Bytecode Alliance SIG-Registry

Bailey Hayes & Kyle Brown

Introducing Registry Protocol: Warg

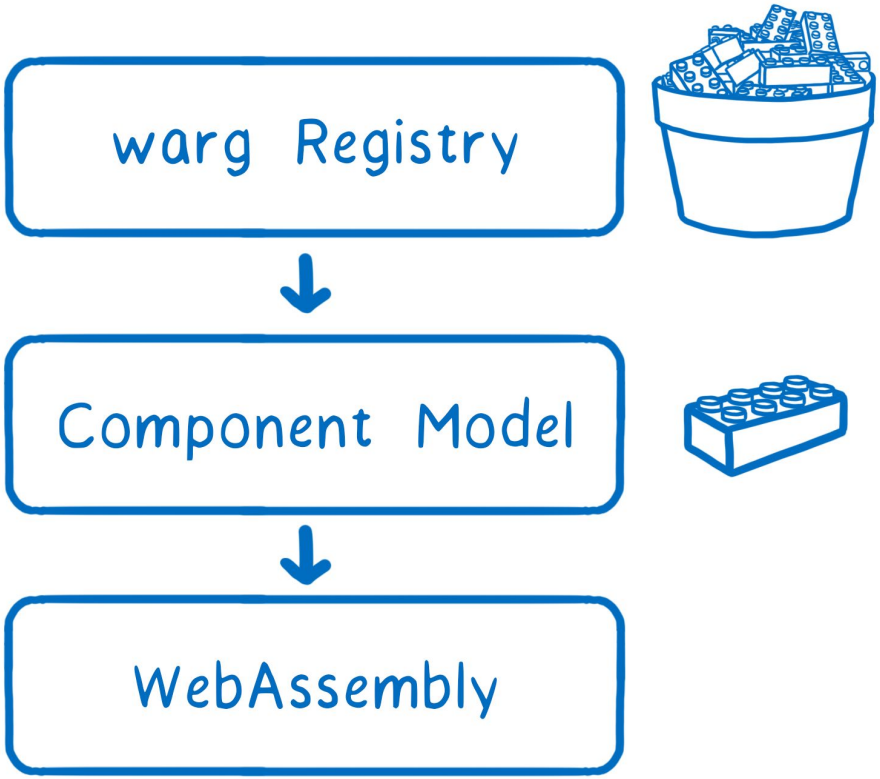


image credit: Lin Clark

Bytecode Alliance
instance

image credit: Lin Clark

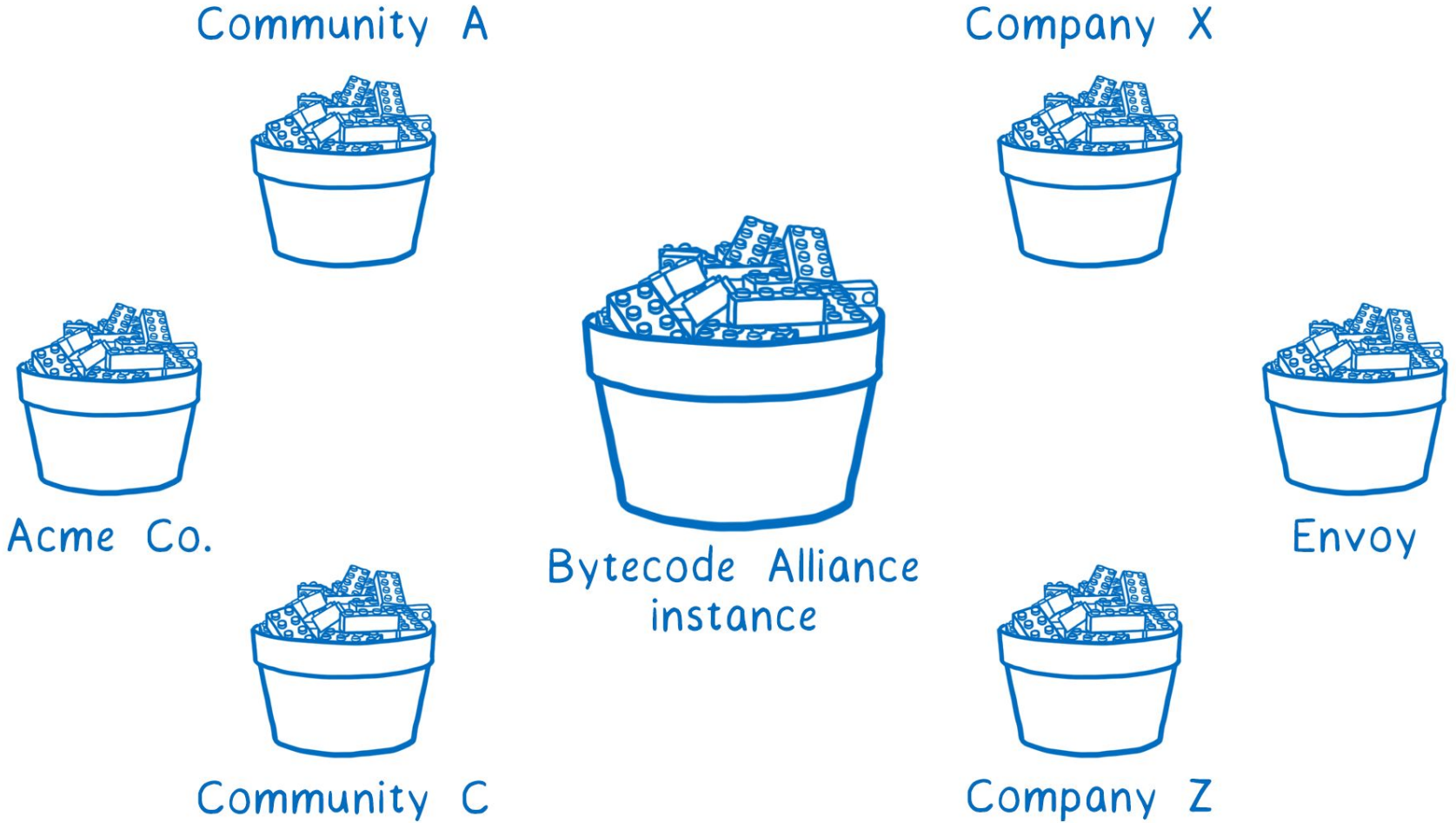


image credit: Lin Clark

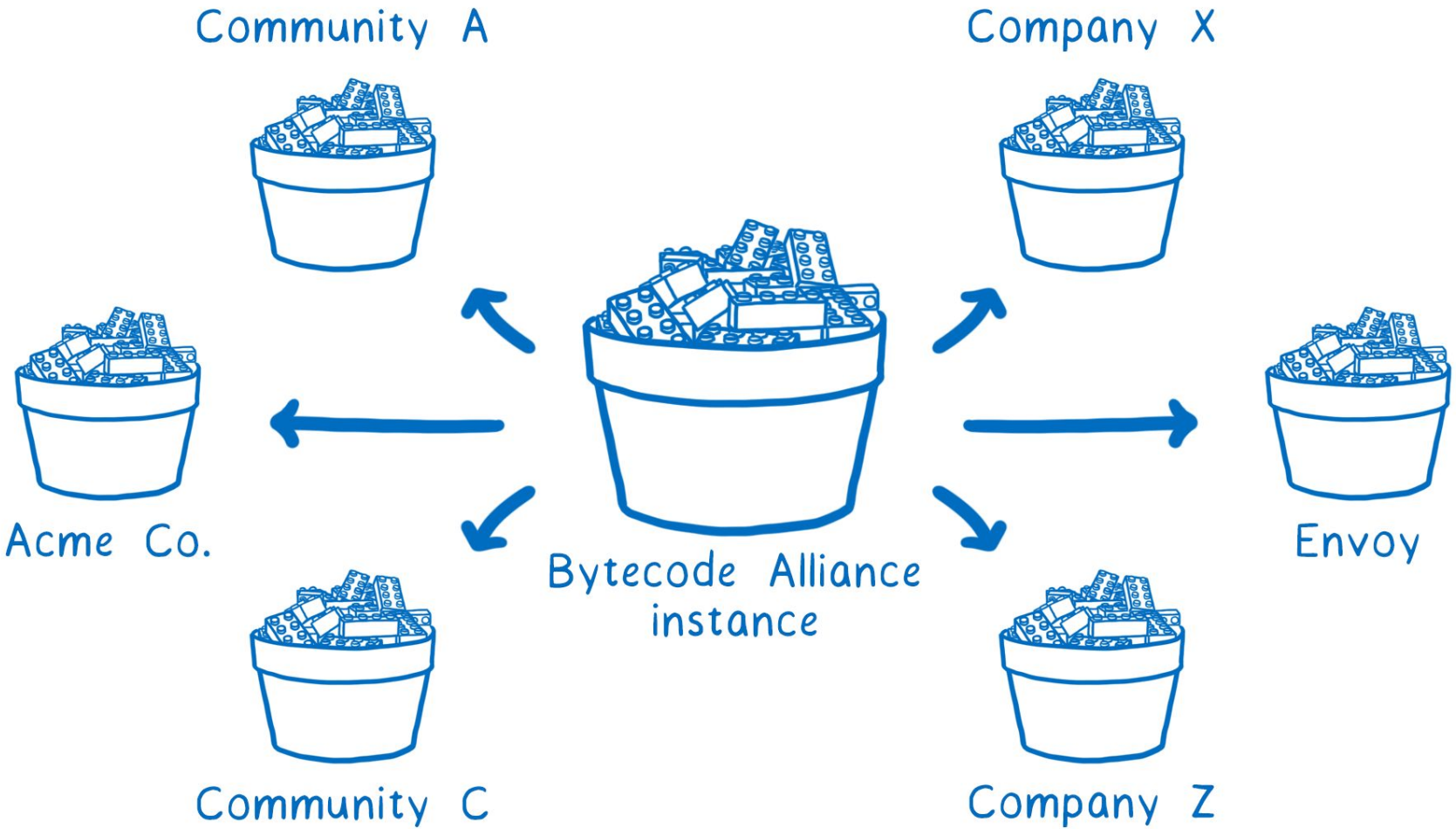


image credit: Lin Clark

Acme Co.

