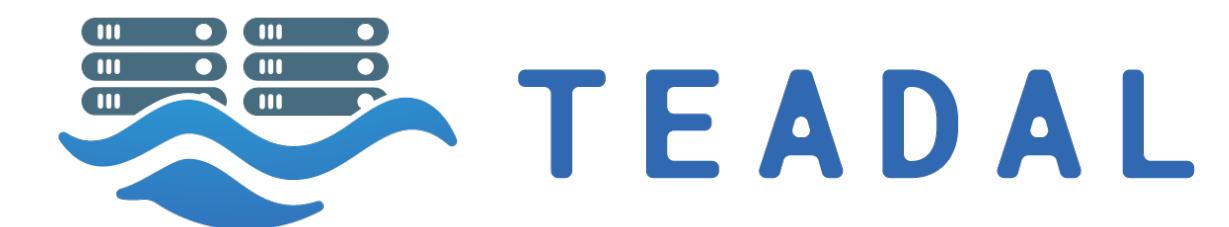


Continuously Building Trustworthy Software

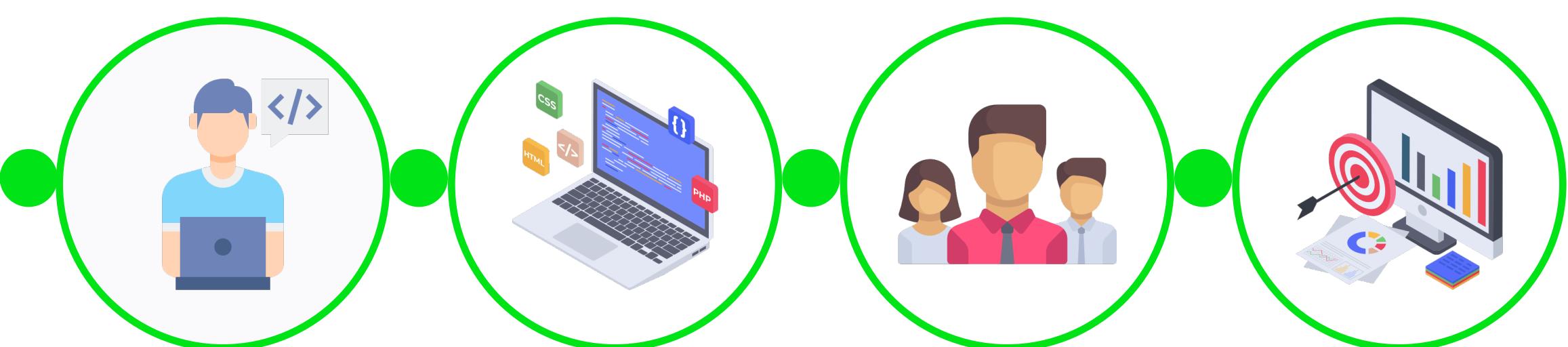
Eduardo Brito, Fernando Castillo, Pille Pullonen-Raudvere, Sebastian Werner



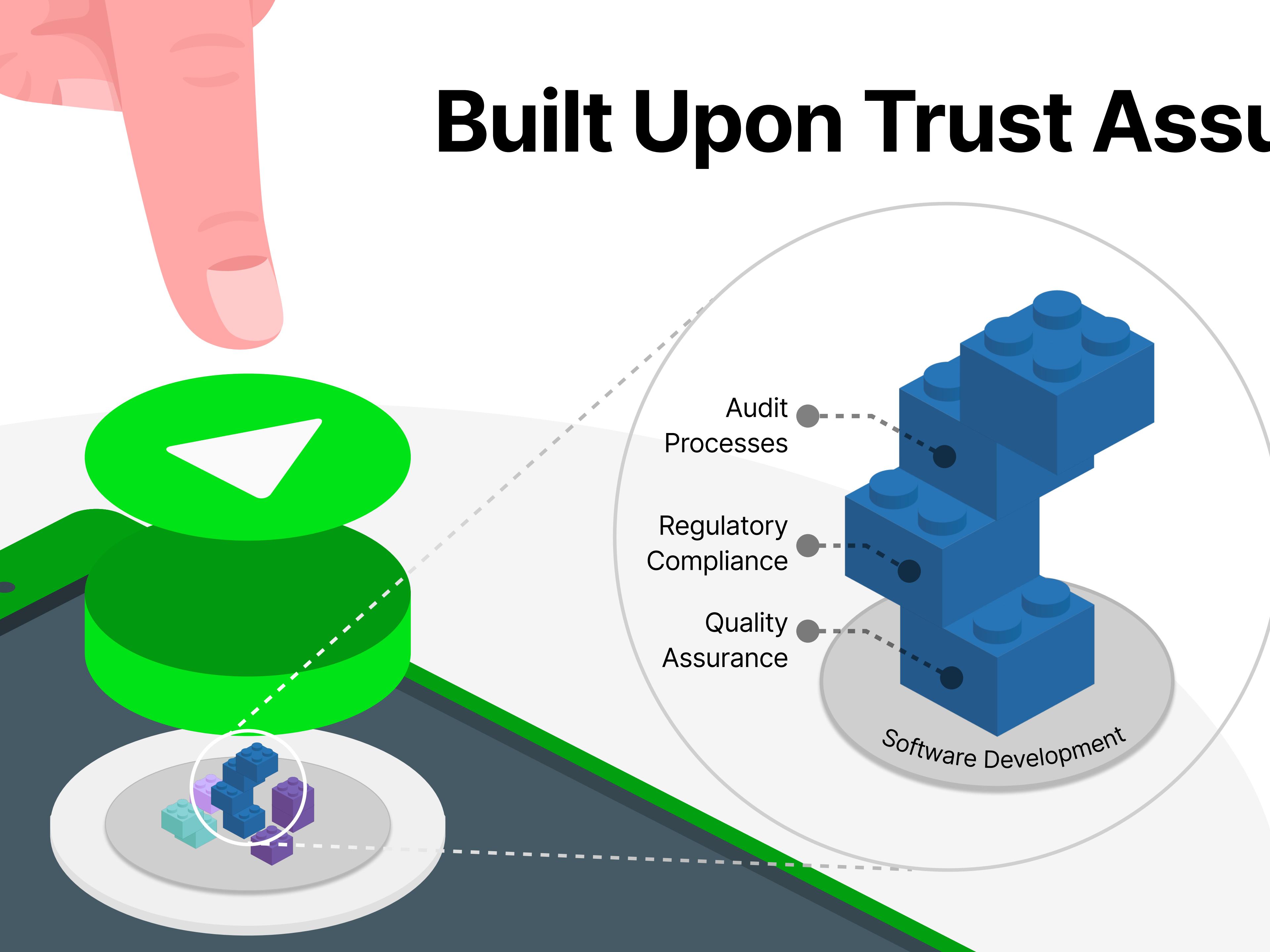
The Ubiquitous Trust Tap



- Current software ecosystems rely on **implicit trust models** that present **increasing risks**.
- **Automated trust** is a necessary paradigm to enforce **verifiable** development and operational **processes**.
- **TrustOps** offers an **evidence-based approach** to building and verifying **software trustworthiness**.
- **Trust-enhancing mechanisms** can be progressively **integrated** into the **software life cycle**.

