CleanStart

SLSA Levels as Architectural States of Trust

Reframing SLSA from compliance tiers to provable states of trust in software creation

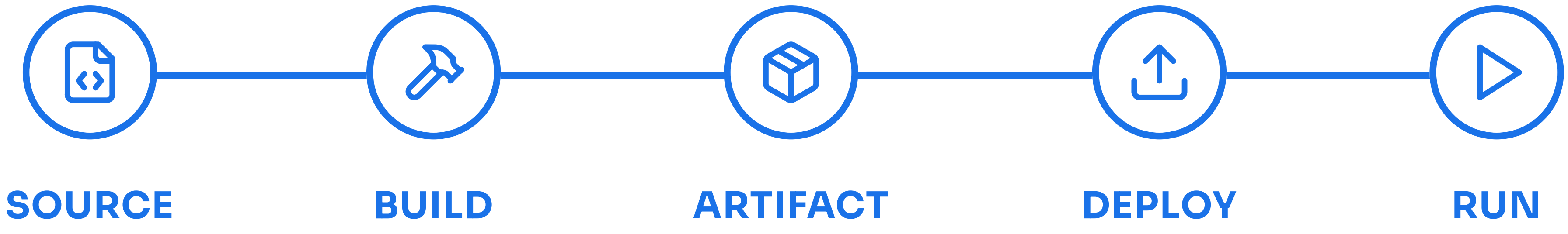
v2026.01.16

Why this Matters

Software supply chain attacks are no longer edge cases. They are more expensive to remediate and take longer to detect than traditional breaches because compromise increasingly occurs before software is deployed, during build and packaging.

This exposes a structural problem. Software compromised during assembly cannot be secured later through scanning, runtime controls, or policy enforcement. Architecture, not tooling volume, determines whether trust can be established at all.