Bytecode Alliance
instance

Envoy

image credit: Lin Clark

image credit: Lin Clark

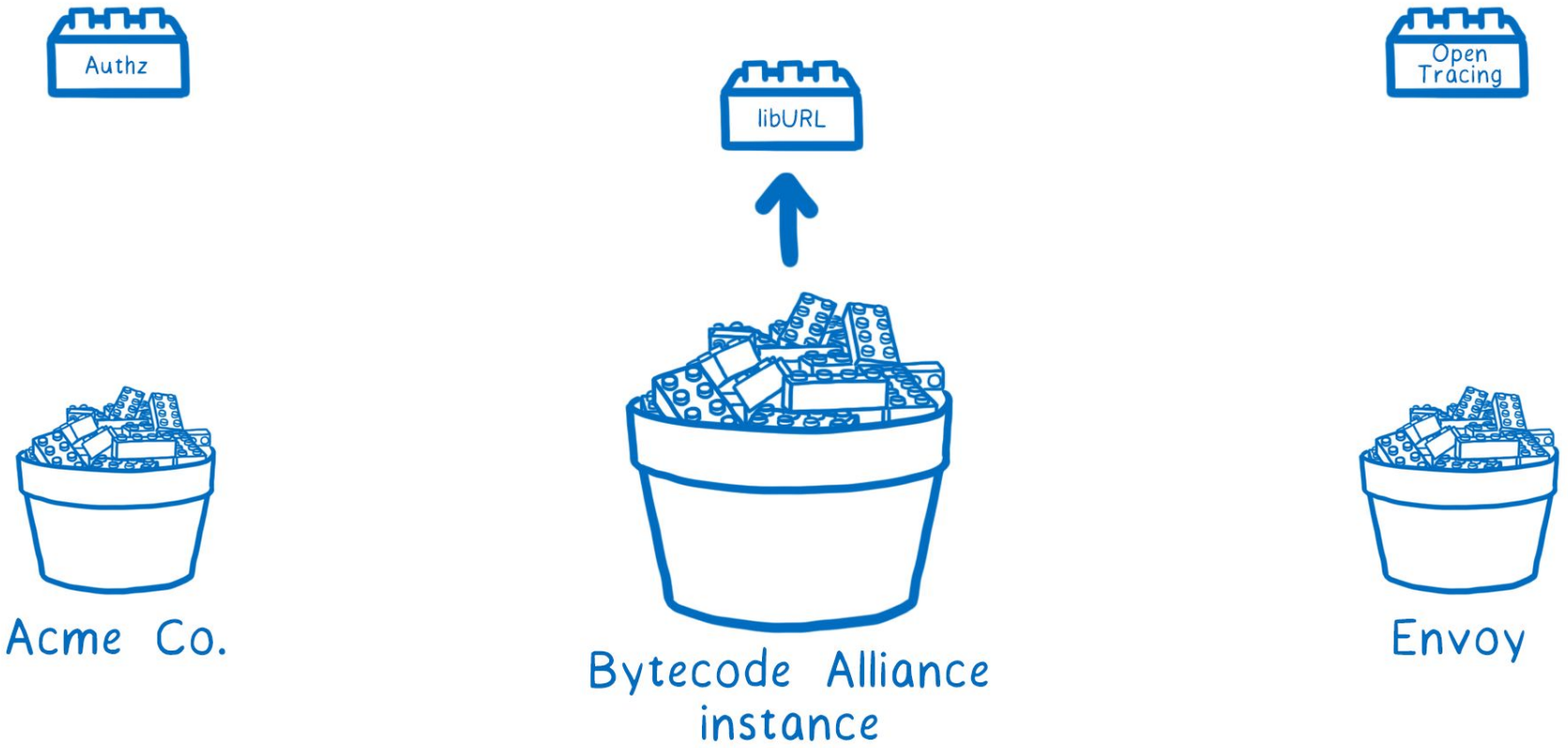


image credit: Lin Clark

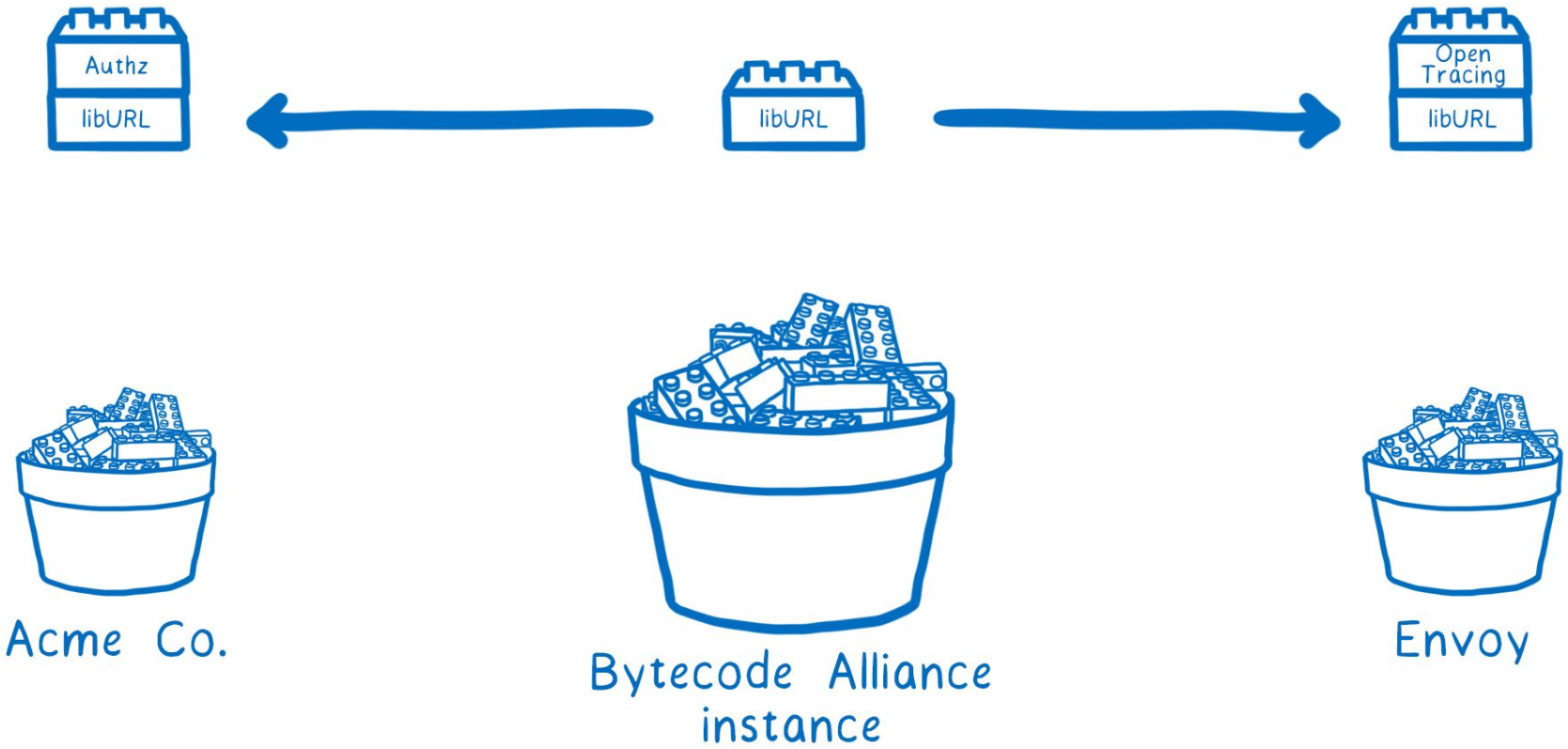


image credit: Lin Clark

Acme Co.

Bytecode Alliance
instance

Envoy

image credit: Lin Clark

image credit: Lin Clark

