CERIAS

The Center for Education and Research in Information Assurance and Security

# Navigating Software Supply Chain Risks: Practitioner Perspectives on Software Signing

Kelechi G. Kalu, James C. Davis

## Background

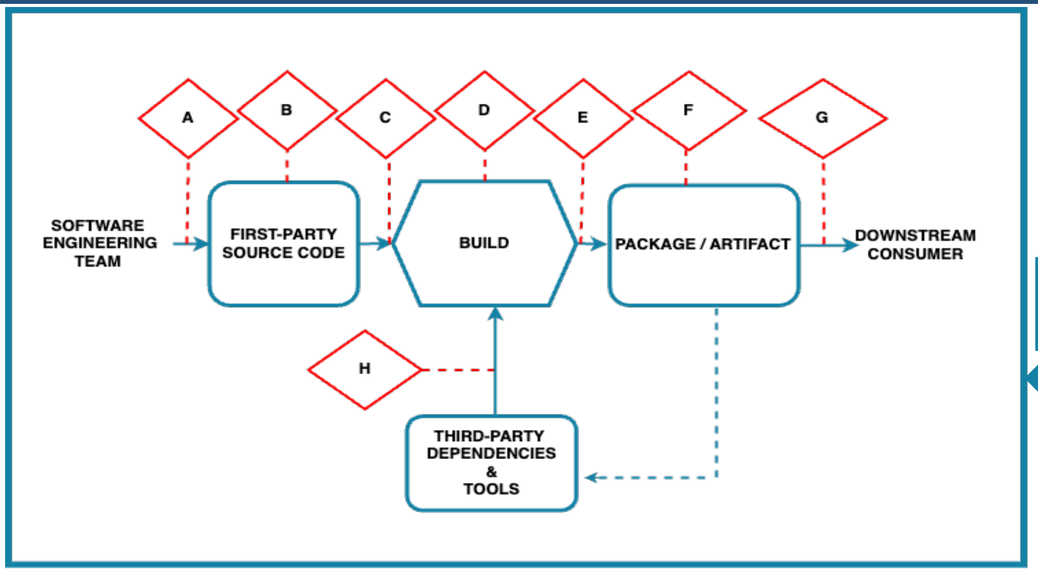


**Figure 1. Software Factory Model: The software factory model with possible attack points A-H**

**A**- Submit unauthorized change/Develop Malicious source code
**B**- Compromise source code repository
**C**- Build from Compromised/modified source
**D**- Compromise build process/Compromised Build tools(compilers, interpreters)
**E**- Upload modified/compromised package
**F**- Compromise Package Repository
**G**- Use compromised package
**H**- Use compromised dependency