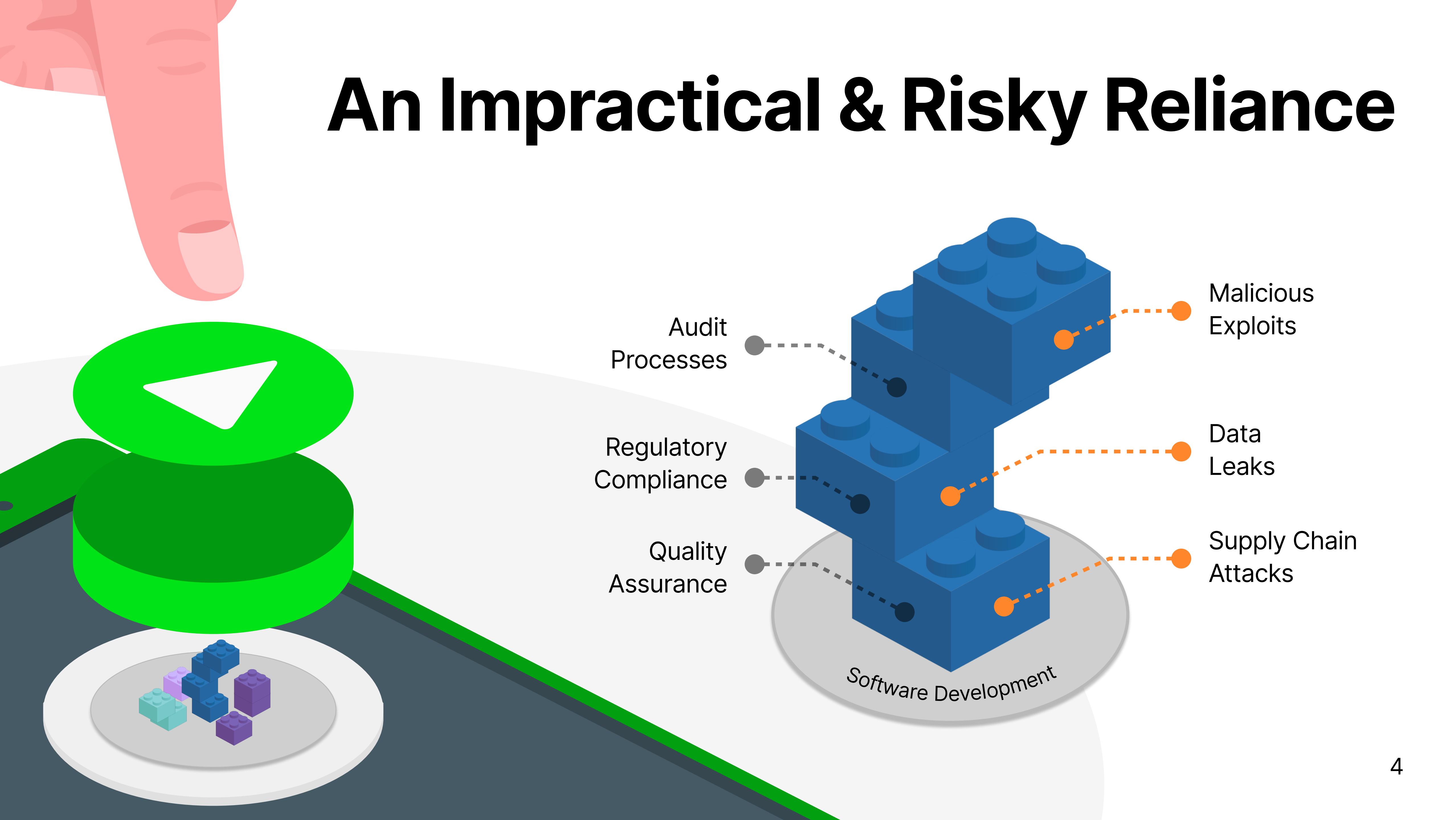


Built Upon Trust Assumptions

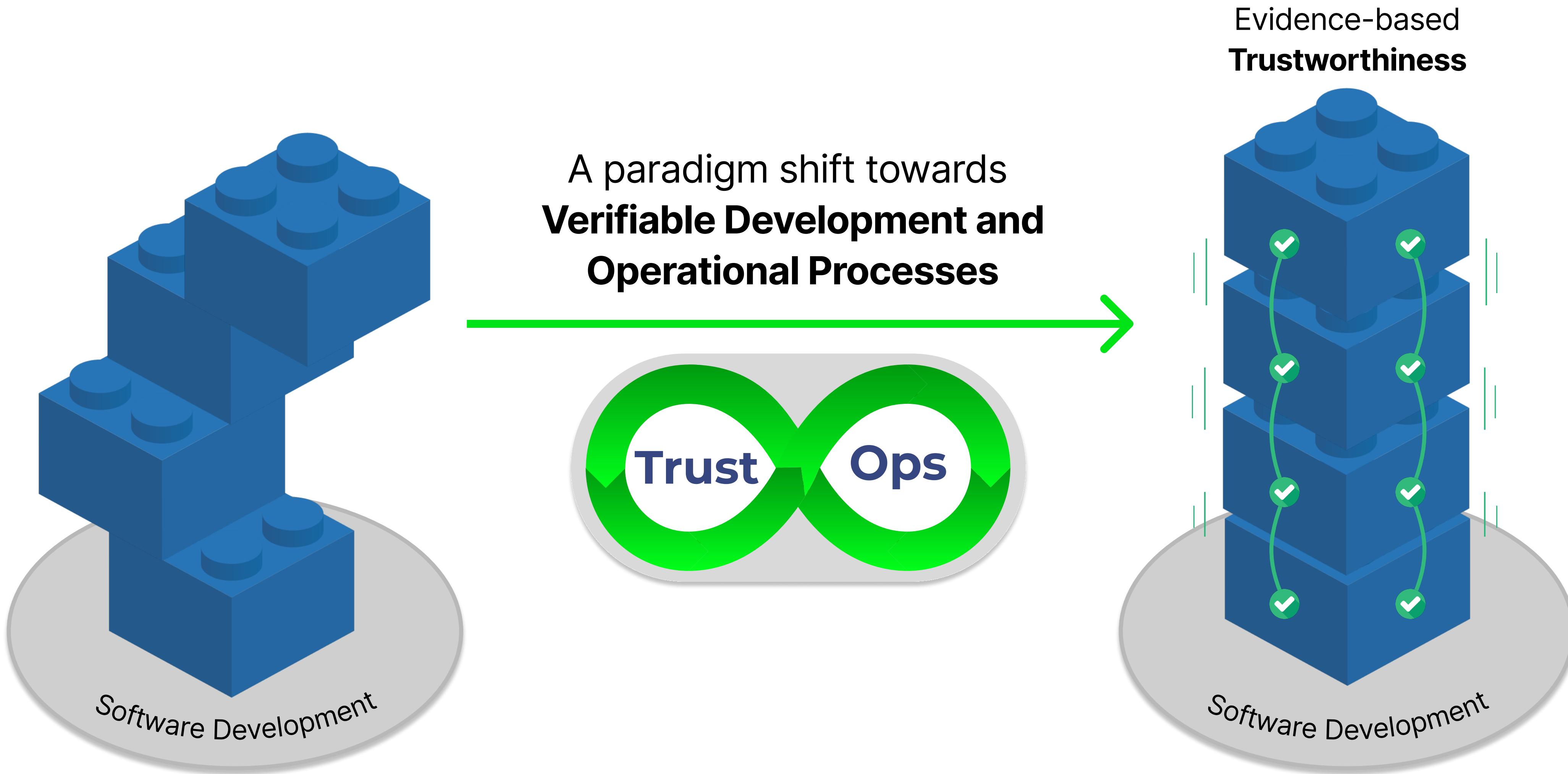


Mainstream methodologies such as **Continuous Compliance**, **DevOps**, **DevSecOps**, adherence to **Industry Standards**, and adoption of **Agile** processes, have been contributing to enhance **Perceived Trust**.