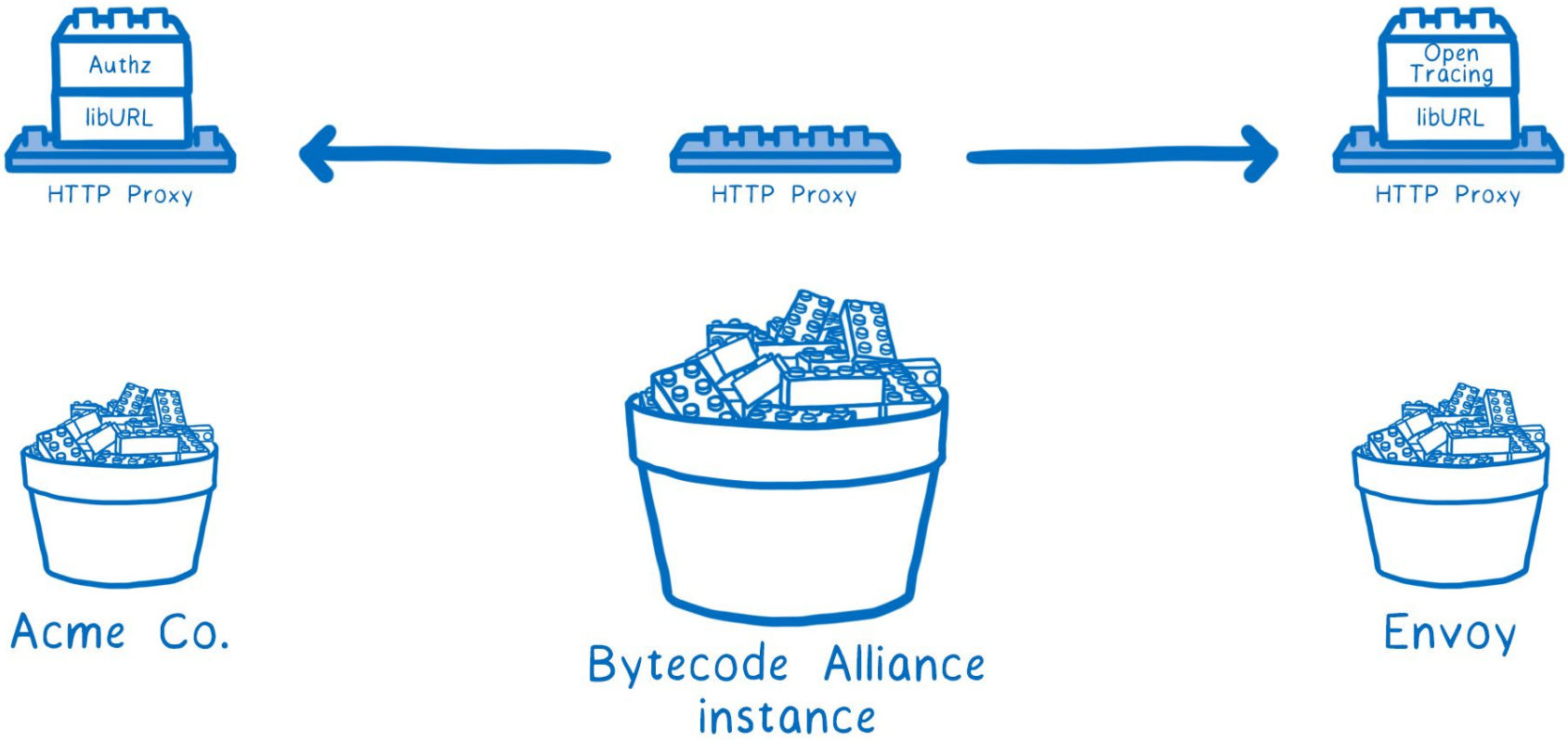


image credit: Lin Clark

HTTP Proxy

HTTP Proxy

Acme Co.

Bytecode Alliance
instance

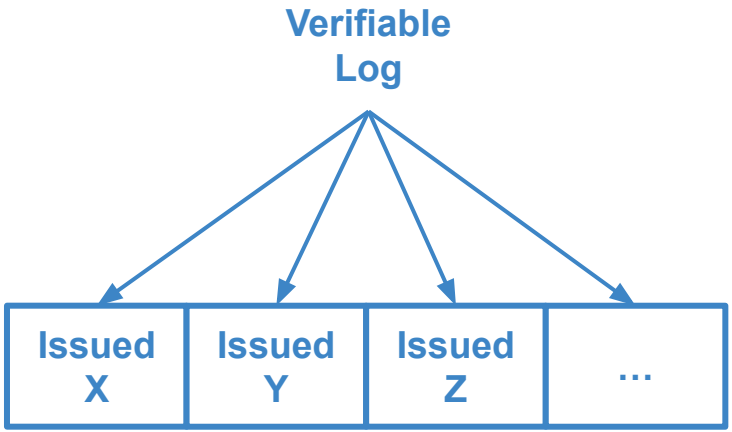
Envoy

image credit: Lin Clark

image credit: Lin Clark

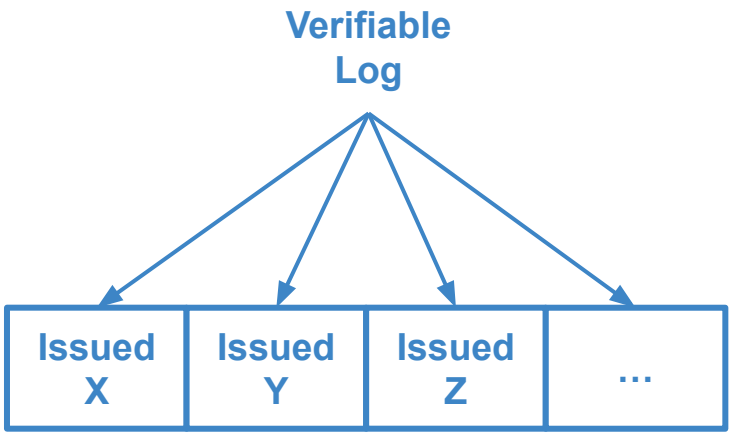
Ideas we can borrow from Certificate & Revocation Transparency