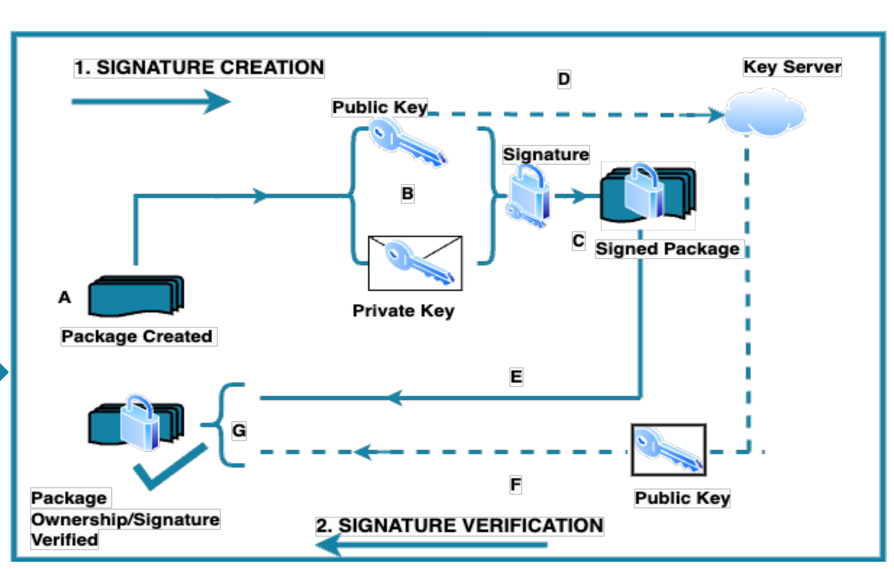


**Figure 2. Software Signing: A typical Software Signature Creation and Verification Process**

**A**- The signer/Developer/Maintainer creates a software Package and initiates a signature creation process.
**B**- A pair of Public and Private Keys are generated.
**C**- Generated Key Pairs are used to create a signature file, which in most cases are appended to the created software and pushed to a package registry or repository.
**D**- The generated public Key is published to a Key Server.
**E**- User downloads signed package from package registry or repository.
**F**- The user obtains the public key from the Key server.
**G**- User verifies the Signature/Ownership status of the signed software package with Public Key.

## Motivation

**1.** Software signing is a recommended baseline security method in Security Frameworks (SLSA, CNCF, etc)

**2.**

| Registry | PyPI | Maven Central | Docker Hub | Hugging Face |
|---|---|---|---|---|
| | 1 year (all time) | 1 year (all time) | 1 year (all time) | 1 year (all time) |
| **Total Artifacts** | **2.56M (9.65M)** | **1.72M (9.8M)** | **4.42M (7.94M)** | **261K (562K)** |
| **Unsigned Artifacts** | 99.6% (98.6%) | 4.7% (9.6%) | 98.6% (97.1%) | 99.9% (99.9%) |
| **Signed Artifacts** | 0.4% (1.4%) | 95.3% (90.4%) | 1.4% (2.9%) | 0.1% (0.1%) |
| Good Signature | 46.9% (49.8%) | 76.0% (75.2%) | 100% (100%) | 22.5% (30.7%) |
| Bad Signature | 0.1% (0.2%) | 0.2% (0.3%) | 0.0% (0.0%) | — |
| Expired Signature | 0.0% (0.1%) | 0.0% (0.0%) | 0.0% (0.0%) | — |
| Expired Public Key | 6.1% (16.4%) | 9.9% (17.2%) | 0.0% (0.0%) | — |
| Missing Public Key | 12.9% (15.7%) | 8.4% (3.7%) | 0.0% (0.0%) | — |
| Public Key Revoked | 20.1% (15.1%) | 0.9% (2.2%) | 0.0% (0.0%) | — |
| Bad Public Key | 13.9% (2.7%) | 4.6% (1.4%) | 0.0% (0.0%) | — |

**Fig 3. Study results show the Adoption rates of software signing in 4 major Software Registries Between Oct 2022 and Sep 2023, all registries aside from Maven Central had less than 2% of artifacts signed. Maven Central is the only registry that mandates signing.**