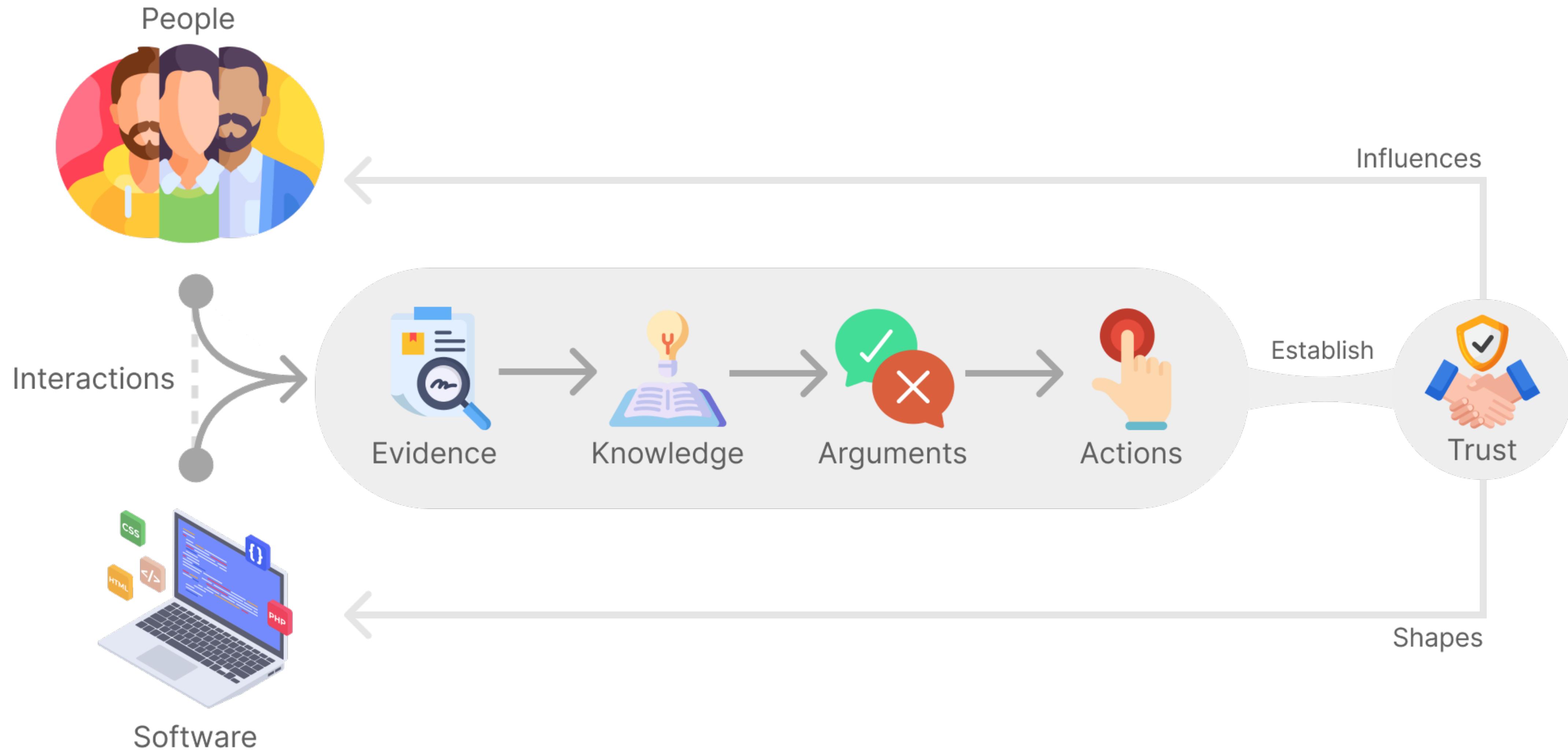
An Impractical & Risky Reliance



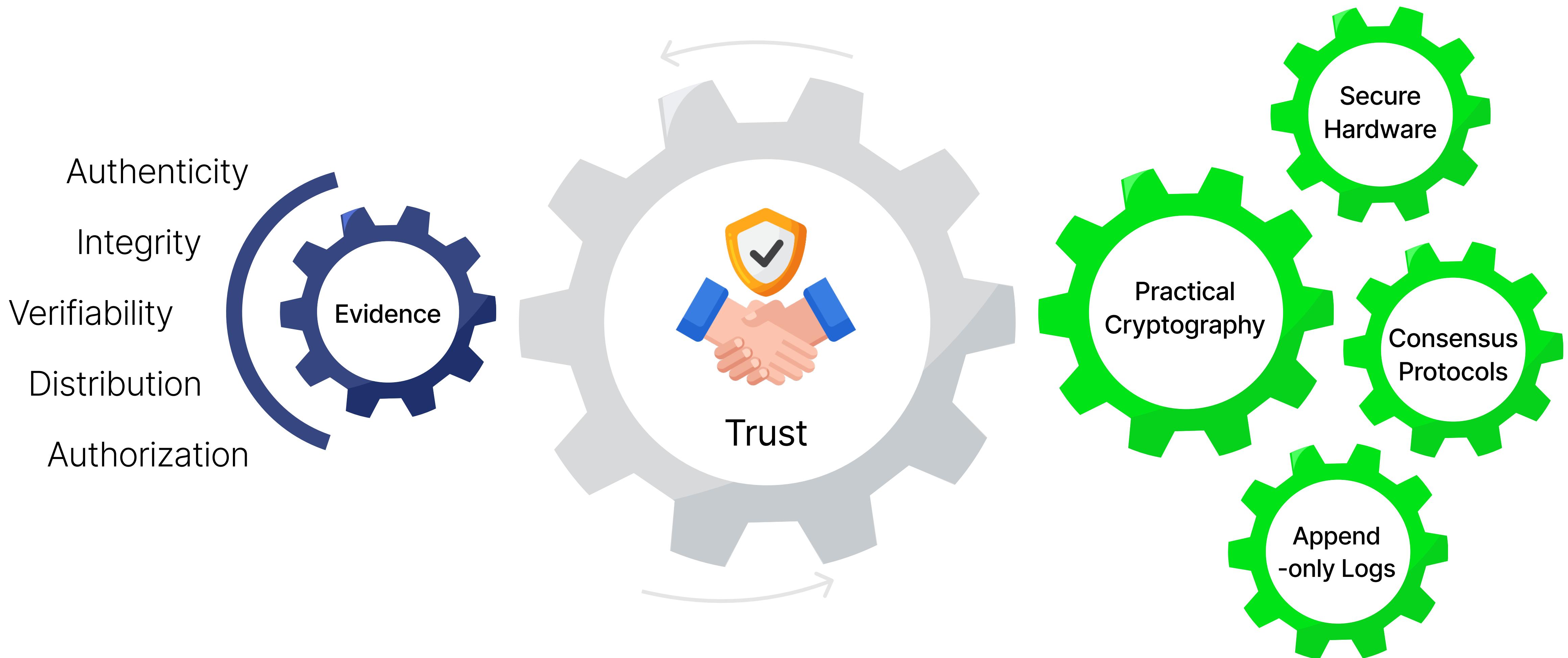
The Need for a New Trust Model



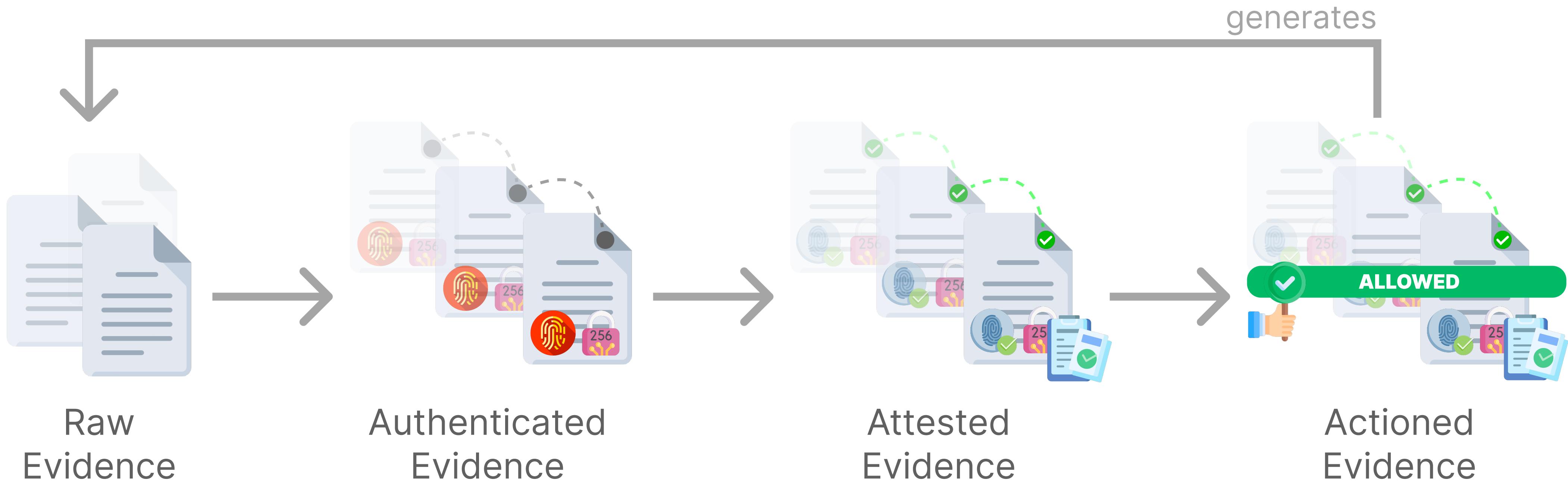
A Socio-Technical Phenomenon



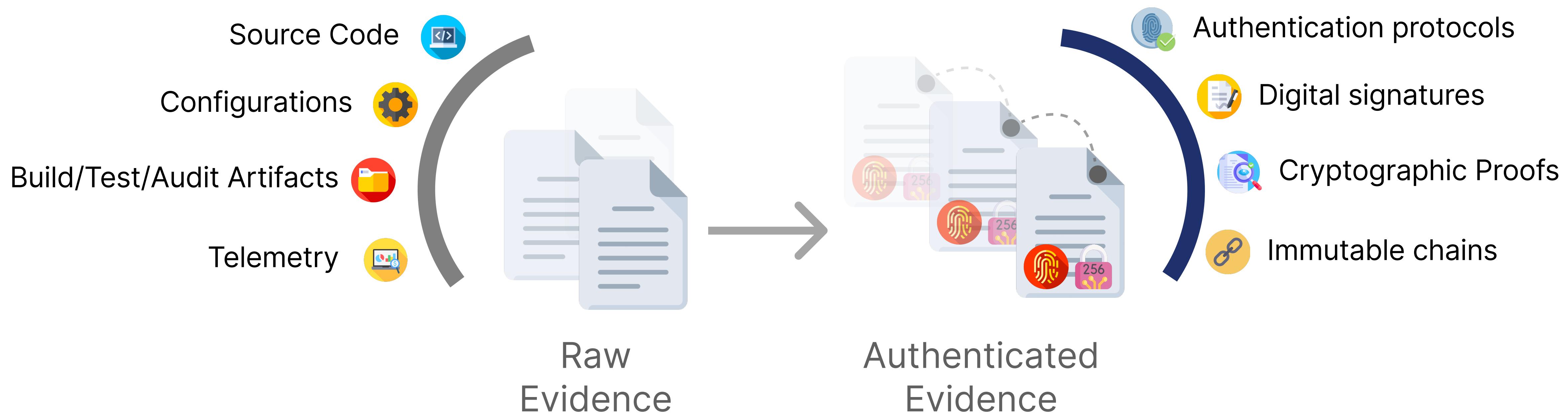
The Automated Trust Paradigm



The Life Cycle of Evidence

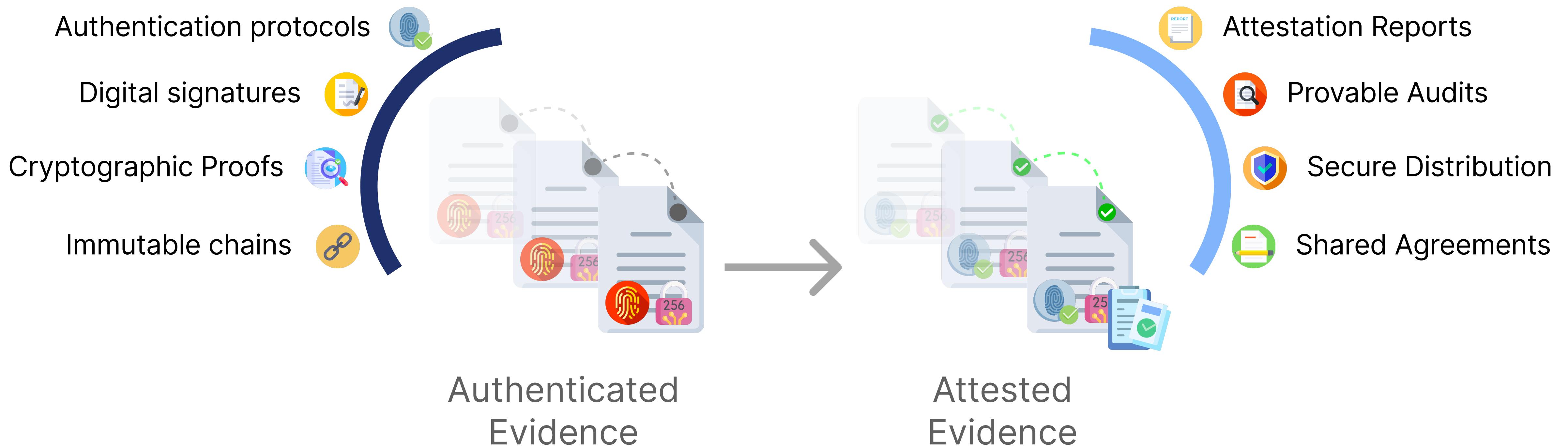


From Raw to Authenticated Evidence



Through an **identity-centric** approach, **enhancing evidence** with contextual metadata, and employing **cryptographic mechanisms** to ensure **authenticity and integrity** of all collected knowledge.