Certificate Transparency

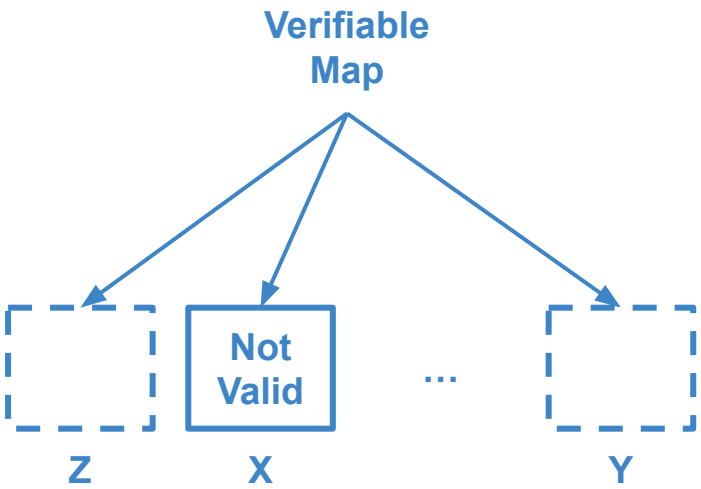


(Sequential entries)

Certificate Transparency with Revocation

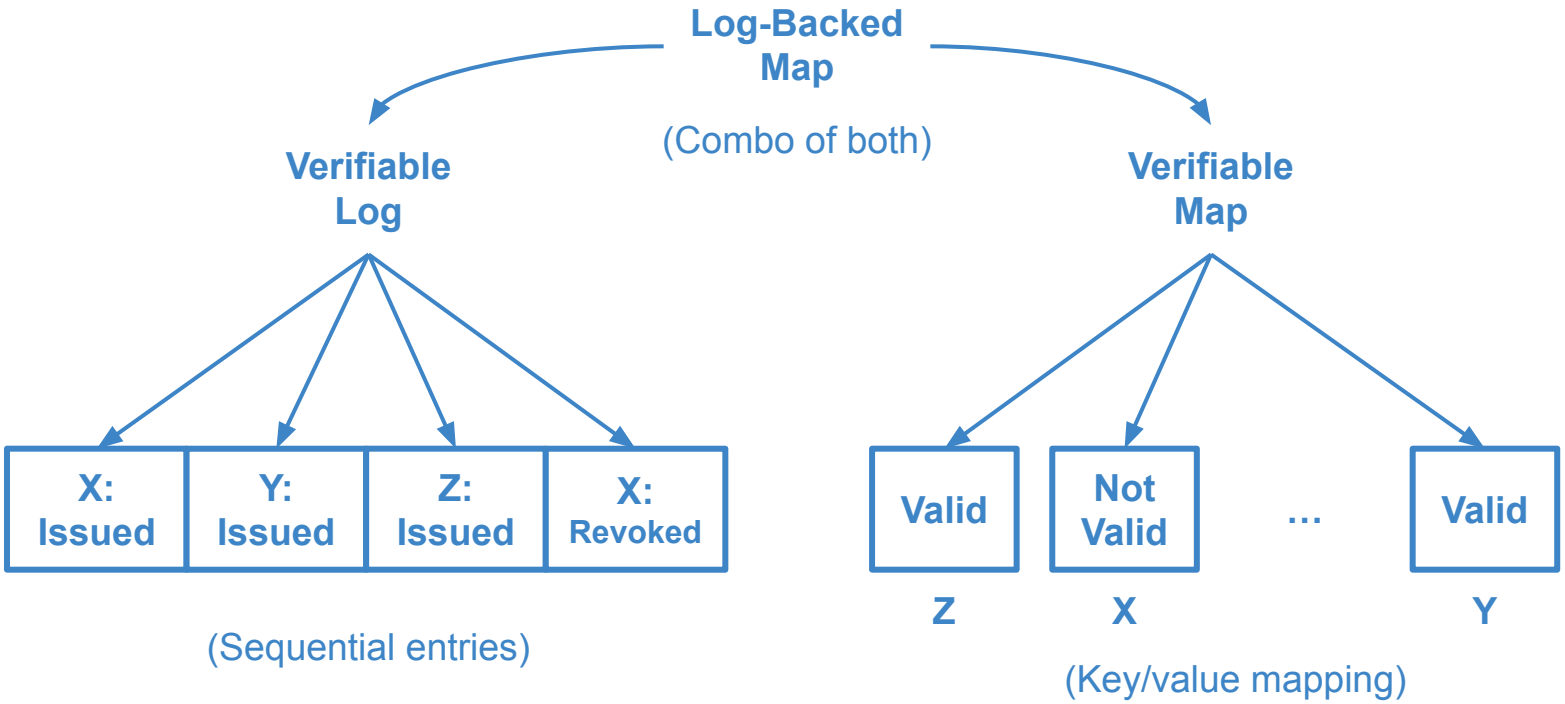


(Sequential entries)



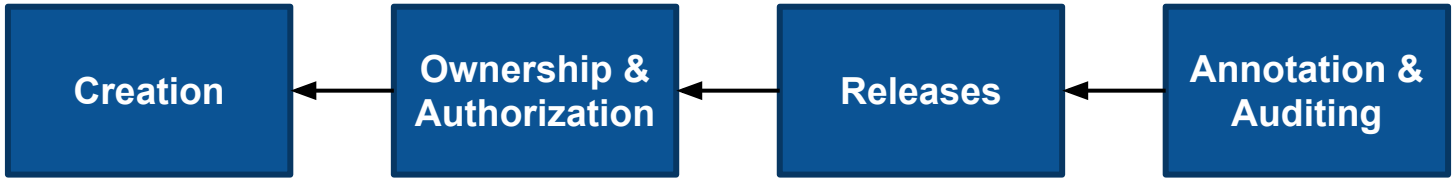
(Key/value mapping)

Certificate Transparency with Revocation

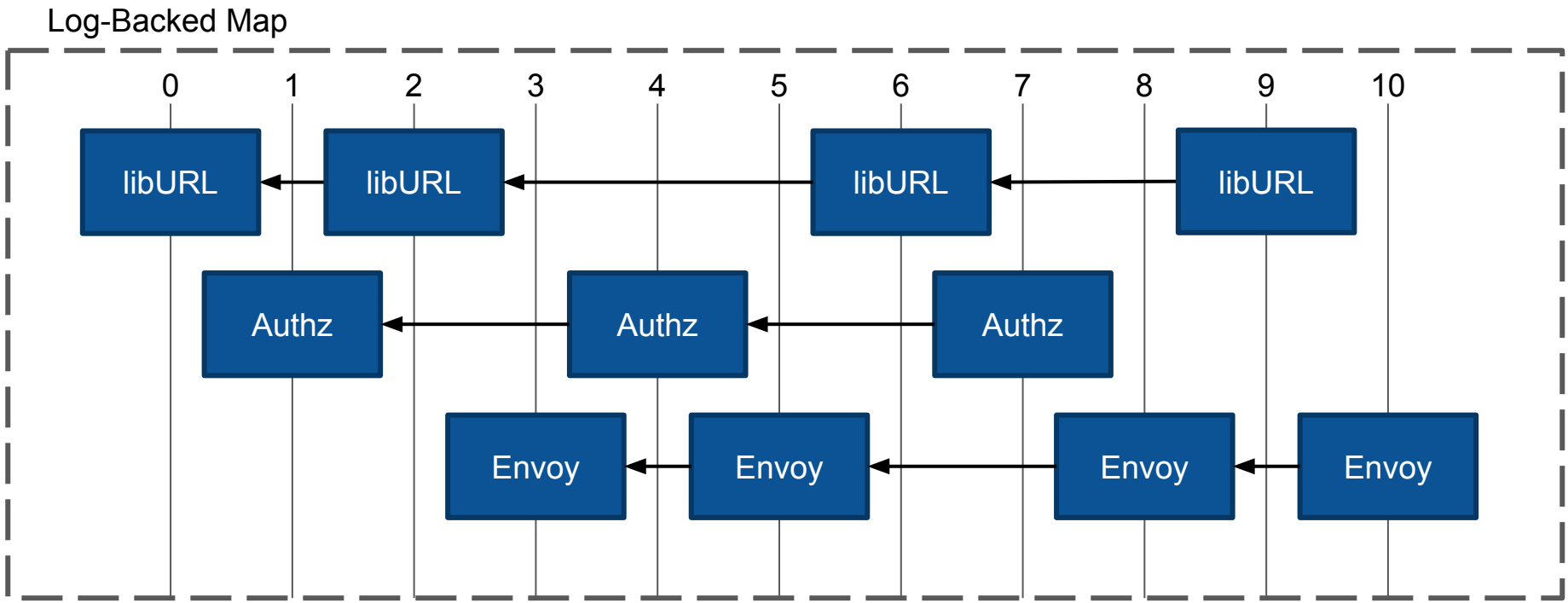


How can we make a registry as transparent as a CA?