Where Trust Is Established



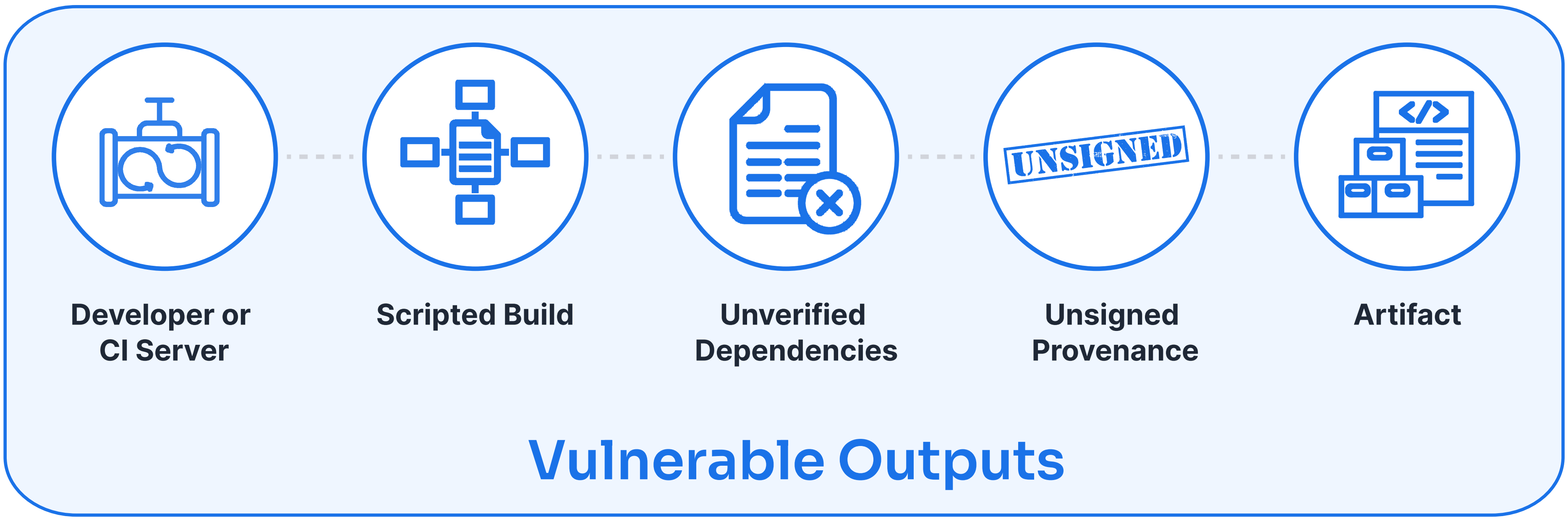
Post-Build: Observe, Not Correct

SLSA Levels 0-1: Visibility Enforced Integrity

At Levels 0 and 1, the architecture focuses on documenting how software is built.

- Builds are automated and scripted
- Provenance is generated describing inputs and steps
- Dependencies are referenced but not verified
- Builds may run on developer machines or shared infrastructure

This stage improves consistency and auditability, but it does not meaningfully constrain attacker behavior.



Architectural Limitation

Provenance exists, but it is informational rather than enforceable. An attacker who can influence dependencies, the build environment, or artifact storage can still produce malicious outputs that appear legitimate.

This class of failure has appeared repeatedly across the ecosystem. Build processes that were documented and automated still produced compromised artifacts because the architecture allowed unverified inputs and trusted mutable environments.

Trust Model

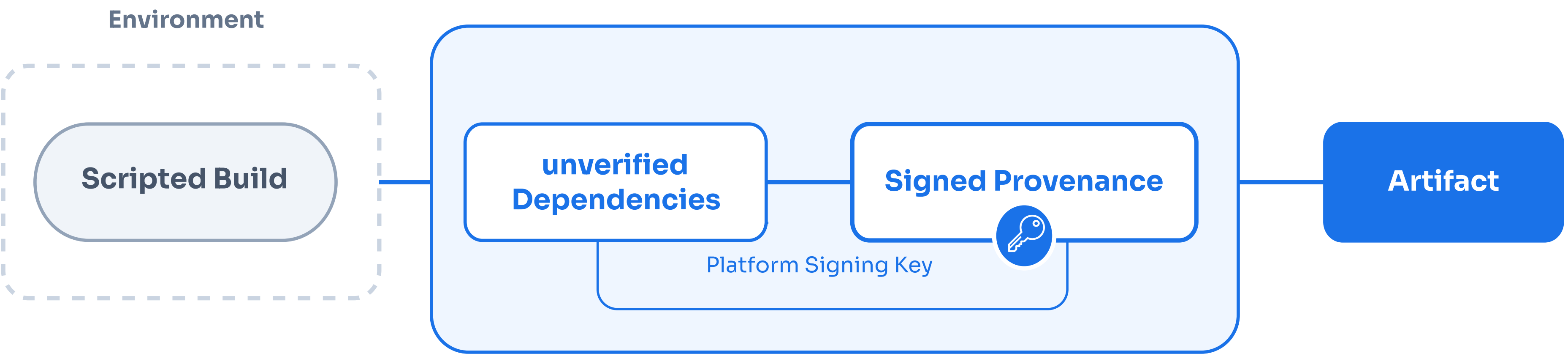
The system assumes the build process and environment are trustworthy.

SLSA Level 2: Cryptographic Proof Without Build Integrity

Level 2 introduces cryptographic signing of provenance and shifts builds to hosted platforms.

- Provenance is cryptographically signed
- Builds run on centralized CI systems
- Artifact tampering after the build becomes detectable

This significantly improves detection. Consumers can verify that artifacts match recorded build metadata and that neither has been altered post-build.



Trusted Outputs

Architectural Impact

The integrity of the build platform itself is still assumed. If the CI system or its signing infrastructure is compromised, it can still generate valid provenance for malicious artifacts.

Several high-profile supply chain incidents illustrate this failure mode. In cases such as SolarWinds and the XZ Utils backdoor, attackers did not bypass scanning or signing. They compromised trusted build paths, producing artifacts that passed downstream validation.