From Authenticated to Attested Evidence



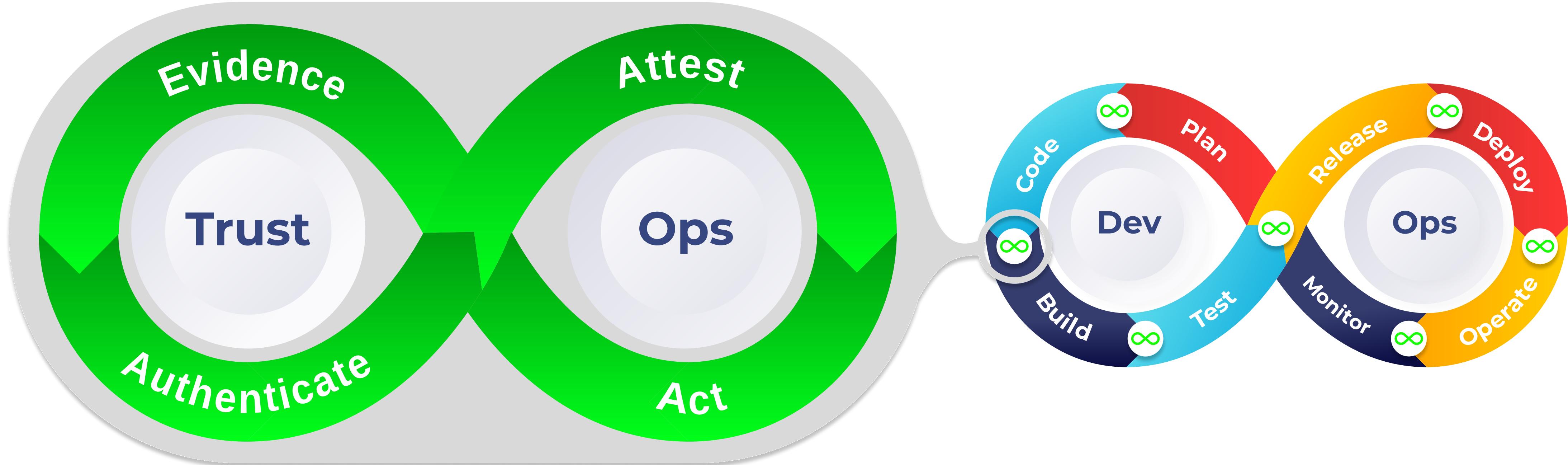
Through **verifiability and consensus** mechanisms, enabling **independent validation** of software properties and actions, leveraging **advanced technologies** like TEEs and ZKPs for **continuous, automated attestation**.

From Attested to Actioned Evidence



Through **automated actions** like authorization policy definition and access control enforcement, supporting **chainable