Package Logs



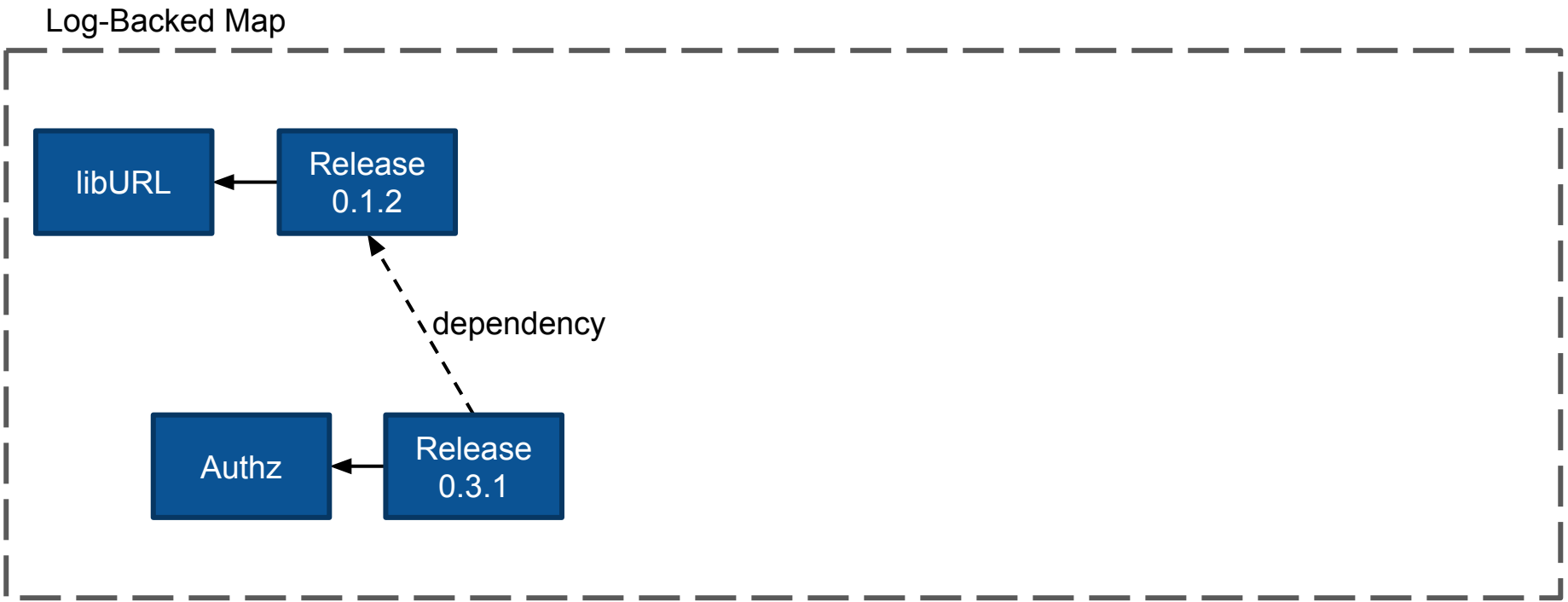
Registry Structure



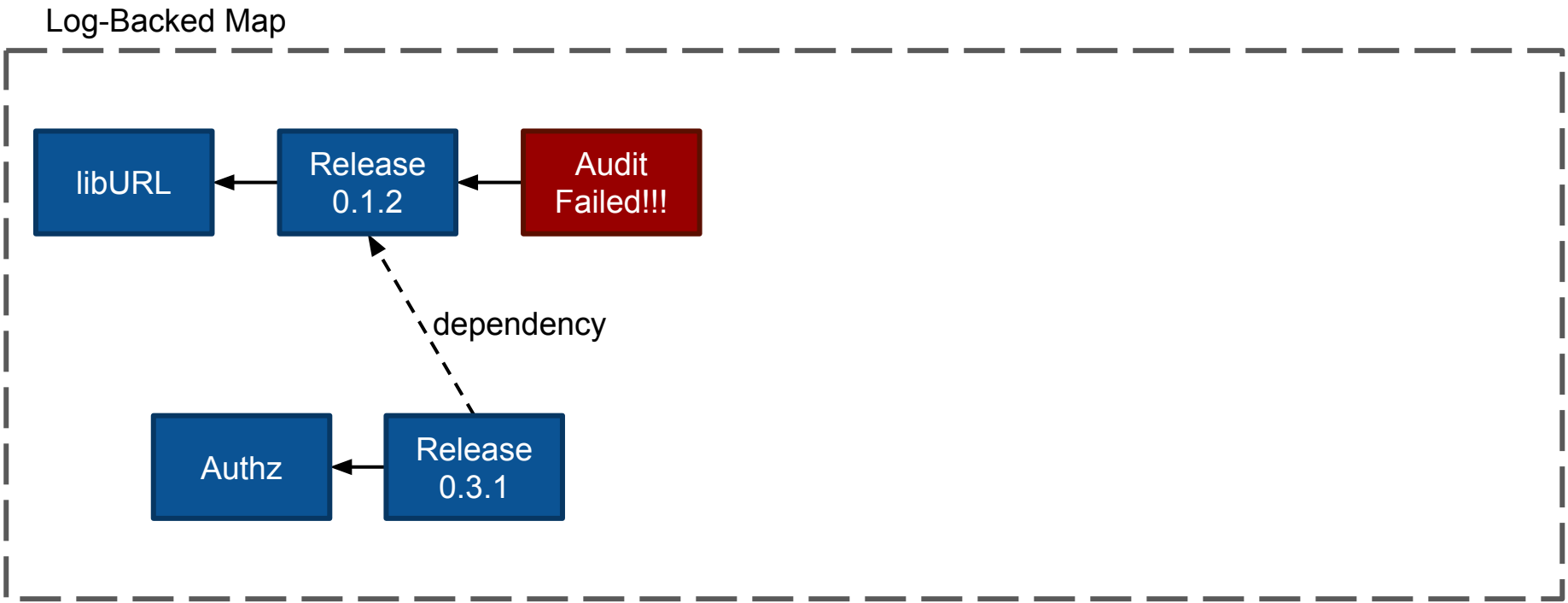
Transparency Helps Manage Vulnerabilities

