TEE を活用した Nix バイナリキャッシュの信頼スコアリング

著者: 野田蒼馬
N 高等学校

GitHub
github.com/akazdayo/nix-cache-5-27

結論

TEE（Trusted Execution Environment）単独では、Nix バイナリキャッシュの信頼根拠として不十分である。Sybil 攻撃により同一運営者が複数の TEE インスタンスを生成でき、TEE 自体の脆弱性（SGAxe, Plundervolt 等）も確認されている。

しかし、TEE 証明を複数の独立的な証拠タイプと組み合わせることで、信頼を「信頼する／しない」の二値から定量的なスコアへ移行できる。本提案では、独立性クラスタに基づく重み付け手法により、Nix バイナリキャッシュの分散化と信頼性向上を両立する。

背景

Nix のバイナリキャッシュは、信頼済み署名鍵によって成果物の信頼性を担保する。しかし、ユーザはキャッシュされた成果物が指定された入力から実際にビルドされたことを、自ら再ビルドしなければ検証できない。この制約により、nixpkgs のような大規模な権威的リポジトリへの利用が集中する構造的課題が生じている。

バイナリ信頼（信頼／非信頼）は粗すぎる粒度であり、異なる証拠レベルを統合する枠組みが求められている。

目的

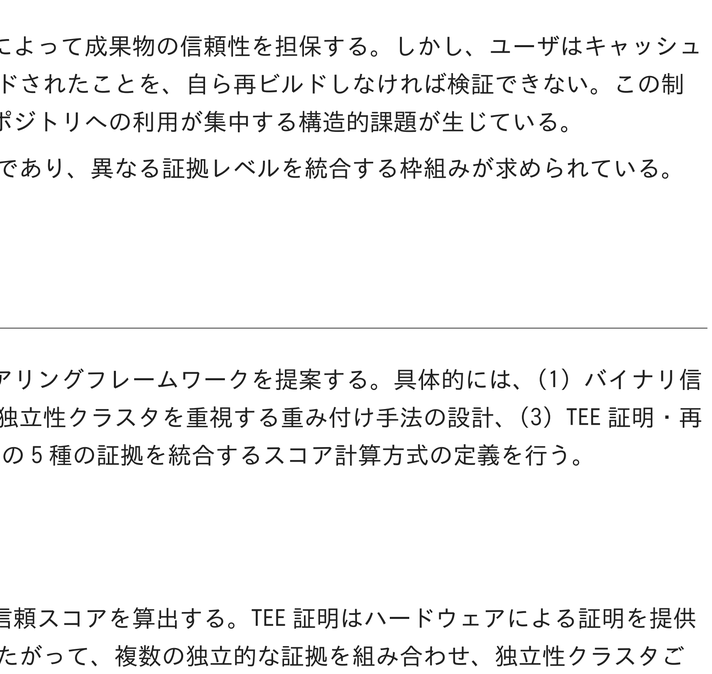
Nix バイナリキャッシュ向けの多証拠信頼スコアリングフレームワークを提案する。具体的には、(1) バイナリ信頼から定量的信頼スコアへの移行、(2) 証拠の独立性クラスタを重視する重み付け手法の設計、(3) TEE 証明・再現性検証・透過性ログ・SLSA 証明書・閾値署名の 5 種の証拠を統合するスコア計算方式の定義を行う。

手法

5 種の証拠タイプを収集し、それらを統合して信頼スコアを算出する。TEE 証明はハードウェアによる証明を提供するが、単独では Sybil 攻撃に脆弱である。したがって、複数の独立的な証拠を組み合わせ、独立性クラスタごとに重み付けを行う。

処理全体は「ビルド実行」「証拠収集」「スコア算出」からなるパイプラインとして構成した。

証拠タイプ	
TEE 証明	Intel TDX, AMD SEV-SNP, AWS Nitro Enclaves
再現性検証	複数独立ビルダの出力一致検証
透過性ログ	Sigstore Rekor, Sigsum, CHAINIAC
SLSA 証明書	in-toto メタデータ, SLSA Lv3+
閾値署名	k-of-n マルチパーティ承認



