SCITT
Internet-Draft
Intended status: Informational
Expires: 8 January 2026

N. Aoki
SOKENDAI

7 July 2025

Supply Chain Use Cases to Design Secure Computing Systems for SCITT Extension
draft-nobuo-scitt-use-cases-extension-00

Abstract

This document includes a collection of representative Computational Supply Chain Use Cases. These use cases aim to identify computational supply chain problems that the industry faces today and act as a guideline for developing a comprehensive security architecture and solutions for these scenarios.

About This Document

This note is to be removed before publishing as an RFC.

Status information for this document may be found at https://datatracker.ietf.org/doc/draft-nobuo-scitt-use-cases-extension/.

Discussion of this document takes place on the SCITT Working Group mailing list (mailto:scitt@ietf.org), which is archived at https://mailarchive.ietf.org/arch/browse/scitt/. Subscribe at https://www.ietf.org/mailman/listinfo/scitt/.

Source for this draft and an issue tracker can be found at https://github.com/aoki-n1/draft-nobuo-scitt-use-cases-extension.

Status of This Memo

This Internet-Draft will expire on 8 January 2026.

Table of Contents

1. Introduction

Supply chain for components that make up a computer system consists of the entire lifecycle, including hardware selection, system design, development, build, integration, deployment, and maintenance. In the software supply chain, SBOM and SCITT architecture are exemplary initiatives that enhance software transparency. Discussions focusing on hardware and its interfaces are also beginning. These supply chain security measures are expected to reduce the complexity of software and provide visibility into its lifecycle, thereby reducing the number of cyber threats that can cause harmful effects such as risks related to the system's attack surface, data leaks, business disruptions, damage to reputation, intellectual property, and financial assets. On the other hand, thorough supply chain security for computer systems can only be achieved by integrating support from hardware to the software stack, enabling effective risk assessment and mitigation. Modern computer systems are influenced by evolving computer architectures and increasingly complex software stacks,

making the integrated management of components not always straightforward. End users, such as consumers, need to be able to evaluate whether suppliers maintain appropriate security practices without requiring access to proprietary intellectual property, necessitating an evolutionary extension of the SCITT specification. Post-SCITT compliant products support compliance management with legal, regulatory, and technical requirements (often differing but overlapping and interrelated), risk assessment, and detection of supply chain attacks throughout the entire lifecycle, prioritizing data privacy.

1.1. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. Generic Problem Statement

Supply chain security is a crucial requirement for ensuring the stable supply of materials that directly impact consumer survival and those widely used by the majority of consumers, while minimizing threats related to the economy, public health, and safety. As an extension of discussions in the physical domain, the definition of software supply chain security in the cyber domain, [SoK-SW-SCS], has been established. This is due to the numerous supply chain attacks targeting vulnerabilities in the software supply chain that have been experienced globally, as well as the academic progress in analyzing these attack vectors. This analysis can also be applied to the supply chains of computer systems, which include both hardware and software. Supply chain attacks on computer systems typically involve attackers gaining initial access, making malicious changes upstream in the supply chain, and exploiting vulnerabilities in the downstream systems that are already in operation.

The SCITT Architecture [I-D.draft-ietf-scitt-architecture] defines the core objects, identifiers and workflows necessary to interact with a SCITT Transparency Service:

* Signed Statements
* Receipts
* Transparent Statements
* Registration Policies

The extended YANG data model with transparency schemers [RFC9472] defines schemers for mapping SBOMs and vulnerability information.

* Access Control Lists

* SBOM Information

* Vulnerability Information

As described above, specifications for software supply chain security are maturing; however, it remains unclear whether existing standard specifications can be followed while also encompassing a scope that extends beyond software.

2.1. Computational Supply Chain Use Cases

2.1.1. Multi-Software Stack and Computer Architecture

Software integration is an essential task in building computer systems. The ecosystemization of software development is advancing, a process that involves procuring various software components from multiple suppliers at different layers and creating packages of varying sizes. These include a considerable number of third-party components. Furthermore, depending on the design, there may be cases where components are not strictly separated from one another. Additionally, modern computer systems adopt a variety of architectures and infrastructures. Similar to the increasing complexity of software stacks, computer architectures continue to evolve to keep pace with advancements in applications and hardware.

End-consumers want:

* all hardware and software components required to build a computer systems are displayed

* the ability to identify and retrieve all components from a secure and tamper-proof location - to receive an alert when a vulnerability scan detects a known security issue on a running component

* verifiable proofs on build process and build environment with all supplier tiers to ensure end-to-end build quality and security

SCITT provides a standardized way to:

* provide a tiered and transparent framework that allows for verification of integrity and authenticity of the integrated hardware and software at both component and product level before using

* notify hardware and software integrators of vulnerabilities identified during security scans of running components

* provide valid annotations on build integrity to ensure conformance

* provide an interface that reconciles the division of responsibilities between the software and hardware sides

3. Privacy Considerations

The privacy considerations of the SCITT Architecture [I-D.draft-ietf-scitt-architecture] apply.

4. Security Considerations

The privacy considerations of the SCITT Architecture [I-D.draft-ietf-scitt-architecture] apply.

5. References

5.1. Normative References

[I-D.draft-ietf-scitt-architecture]
Birkholz, H., Delignat-Lavaud, A., Fournet, C., Deshpande, Y., and S. Lasker, "An Architecture for Trustworthy and Transparent Digital Supply Chains", Work in Progress, Internet-Draft, draft-ietf-scitt-architecture-14, 4 July 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-scitt-architecture-14>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/rfc/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/rfc/rfc8174>.

5.2. Informative References

[RFC9472] Lear, E. and S. Rose, "A YANG Data Model for Reporting Software Bills of Materials (SBOMs) and Vulnerability Information", RFC 9472, DOI 10.17487/RFC9472, October 2023, <https://www.rfc-editor.org/rfc/rfc9472>.

[SoK-SW-SCS] Okafor, C., Schorlemmer, T., Torres-Arias, S., and J. Davis, "SoK: Analysis of Software Supply Chain Security by Establishing Secure Design Properties", ACM, Proceedings of the 2022 ACM Workshop on Software Supply Chain Offensive Research and Ecosystem Defenses pp. 15-24, DOI 10.1145/3560835.3564556, November 2022, <https://doi.org/10.1145/3560835.3564556>.

Author's Address

Nobuo Aoki
The Graduate University for Advanced Studies (SOKENDAI)
Japan
Email: n_aoki@ieee.org