libURL Releases 0.1.2

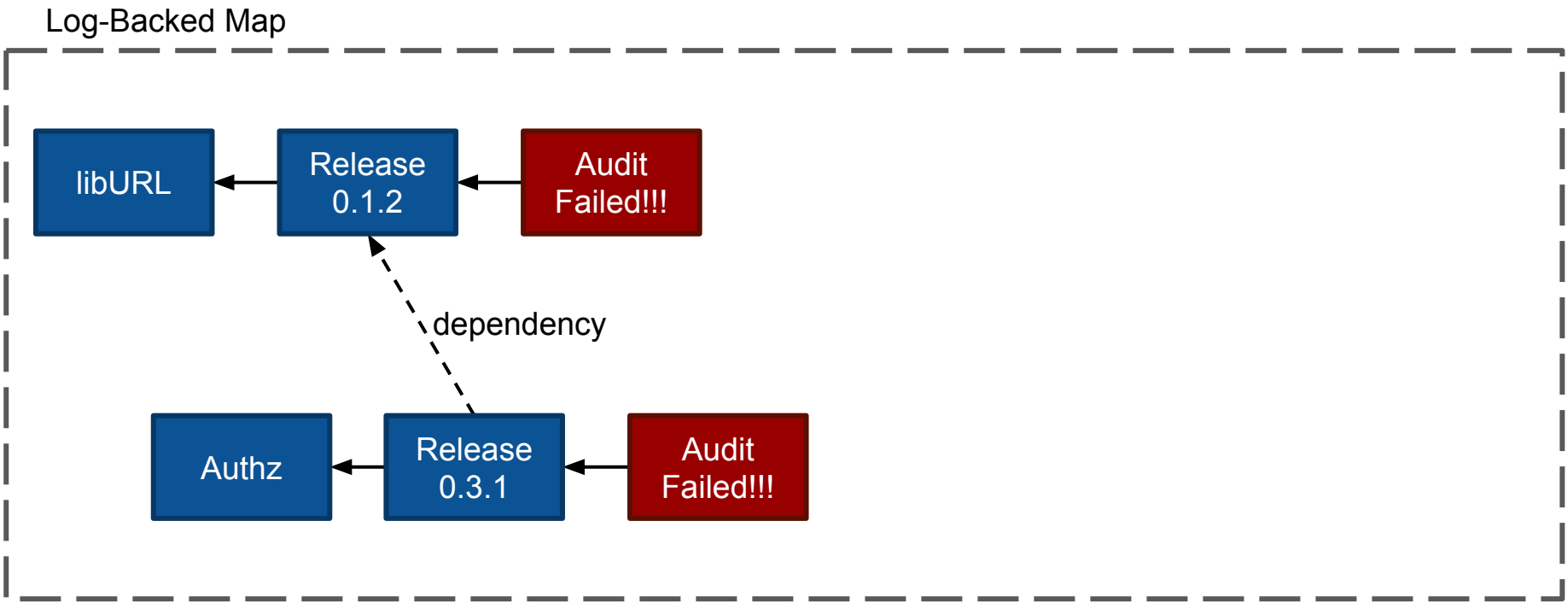
Authz Releases 0.3.1



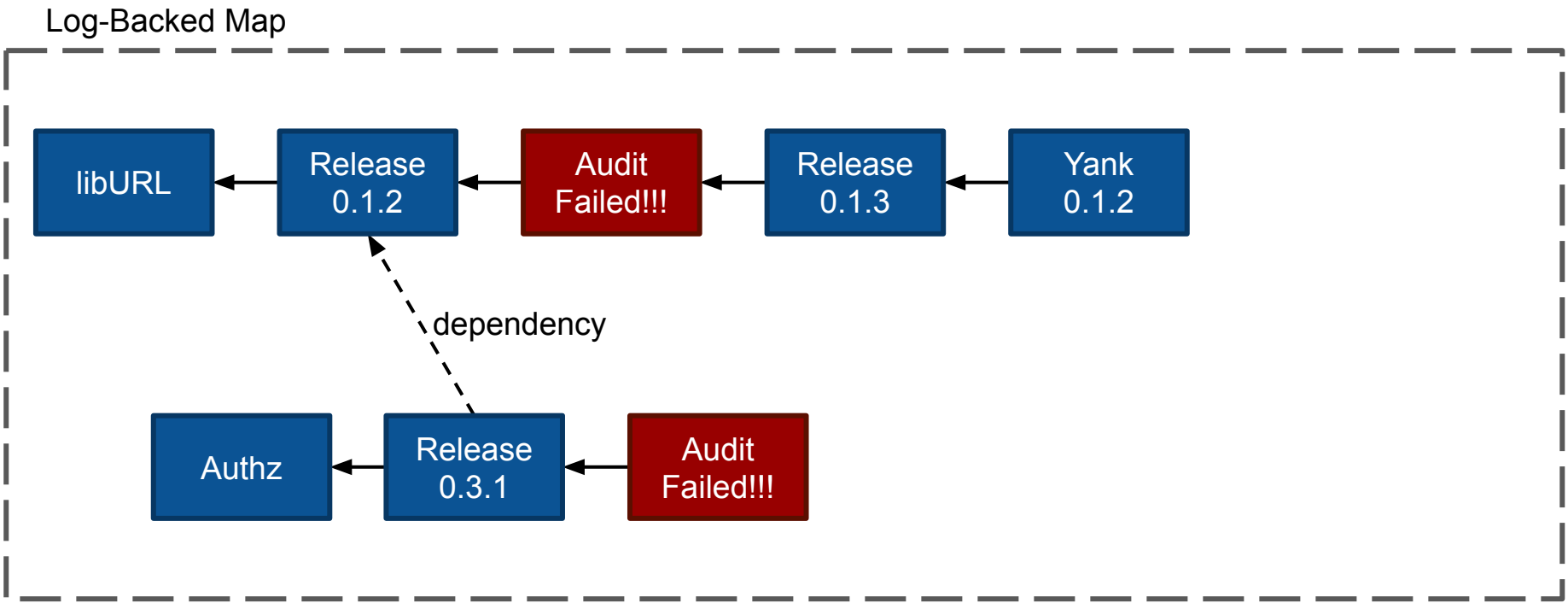
Vulnerability Found in libURL!!



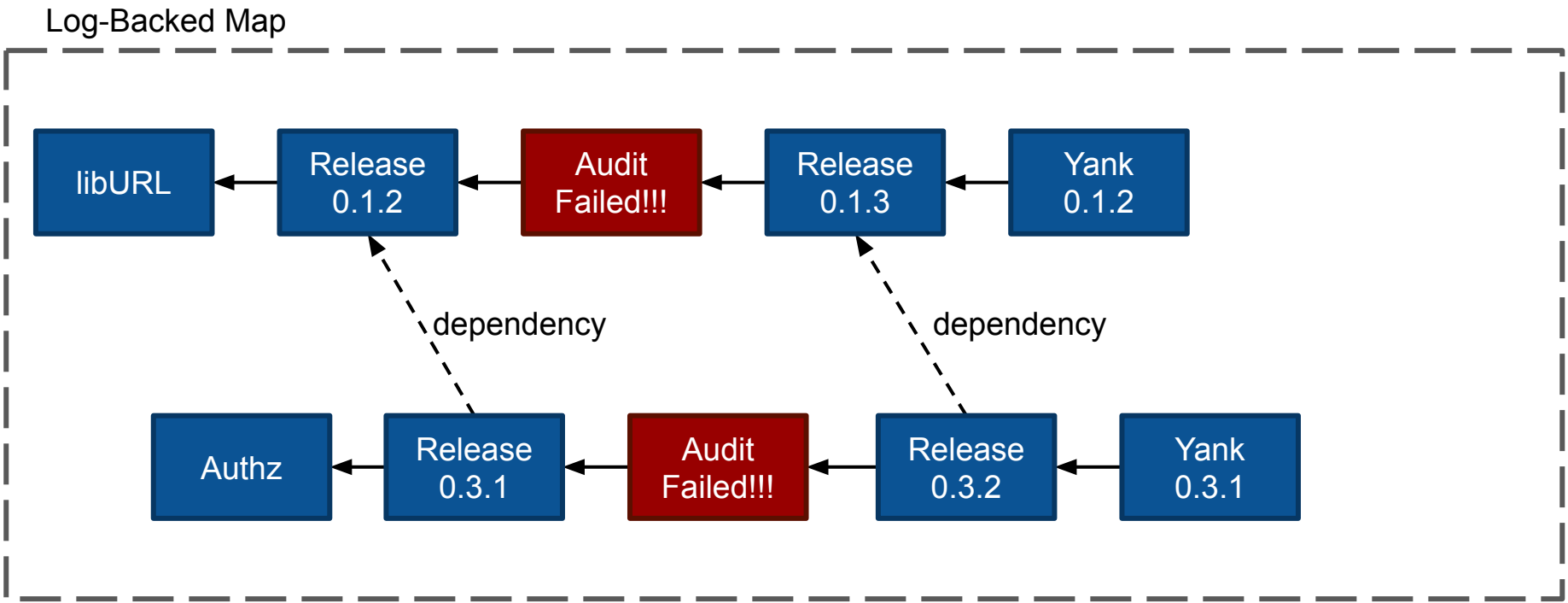
Vulnerability also in Authz!