Trust Model

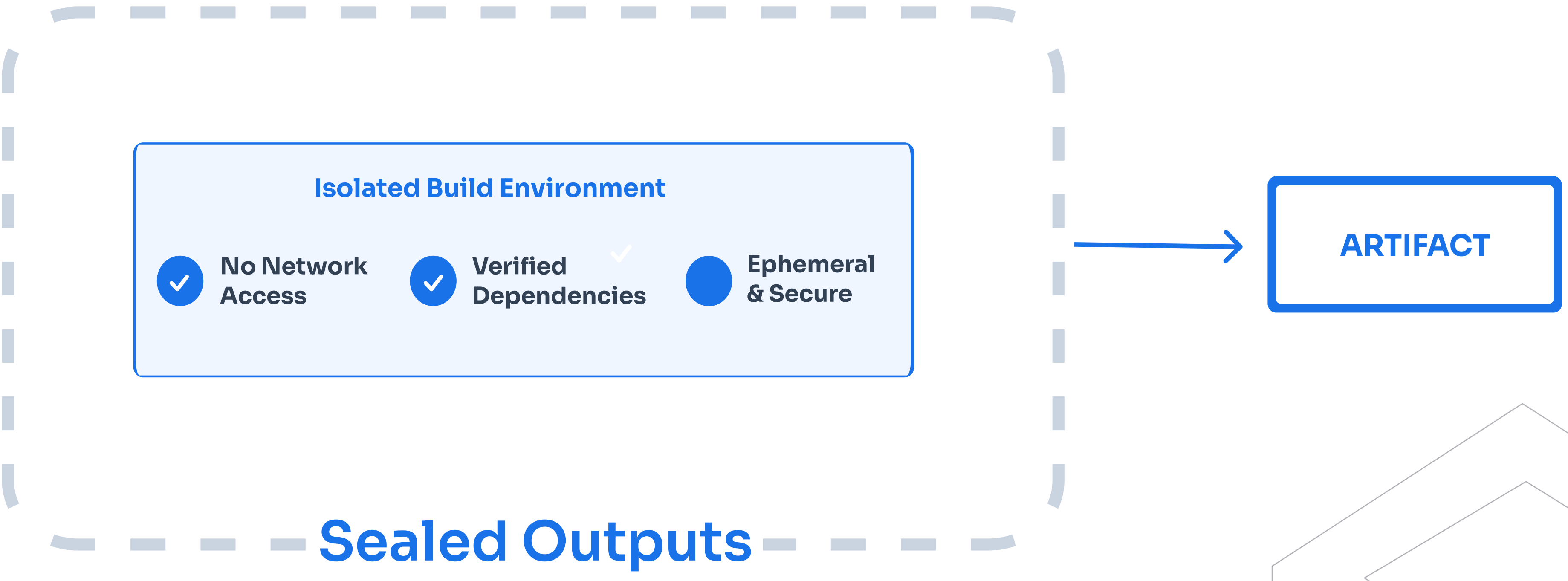
Trust shifts from individual developers to the build platform.

SLSA Level 3: Build Integrity as a Design Constraint

Level 3 represents the first fundamental architectural shift. The build process itself becomes a protected system.

- Builds are hermetic, with no network access during execution
- Dependencies are declared and verified before build time
- Build environments are isolated, ephemeral, and non-persistent
- Platforms are hardened to prevent cross-build interference

At this level, entire classes of supply chain attacks become structurally infeasible. Dependency confusion, build-time injection, and persistence-based contamination are prevented by design rather than detected after the fact.



Architectural Impact

Security moves upstream into software assembly. Risk is constrained before an artifact exists, rather than managed after deployment.

Implementation Implication

This architecture forces explicit dependency declaration, controlled input resolution, and isolated execution. These are not optional best practices. They are required consequences of moving the trust boundary into the build itself.