and verifiable accountability, explainability, and observability** in software development and operations.

Augmenting the DevOps Life Cycle



Continuously Building Trustworthy Software

From Planning to Coding



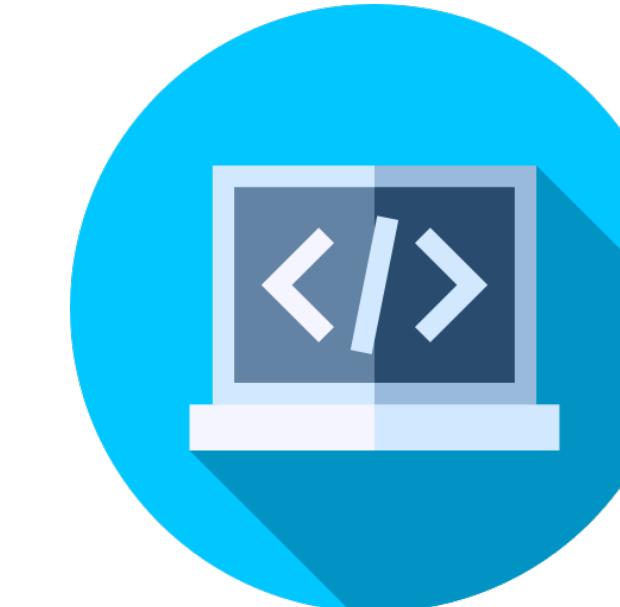
Issues, Tickets, User Stories,
Requirements, Risk
Assessment, etc...



Augmenting project
management and
planning tools to support
signed artifacts,
immutable logs, and
automated attestation.



Code, Version Control, Dev
Environment, Formatting, AI
Assistant Configs, etc..