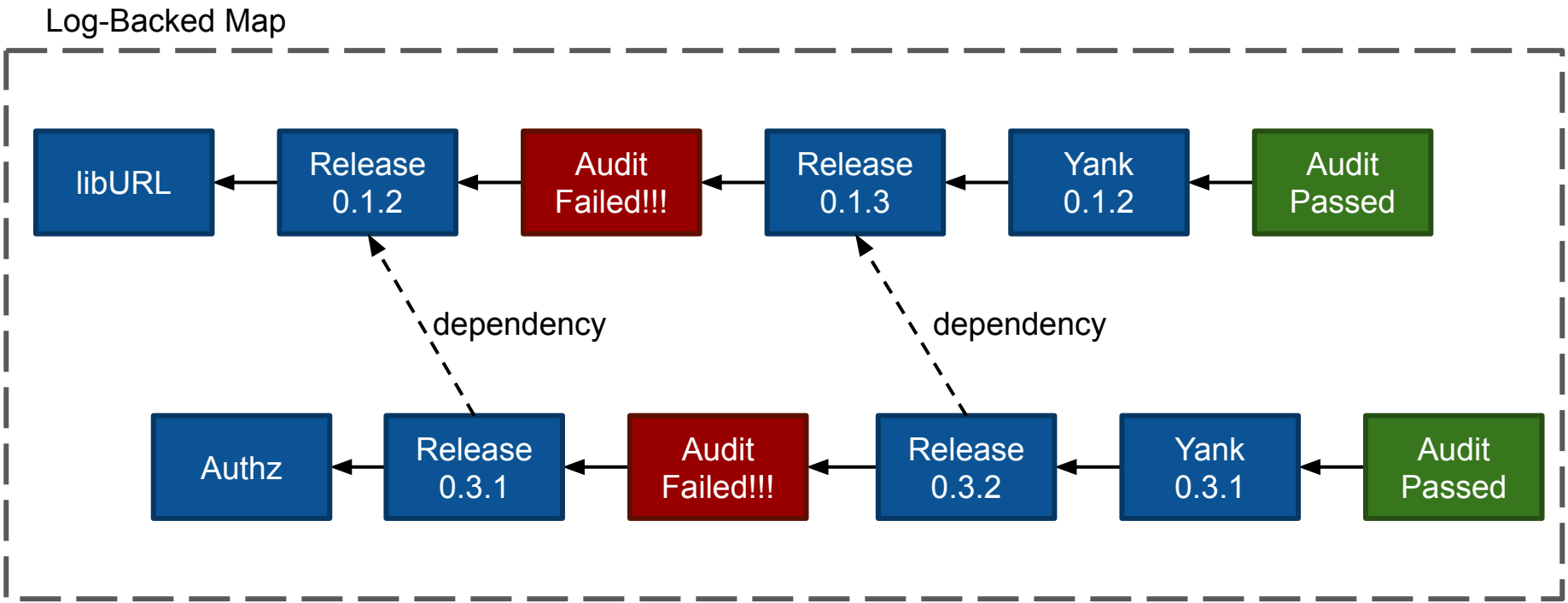
libURL Mitigates Vulnerability



Authz Mitigates Vulnerability

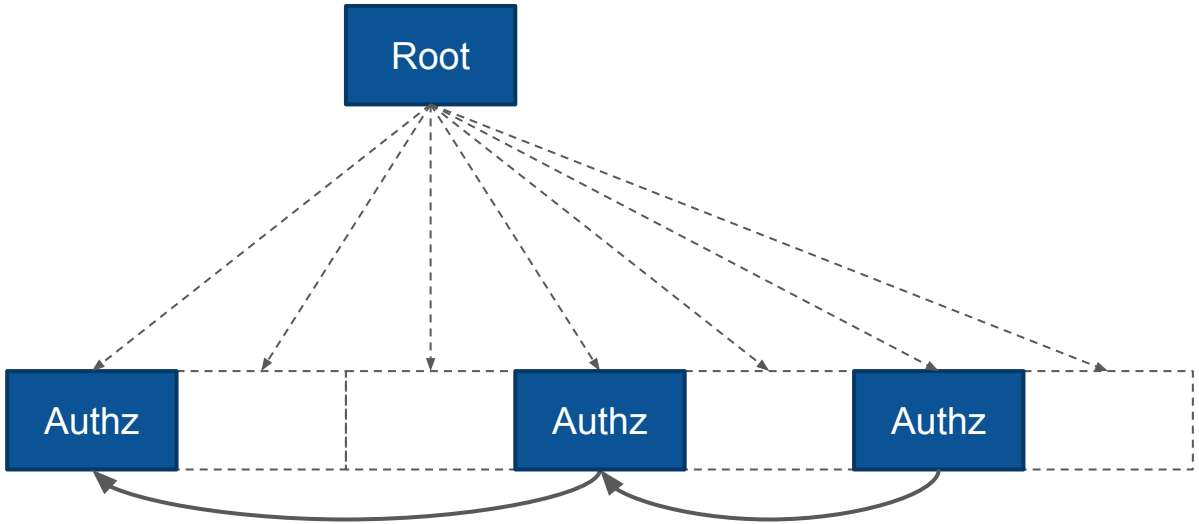


Both Pass their next Audits

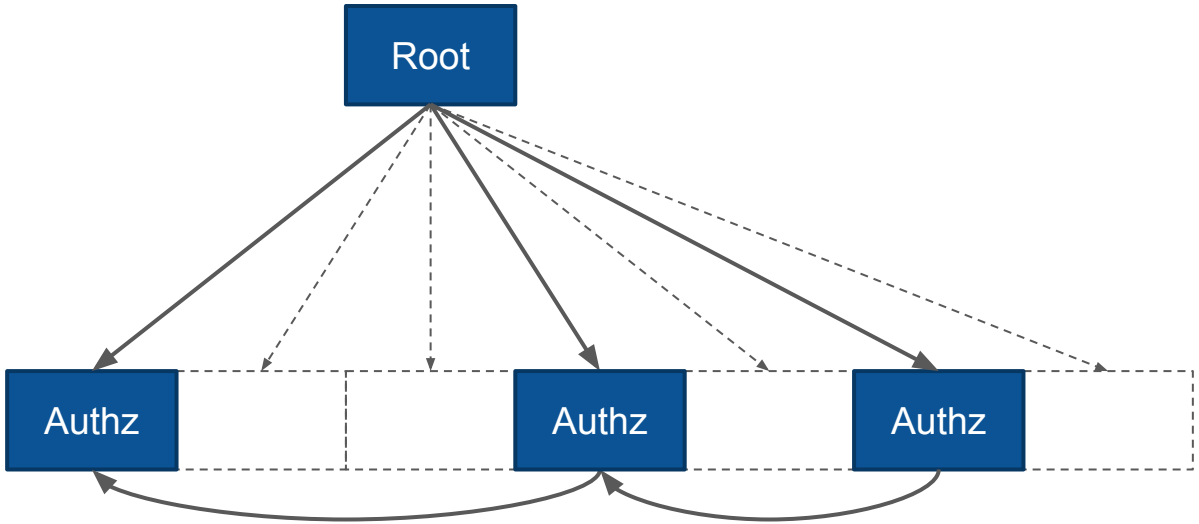


Transparency Helps Detect Operator Compromise

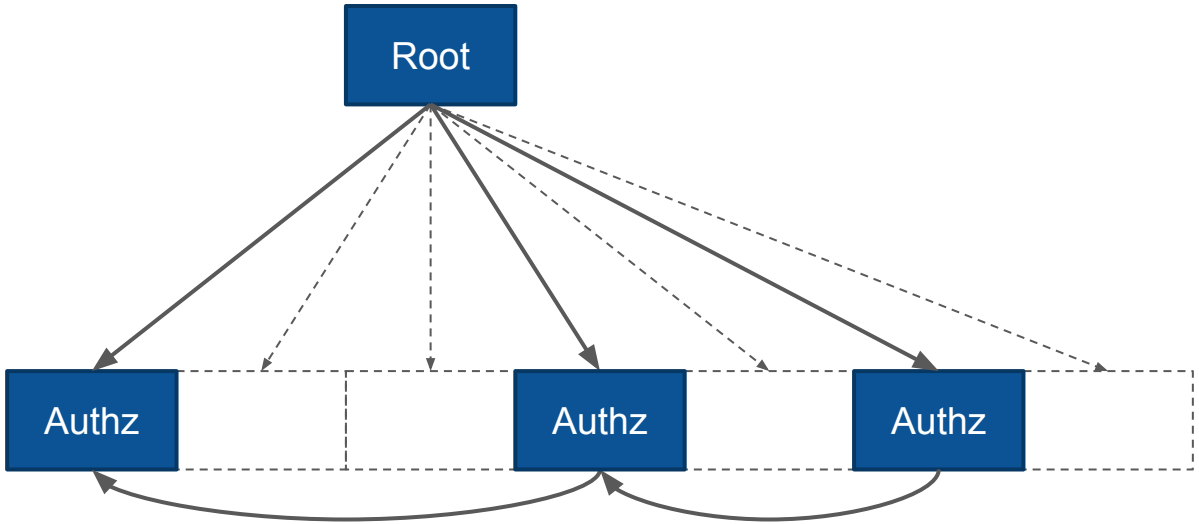
User Downloads Authz Package



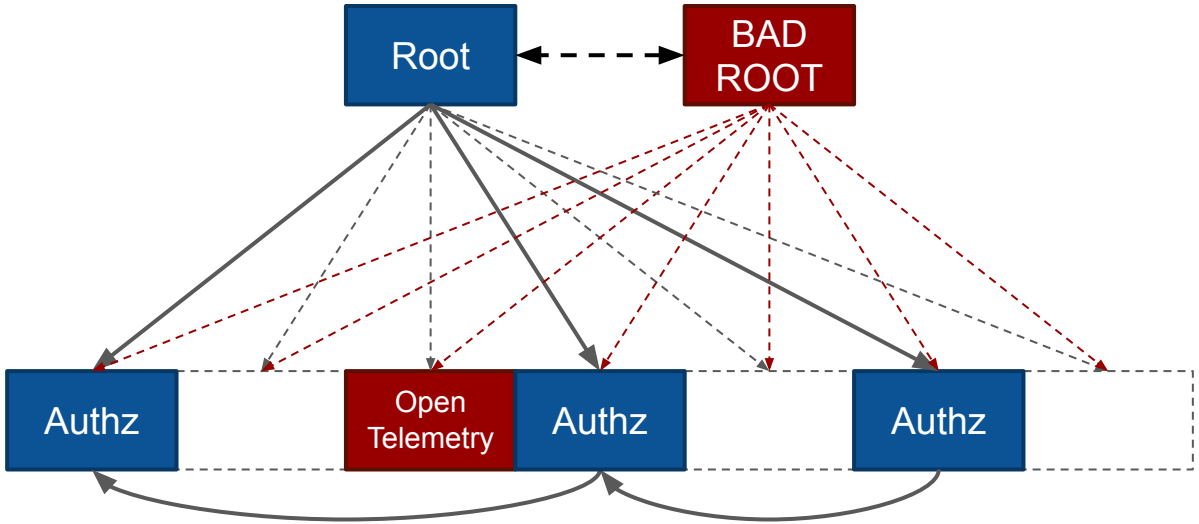
User Verifies Authz Package



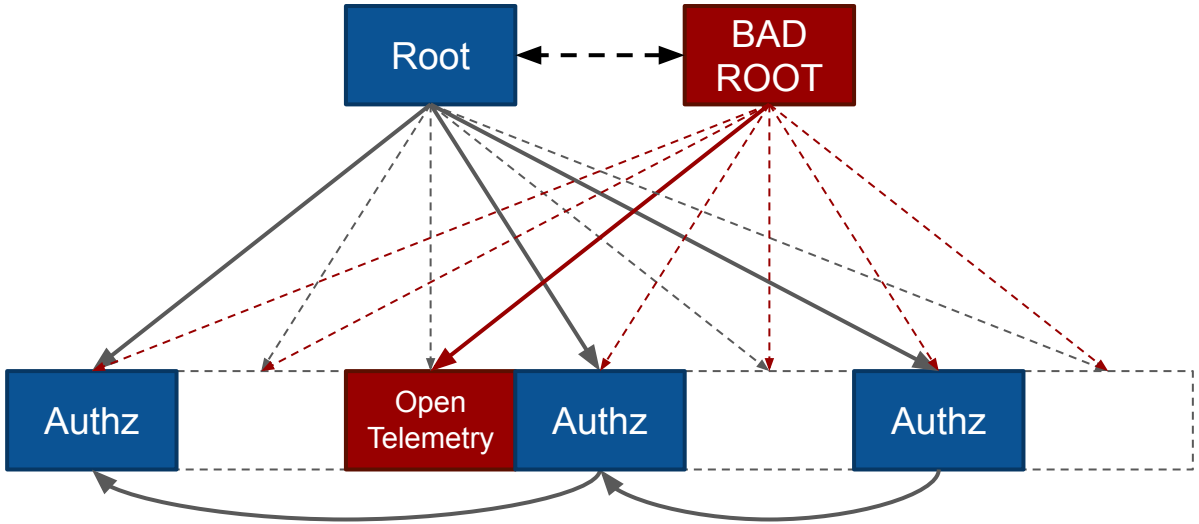
Registry Operator Compromised!



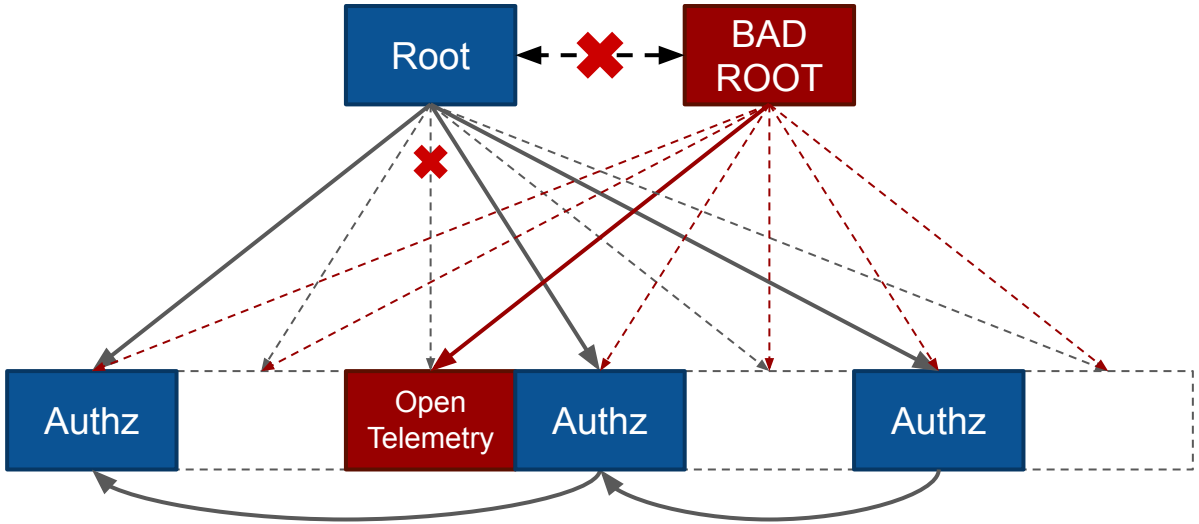
User Downloads OpenTelemetry Package



User Verifies OpenTelemetry Package



User Detects Conflict



Harbor

Bindle

OCI

Trillian

Rekor

TUF

sigstore

crates.io

certificate-transparency

NPM

Bindle

Harbor

OCI

Trillian

Rekor

TUF

sigstore

crates.io

certificate-transparency

NPM

Harbor

Bindle

OCI

Trillian

Rekor

TUF

sigstore

crates.io

certificate-transparency

NPM

Component Model

Federation & Interoperability

Package Transparency

Acknowledgements

- Collaboration with Lann Martin & Nathaniel McCallum & Luke Wagner
- Art from Lin Clark
- SIG-Registry group