**3.**

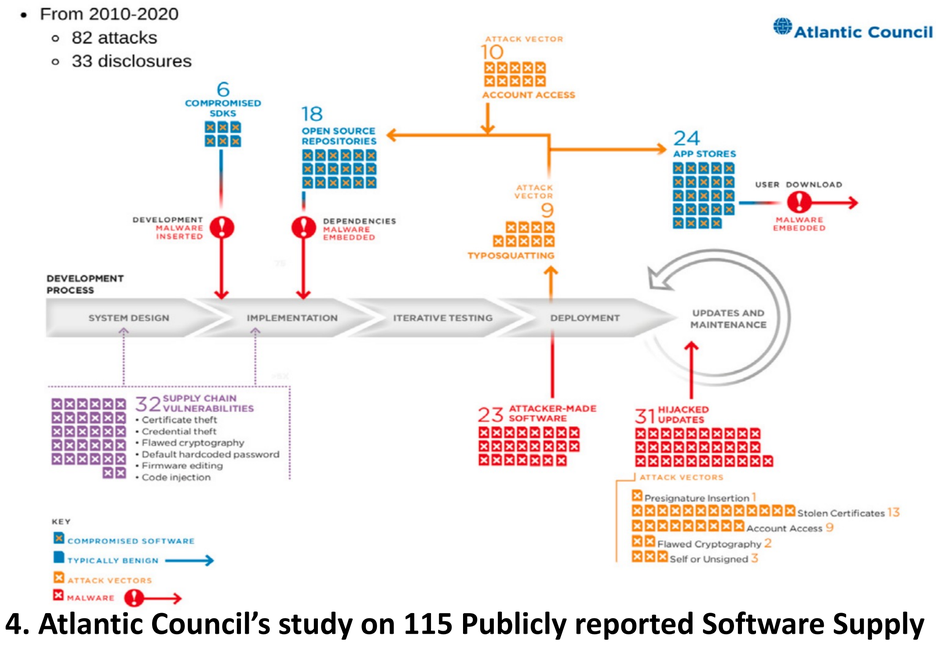


**Fig 4. Atlantic Council's study on 115 Publicly reported Software Supply chain attacks and Publicly disclosed incidents -> 31 attacks are related to Software Signing exploits - The highest attack Vector.**

## Methodology

### Research Objectives

**Risk Theme**
- **RQ1** - What are the perceived prevalent software supply chain risks faced in practice?

**Software Signing Implementation Theme**
- **RQ2** – What is the perceived importance of Software Signing in mitigating perceived risks?
- **RQ3** - How do software teams implement Software Signing as a security measure to mitigate software supply chain risks?
- **RQ4** - What factors influence the selection/adoption of specific signing tools over others in the context of mitigating software supply chain risks?

- **Qualitative Interviews**
  - **N = 18**
  - **Median Years of Experience = 13**
- **Deductive and Thematic Analysis**

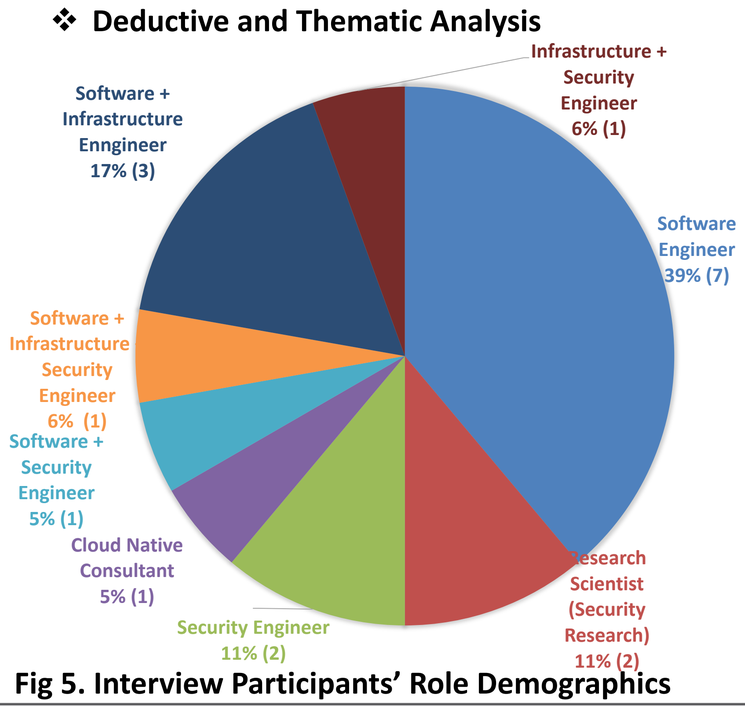


**Fig 5. Interview Participants' Role Demographics**

## Preliminary Results

A. Diverse Understanding of what is/constitute a Software Supply Chain Attack or Security.
B. Different Software Supply Chain 'risk points' identified by Experts

C. Importance of Signing to Organizational Security Processes.
D. Influence of Regulations and Frameworks on Signing and Other Practices Adopted by Teams

E. Identification of Challenges (Organizational and Technical) in Implementing Signing within Teams
F. Role of Signing in Dependency Management and Source Control

G. Strengths and Criticisms of some Software Signing Implementations.

PURDUE UNIVERSITY
NSF
CISCO
CERIAS