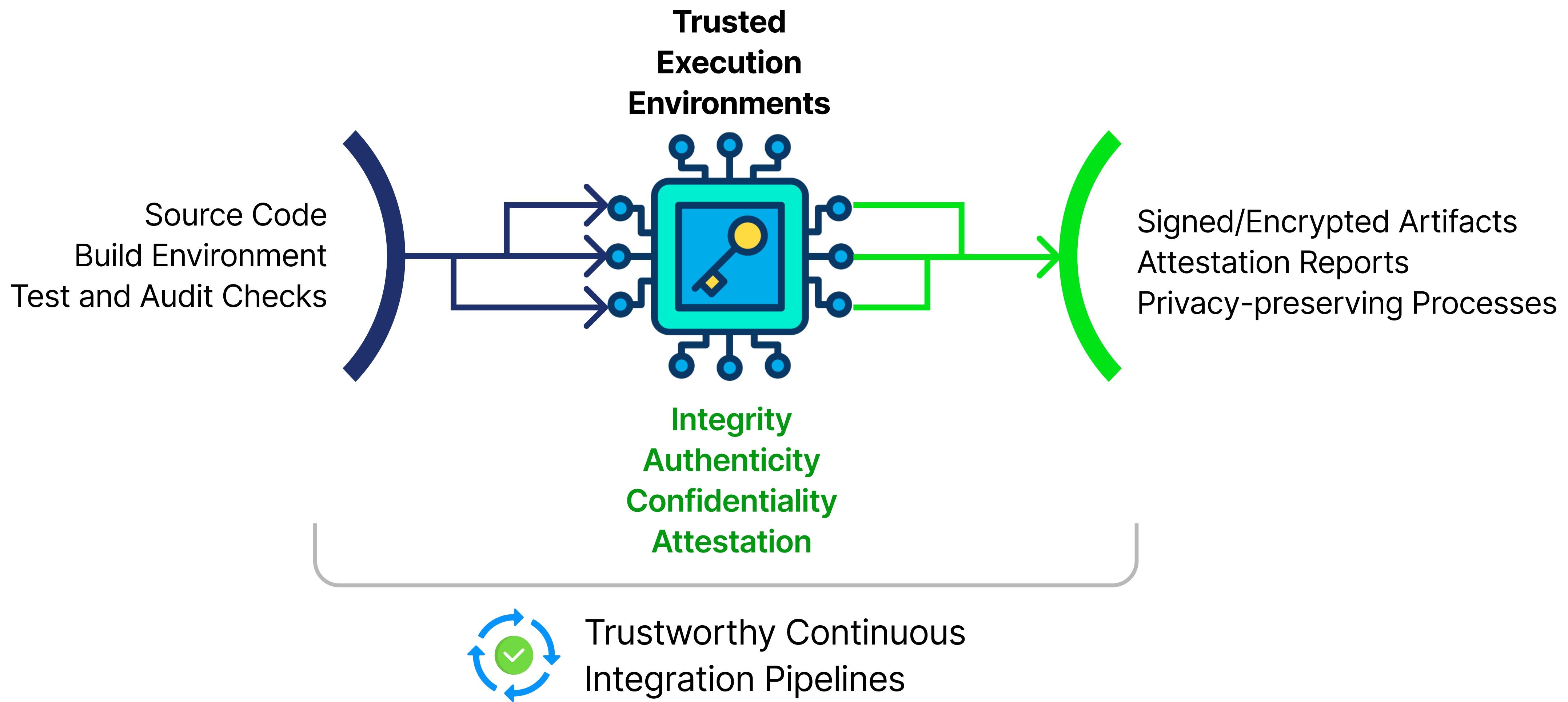
Attested planning evidence
and authorized execution of
planned changes



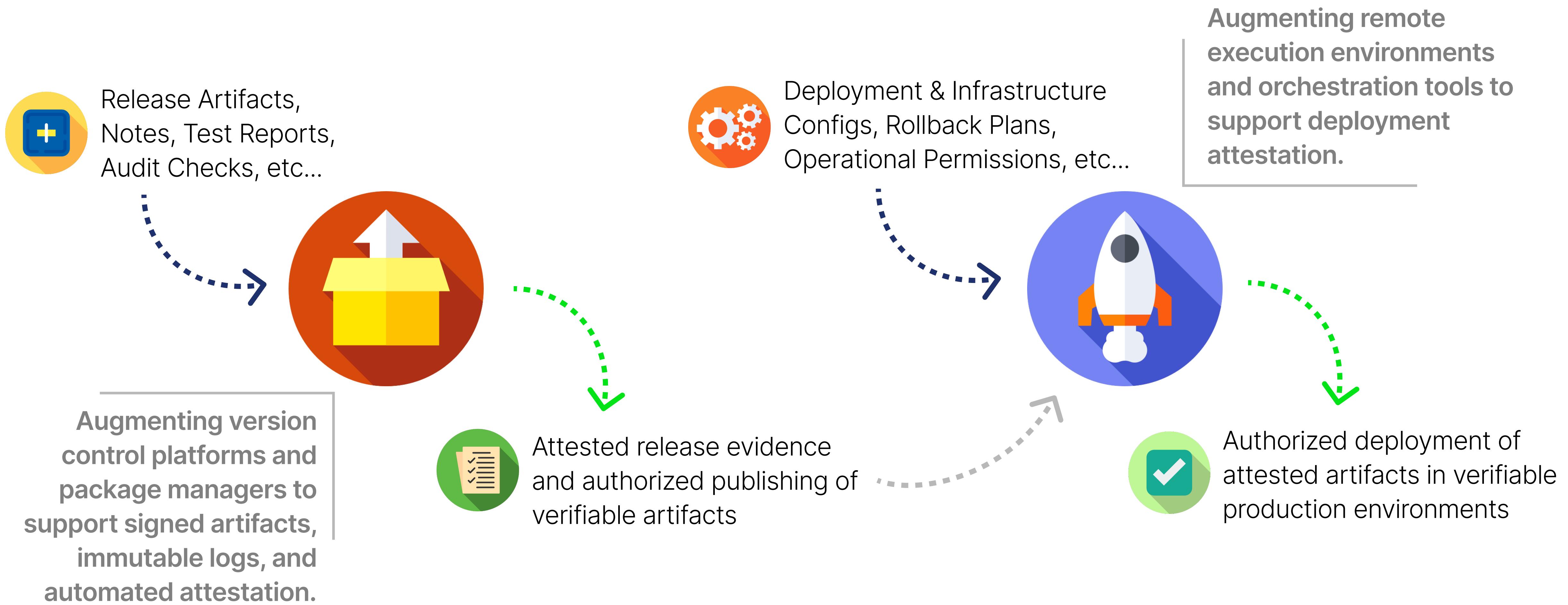
Succinct correctness
proofs of development
practices

Augmenting development
environments to support
evidence provision,
authentication, and
automated attestation.

