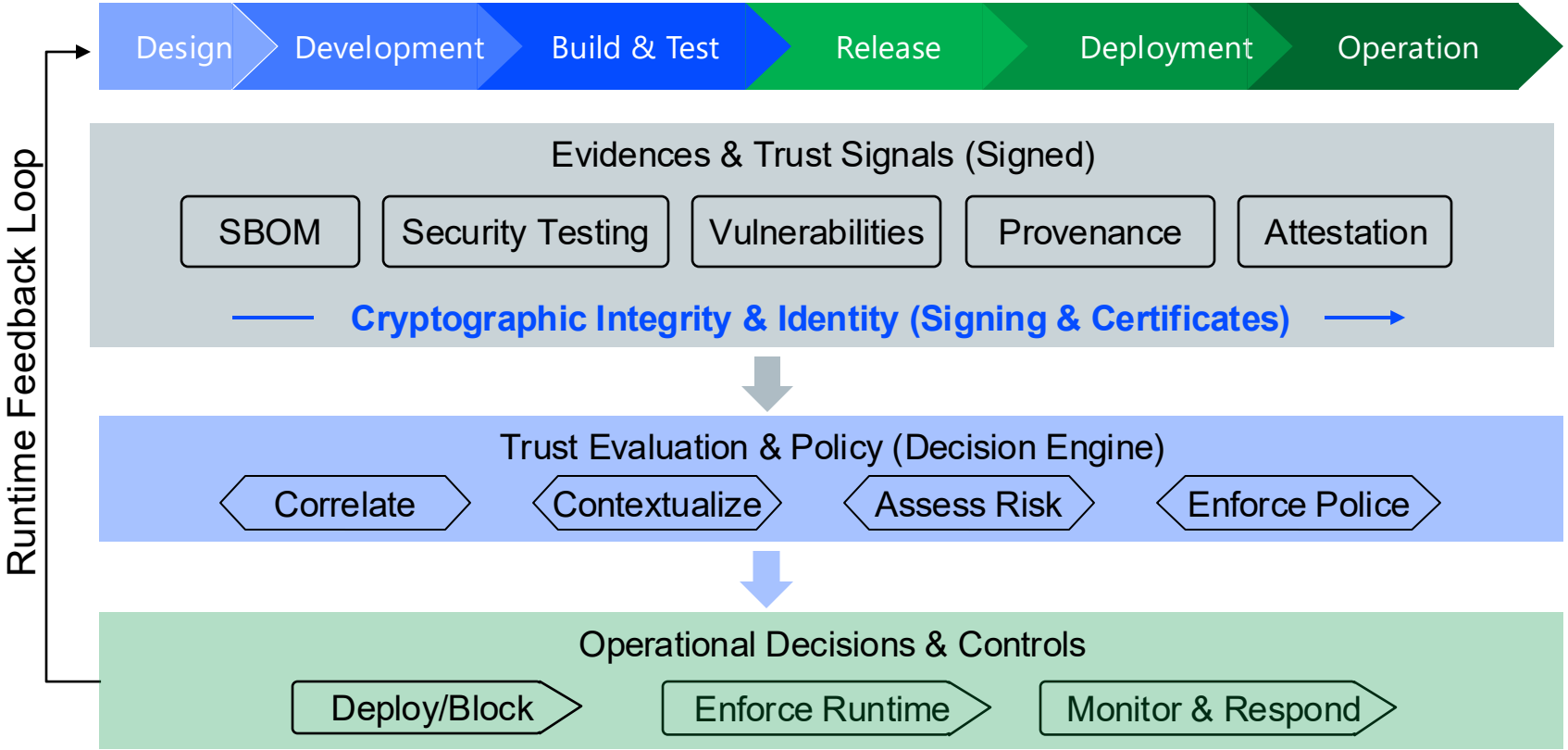
© 2026 Aurora Networks. All rights reserved.

Shift: From Artifacts to Evidence-Based Trust

- Static artifacts → Evidence
- Single signals → Correlated signals
  - SBOM • Testing • Vulnerabilities • Provenance • Attestations
- Point-in-time checks → Continuous, contextual validation
- Audit output → Operational decision

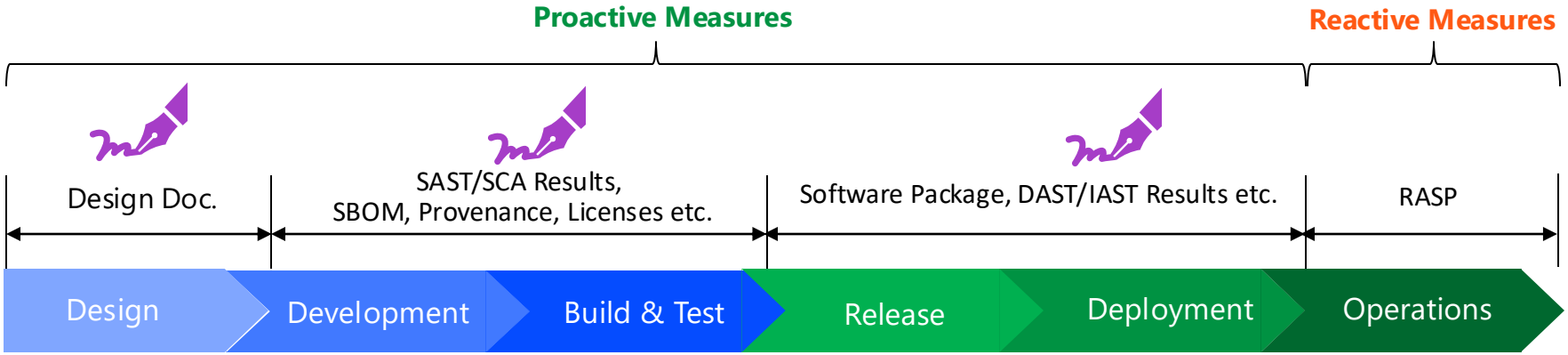
© 2026 Aurora Networks. All rights reserved.

Building Trust in SLDC



 | © 2026 Aurora Networks. All rights reserved.

Security Measures in SLDC



SAST: Static Application Security Testing

DAST: Dynamic Application Security Testing

SCA: Software Composition Analysis

IAST: Interactive Application Security Testing

RASP: Runtime Application Self-Protection

 © 2026 Aurora Networks. All rights reserved.

Key Takeaways

Compliance artifacts are necessary—but not sufficient for operations

Trust must be continuously evaluated across the lifecycle

Multiple signals must be evaluated together

Evidence—not assumptions—drives decisions

© 2026 Aurora Networks. All rights reserved.

Xin Qiu

Head of Security Solutions and PKI Center™

Aurora Networks

Email: xin.qiu@auroranetworks.com

LinkedIn: https://shorturl.at/QMDkN

pki-center.com

 © 2026 Aurora Networks. All rights reserved.