証拠収集から信頼スコア算出までのフロー

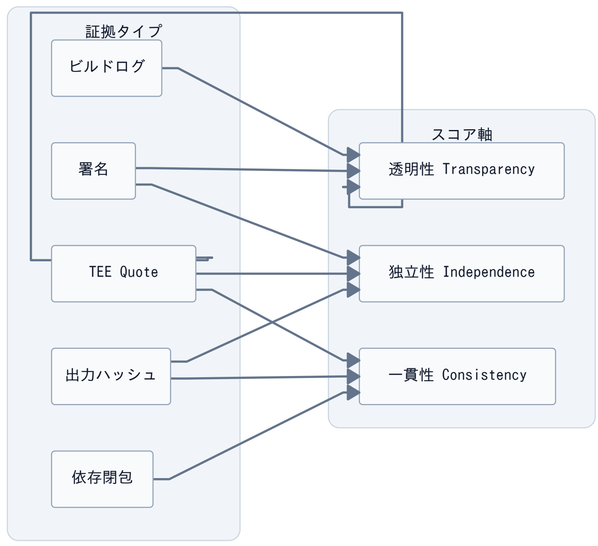
比較

各アプローチの特徴を比較する。提案手法は、複数の証拠タイプを統合し、独立性クラスタによる重み付けを行うことで、いずれの単一アプローチよりも高い信頼性を実現する。

項目	再現性ビルド	透明性ログ	TEE のみ	提案手法
証拠型	出力一致, fill: light	ログ記録	ハードウェア証明	統合
独立性	高い	中程度	低い (Sybil)	高い
検証可能性	ビルド再実行	ログ照合	Quote 検証	多軸検証
実装難度	低い	中程度	高い	中程度
Sybil 耐性	高い	中程度	低い	高い

信頼スコア設計

信頼スコアは 3 つの評価軸に基づいて計算される。一貫性（Consistency）は同一入力から同一出力が得られる度合い、独立性（Independence）は証拠ソース間の運営的・技術的独立性、透過性（Transparency）は外部検証可能性の度合いを評価する。各軸のスコアを独立性クラスタごとに集約し、生の証拠数よりも独立性を重視する重み付けを適用する。

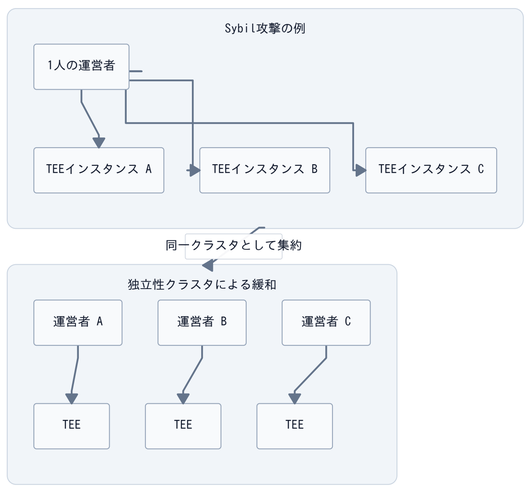


スコア軸と証拠タイプのマッピング

証拠レコード構造	
導出ハッシュ	derivation hash
ソースリビジョン	source revision
依存閉包ハッシュ	dependency closure hash
出力ハッシュ	output hash
ビルダ証明	builder provenance
ビルドログ	build log
TEE Quote	TEE attestation quote

TEE の限界と脅威

TEE はハードウェアレベルの隔離を提供するが、信頼の唯一の根拠としては不十分である。Sybil 攻撃により、単一の運営者が多数の TEE インスタンスを生成し、独立した証拠源であるかのように偽装できる。また、TEE プラットフォーム自体に既知の脆弱性が存在する。



Sybil 攻撃と独立性クラスタによる緩和

既知の TEE 脆弱性	
SGAxe	SGX 側チャネル攻撃、Enclave メモリ漏洩
Plundervolt	電圧操作による SGX 侵害
Foreshadow	L1 ターミナルフォールトによる読み出し
TEE.fail	TEE プラットフォーム包括的脆弱性分析

考察

独立性クラスタによる重み付けは、生の証拠数よりも信頼性の高い評価を提供する。同一運営者が複数の TEE インスタンスを起動しても、それらは 1 つのクラスタとして集約されるため、Sybil 攻撃に対してロバストである。

Nixpkgs の再現性は 2023 年時点で 91% のビットレベル再現性と 99.8% の再ビルド可能性を達成しており、再現性検証を証拠として活用する基盤は整っている。しかし、現在の構造的課題として、大規模リポジトリへの信頼集中が進んでおり、信頼スコアによる分散化の促進が期待される。

今後の課題・展望

今後の課題として、まず信頼スコアリングのプロトタイプ実装を開発し、Nix の substitutor プロトコルへの統合を行う必要がある。また、新しい証拠が追加された際の動的スコア更新手法の設計も求められる。

さらに、ビルダの独立性をどのように評価・検証するかの手法論的整理が必要である。TEE 技術の進化（Intel TDX 第 2 世代、AMD SEV-SNP v3 等）に伴う信頼モデルの変化にも継続的に対応していく必要がある。

参考文献

[1] NixOS, “Nix Manual — Trusted Binary Caches.” nixos.org/manual/nix/stable

[2] SLSA, “Supply-chain Levels for Software Artifacts.” slsa.dev

[3] in-toto, “A framework to secure software supply chains.” in-toto.io

[4] Sigstore, “Rekor Transparency Log.” sigstore.dev

[5] Lau et al., “CHAINIAC: Proactive Software Supply Chain Integrity.” IEEE Symposium on Security and Privacy, 2019.

[6] Reproducible Builds, “Nixpkgs Reproducibility Status.” r13y.com

[7] Moghimi et al., “SGAxe / Plundervolt / Foreshadow — TEE Vulnerability Papers.” tee.fail

[8] Coppieters et al., “Attestable Builds: Reproducible Builds + TEE.” ACM CCS, 2024.