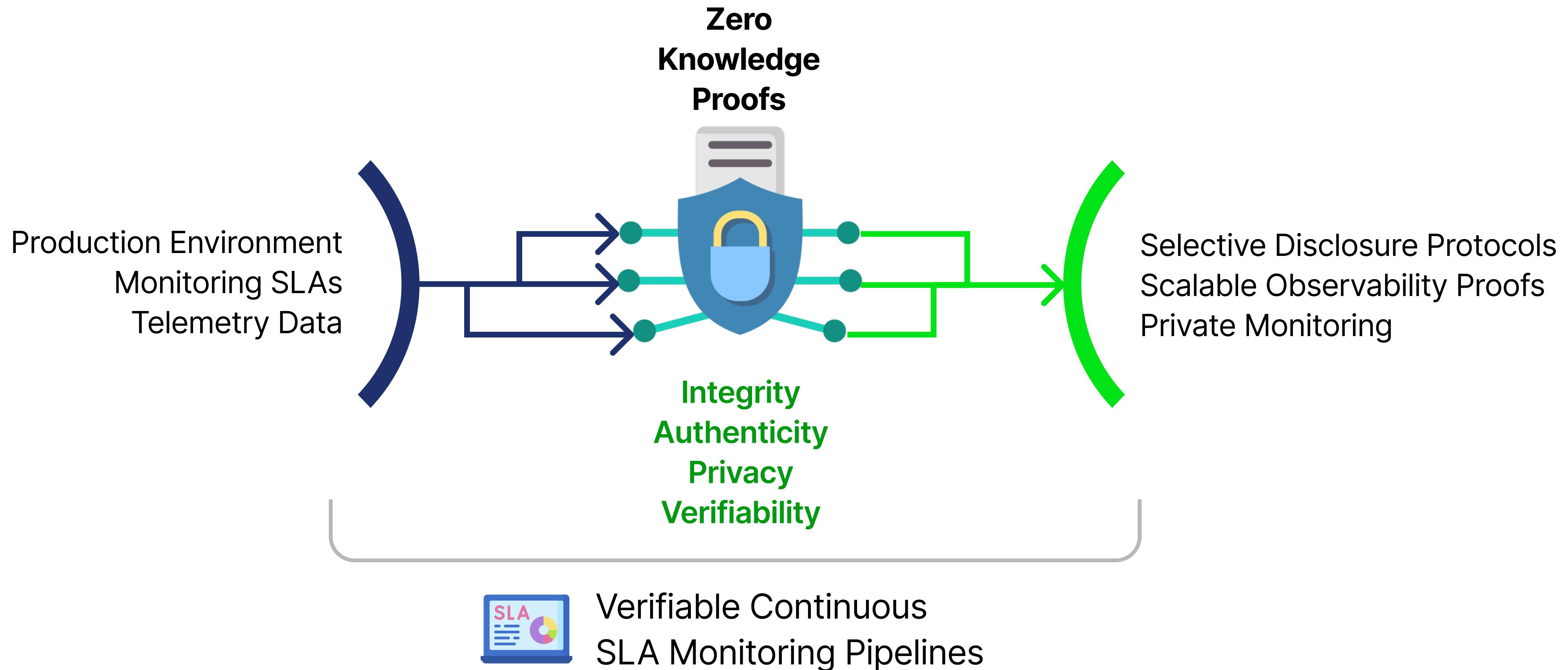
From Building to Testing



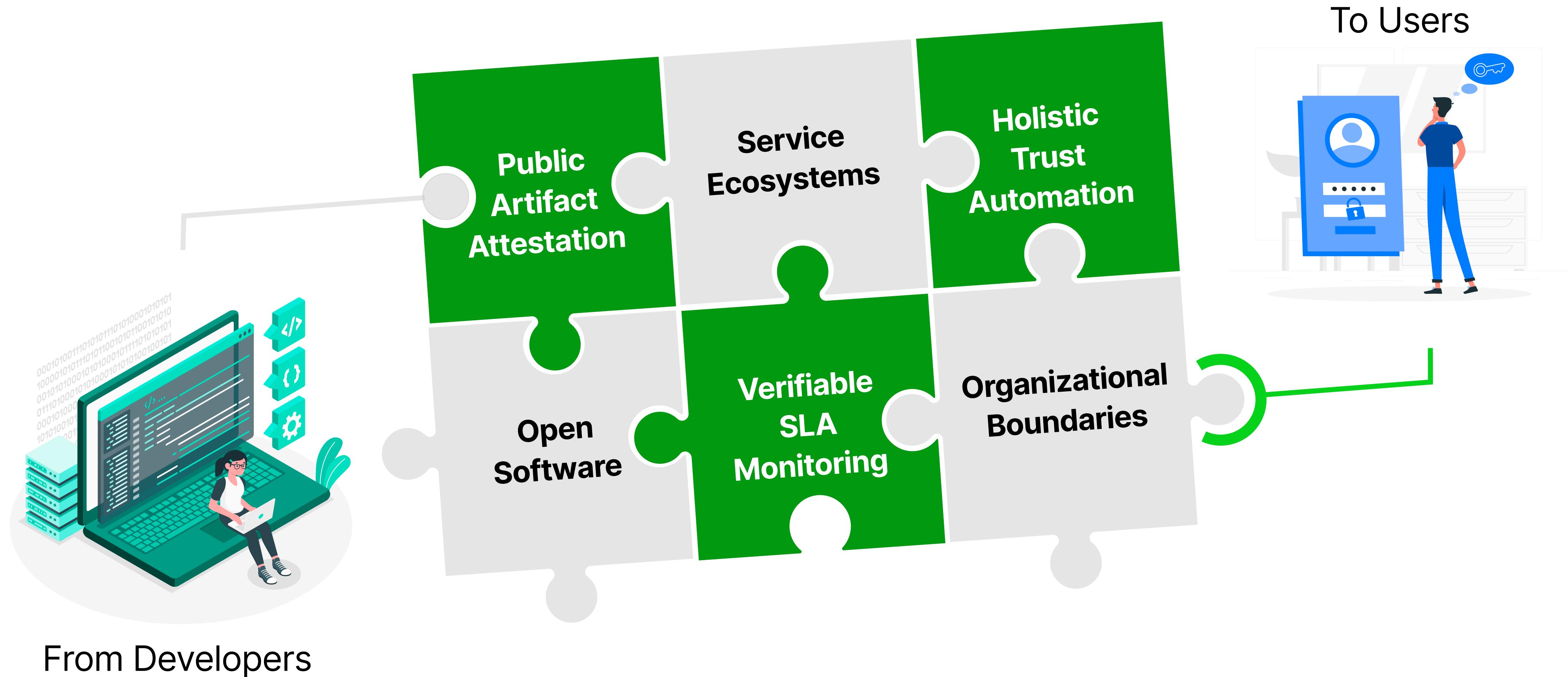
From Releasing to Deploying



From Operating to Monitoring



Aligning the Software Supply Chain



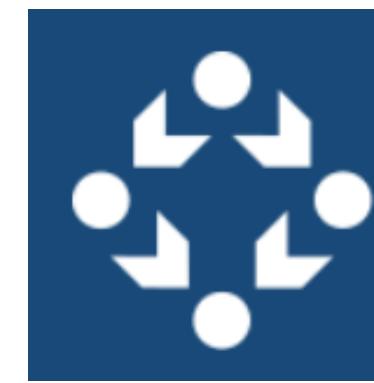
Recent News and Advancements



Nix ecosystem supply chain security project

Nixpkgs, 2023

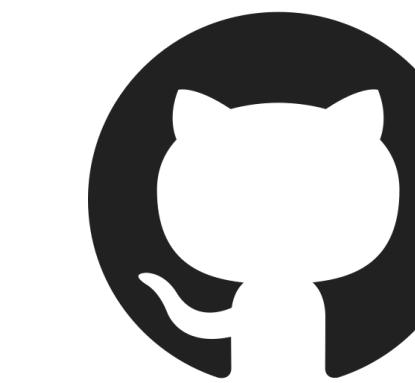
Reinforcing the security framework of Nix open source build and configuration management system across the full software development life cycle.



Reproducible builds as requirement

Reproducible Builds, 2023-2024

Ensuring the integrity of open source software throughout its lifecycle, from source to binary code, safeguarding open source software supply chains.

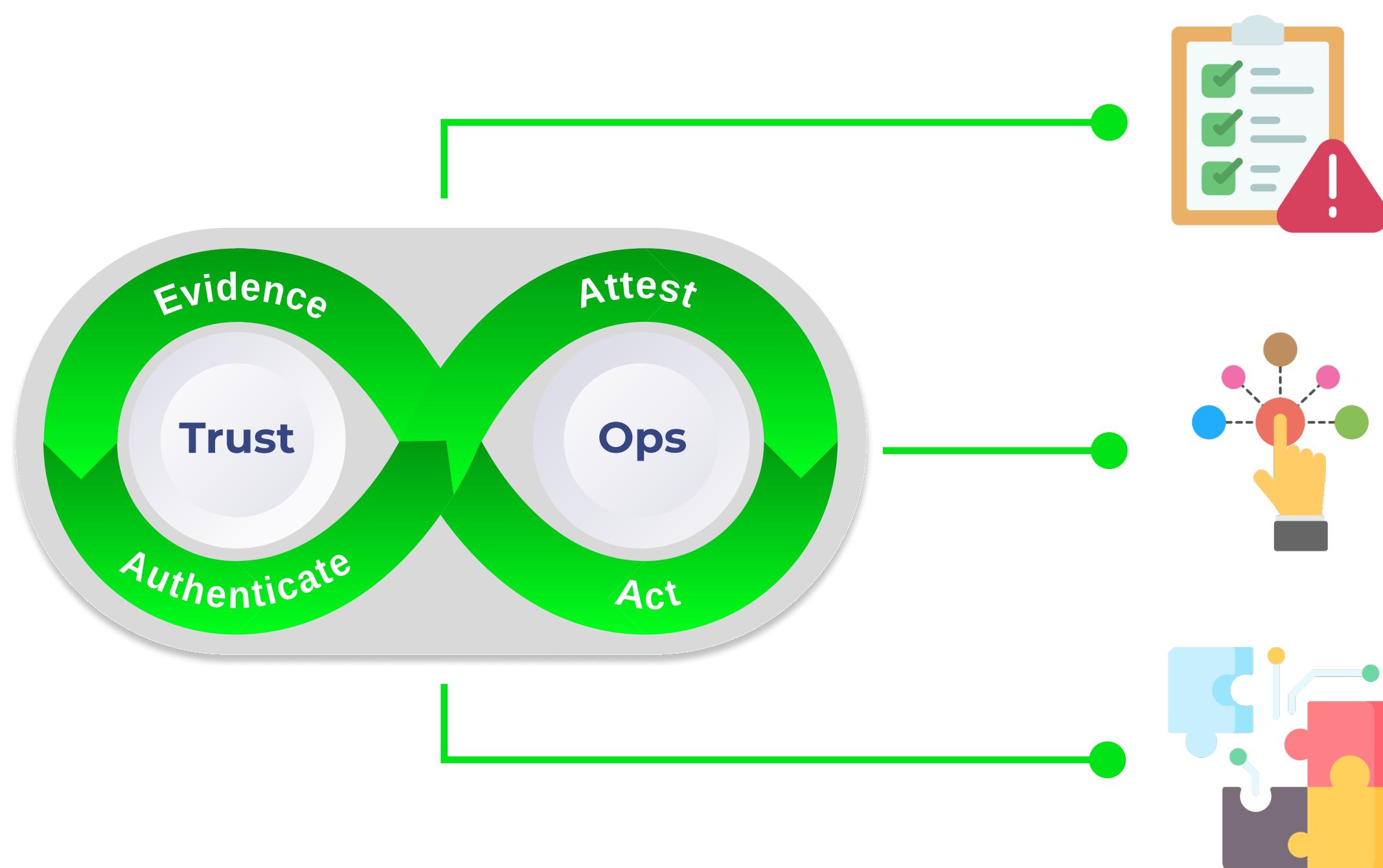


Introducing Artifact Attestations

GitHub, 2024

