CERIAS

The Center for Education and Research in Information Assurance and Security

# Securing the Software Package Supply Chain for Critical Systems using Permissioned Blockchains

Akash Ravi
Computer and Information Technology, Purdue University, West Lafayette

## Problem Statement

- Critical Infrastructures increasingly make use of complex software systems to support their operations.
- These systems often make use of external/public software modules or packages to help them abstract common software functionalities.
- While this has numerous benefits, it opens up an attack surface and this software supply chain can be compromised to distribute malware masqueraded as package updates or add-ons.
- With multiple interlinks and complex workflows, any “downstream" systems that rely on these software solutions are also at risk of catastrophic failure.

## Current Security Landscape

- Package distribution registries such as NPM and PyPI have implemented measures to validate package dependency chains, perform malware scanning, and audit usage history.
- Offerings such as the Snyk Intel vulnerability database, Sonatype OSS index and The Update Framework (TUF) provide ways to implement checks and balances.
- Researchers have experimented with next-gen firewalls, OS hardening, hardware checksums, and even InterPlanetary File Systems (IPFS) based methods to verify the security of software packages.

## Proposed Solution

- The proposed blockchain-based architecture enforces controls by splitting the stakeholders into 4 different discrete entities: Publishers, Package Registries, Observers, and Users/Developers
- Multiple observers scan, verify and attest to the security of packages on the permissioned ledger.
- Users will be able to read/verify off the blockchain as appropriate to enforce checks.
- Observers will have a dynamic rank and combined with a Proof of Authority (PoA) consensus algorithm, the multi-party signature (MPS) for a block commit is expected to be realizable and secure.
- The advantages include a narrow distribution of trust, a zero-trust approach, a contribution to Open Source Intelligence (OSINT), and non-intrusive compatibility with existing frameworks.

## Protection Against Malicious Entities:

- Deployment systems can interface with the ledger to verify any package being installed.
- For zero-day vulnerabilities discovered in used packages, post install hooks can be used to periodically scan deployments.
- In adverse attacks, an observer itself could be compromised and act maliciously. The proposed architecture remediates this using MPS and POA.
- All attack scenarios have been simulated and validated by integrating Corda and NPM install scripts.

**Selected References**:

1. Marjanović, J., Dalčeković, N., & Sladić, G. (2021, May). Improving critical infrastructure protection by enhancing software acquisition process through blockchain. In 7th Conference on the Engineering of Computer Based Systems (pp. 1-7).
2. Bandara, E., Shetty, S., Rahman, A., & Mukkamala, R. (2021, November). Let'sTrace—Blockchain, Federated Learning and TUF/In-ToTo Enabled Cyber Supply Chain Provenance Platform. In Military Communications Conference (MILCOM) (pp. 470-476). IEEE.

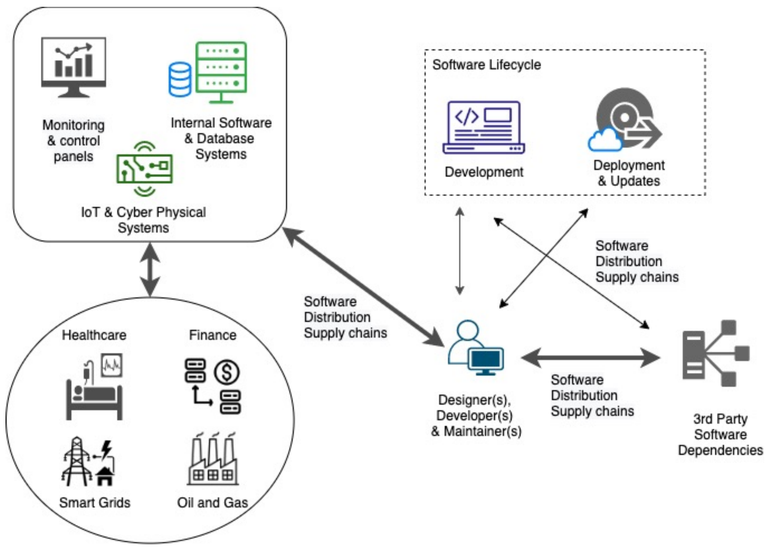


Fig.1. Involvement of Software Supply Chains in critical systems

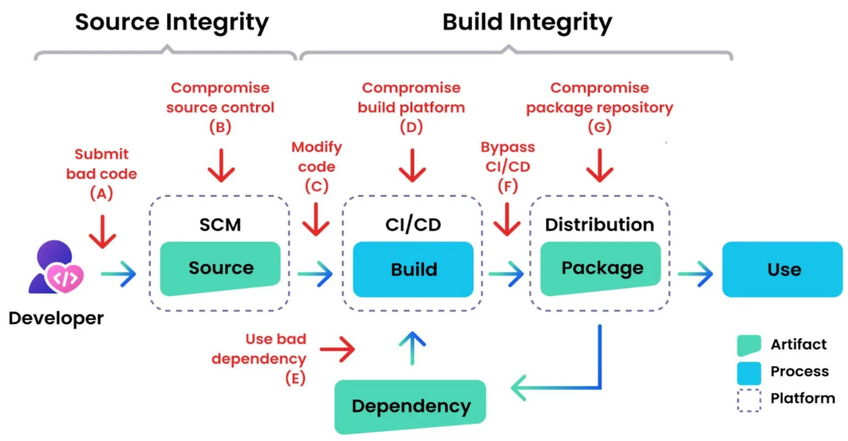


Fig. 2. Supply Chain attacks for Software Artifacts (from snyk.io)