Trust Model

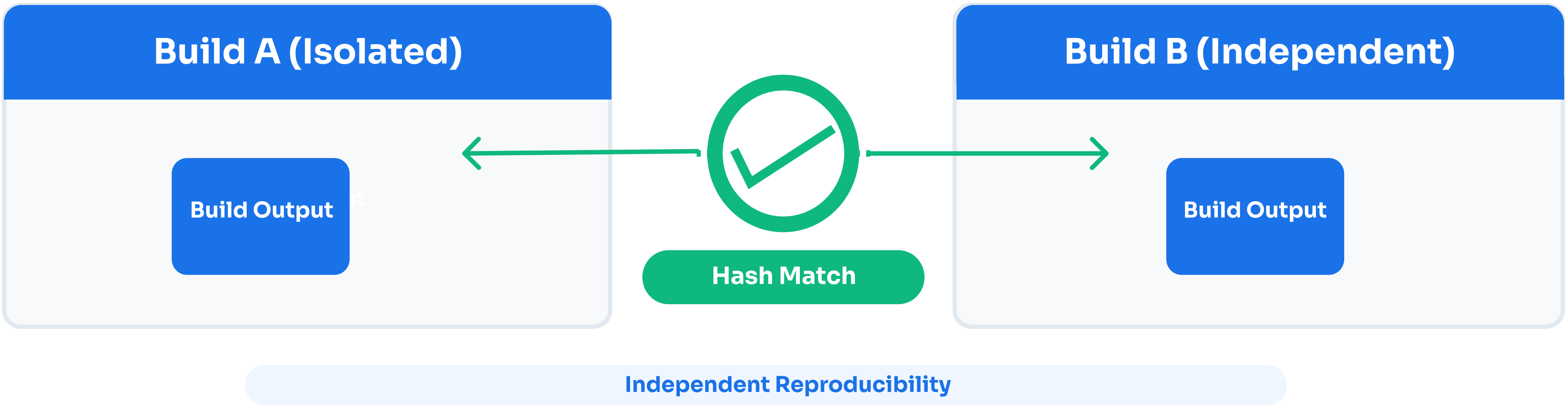
Only declared, verified inputs are allowed to influence the build.

SLSA Level 4: Independent Verification & Trust Reduction

Level 4 removes reliance on any single actor or system.

- Two-person review is enforced for all changes
- Builds are reproducible and deterministic
- Artifacts can be independently rebuilt and verified
- Audit logs provide complete, tamper-evident traceability

At this level, trust no longer depends on who built the software or which platform produced it. Integrity is established through independent verification.



Architectural Impact

Insider threats and platform compromise cease to be silent failure modes. Any deviation between independently rebuilt artifacts becomes immediately observable.

Ecosystem Alignment

This shift toward reproducibility and verifiable provenance is increasingly reflected across open-source foundations, hyperscalers, and regulated procurement frameworks. The industry is converging on verification as a prerequisite for trust, not a differentiator.

Trust model

Trust is derived from reproducibility and verification, not reputation.

CleanStart's Architectural Alignment to SLSA

CleanStart is designed around SLSA Level 3 and Level 4 as baseline assumptions, not maturity goals.

- Hermetic, isolated builds by default
- Pre-verified and immutable dependencies
- Ephemeral build environments with no network access
- Cryptographically signed, detailed provenance

Reproducible builds and enforced two-person review

These controls are embedded into the platform rather than implemented per pipeline. Teams do not retrofit SLSA controls into CI workflows. They consume artifacts that already meet these trust requirements.

Operational Implications

This architectural approach changes how teams operate:

- Security teams validate provenance instead of chasing
- vulnerability noise

Platform teams simplify pipelines by removing fragile

- enforcement logic

Compliance teams receive audit evidence as a natural build

- output

Developers stop inheriting upstream risk by default

Security is enforced before software is created, not retrofitted afterward.

Key Takeaway

SLSA levels are not incremental improvements.
They define **architectural trust boundaries**.

Below Level 3	compromise remains possible by design.
At Level 3	compromise becomes structurally difficult.
At Level 4	trust becomes independently verifiable.

That is the architectural difference between observing risk and proving integrity.

References

1. SLSA Framework (Supply-chain Levels for Software Artifacts)
2. SLSA v1.0 Specification
3. SolarWinds Supply Chain Compromise
4. XZ Utils Backdoor Incident Analysis (2024)
5. Sigstore Project
6. Reproducible Builds
7. NIST SP 800-218