Providing a tamper-proof link between software artifacts and their source code, powered by the open-source Sigstore project, enhancing software supply chain security.

Research and Adoption Challenges



Evidence Management: Handling and storing TrustOps evidence is complex due to privacy concerns, varied evidence types, and potential interoperability issues.

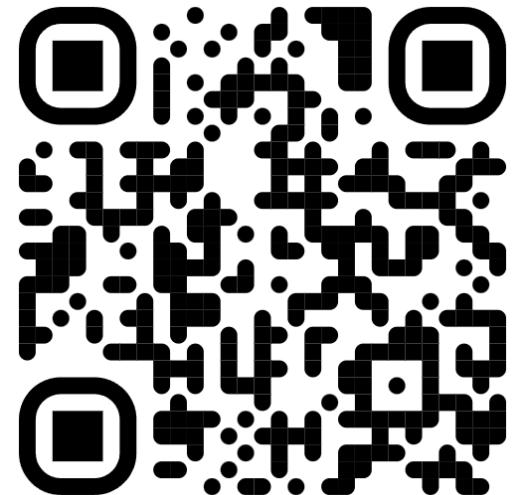
Usability: Improving the usability of technologies like ZKPs and TEEs is essential for TrustOps, alongside thorough threat modeling and user education.

Integration: Successful TrustOps adoption requires coexistence with non-adhering applications and creation of tools and guidelines across the software life cycle.

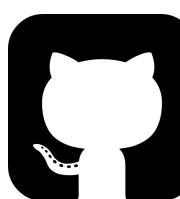


Eduardo Brito, Fernando Castillo, Pille Pullonen-Raudvere, Sebastian Werner

Continuously Building Trustworthy Software



eduardo.brito@cyber.ee



github.com/trustops