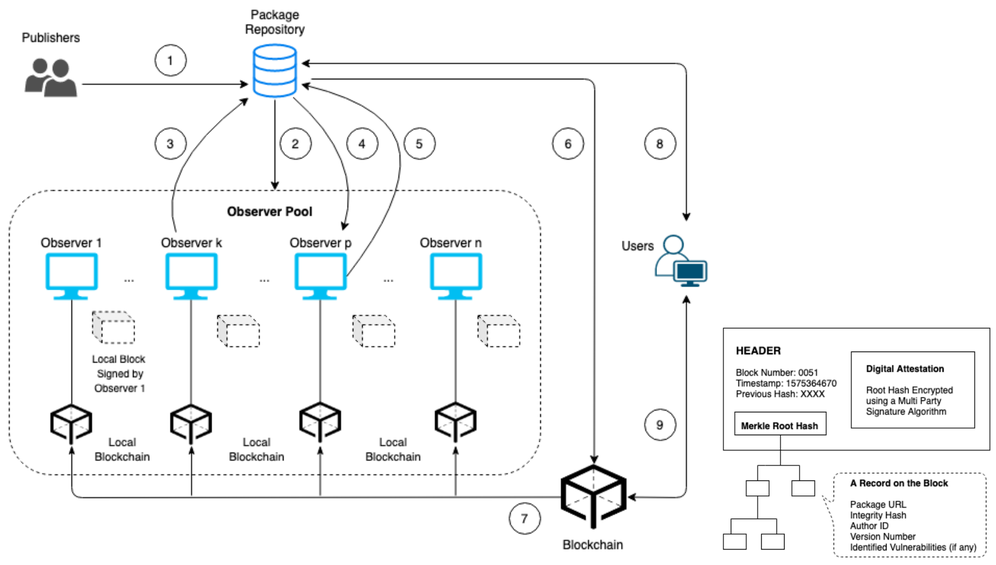


Fig.3. Interaction between entities and the block payload

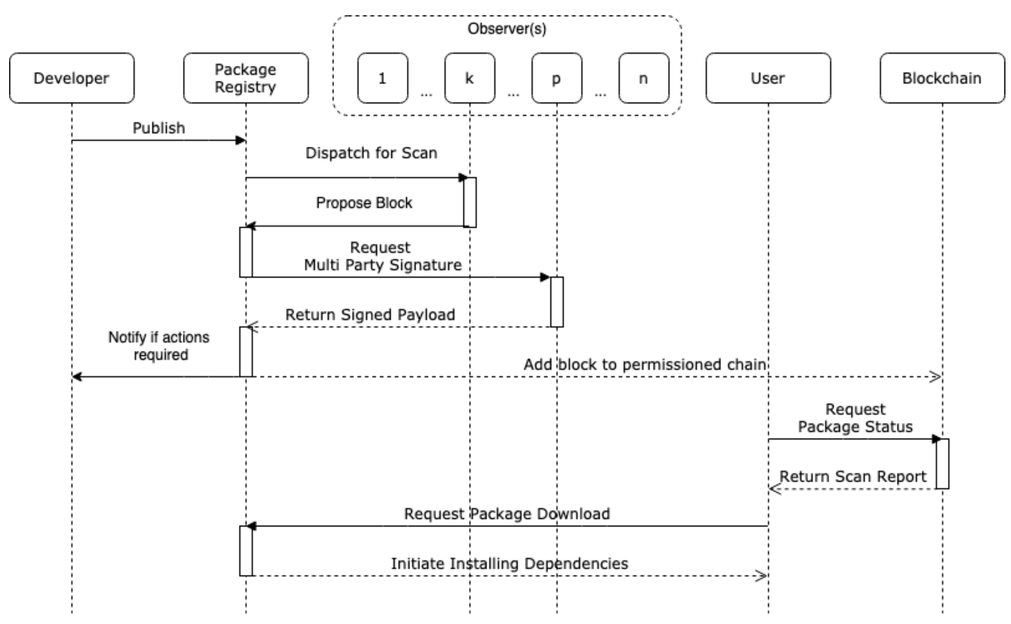


Fig.4. Sequence Diagram of the Proposed Architecture

PURDUE UNIVERSITY

CERIAS