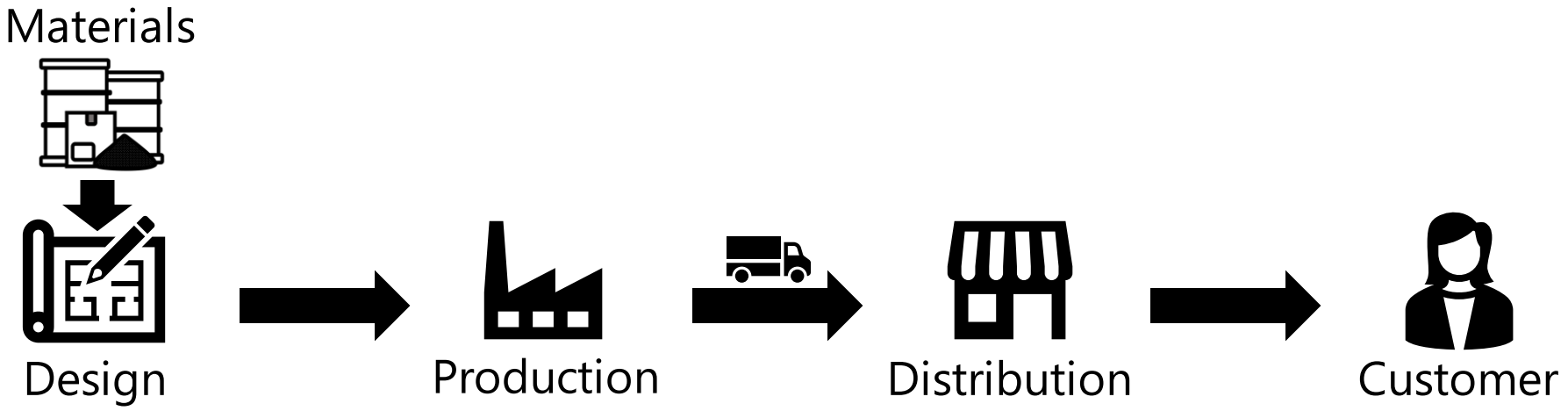
Trustworthy Digital Supply Chain Transparency Services

Henk Birkholz <henk.birkholz@sit.fraunhofer.de>
on behalf of SCITT contributors (W. Bartholomew, H. Birkholz, S. Clebsch, A. Deligat-Lavaud, Y. Deshpande, C. Fournet, B. Knight, S. Lasker, R. Martin, S. Provine, M. Riechert, A. Stewart, K. Williams, R. Williams, … see scitt@ietf.org)

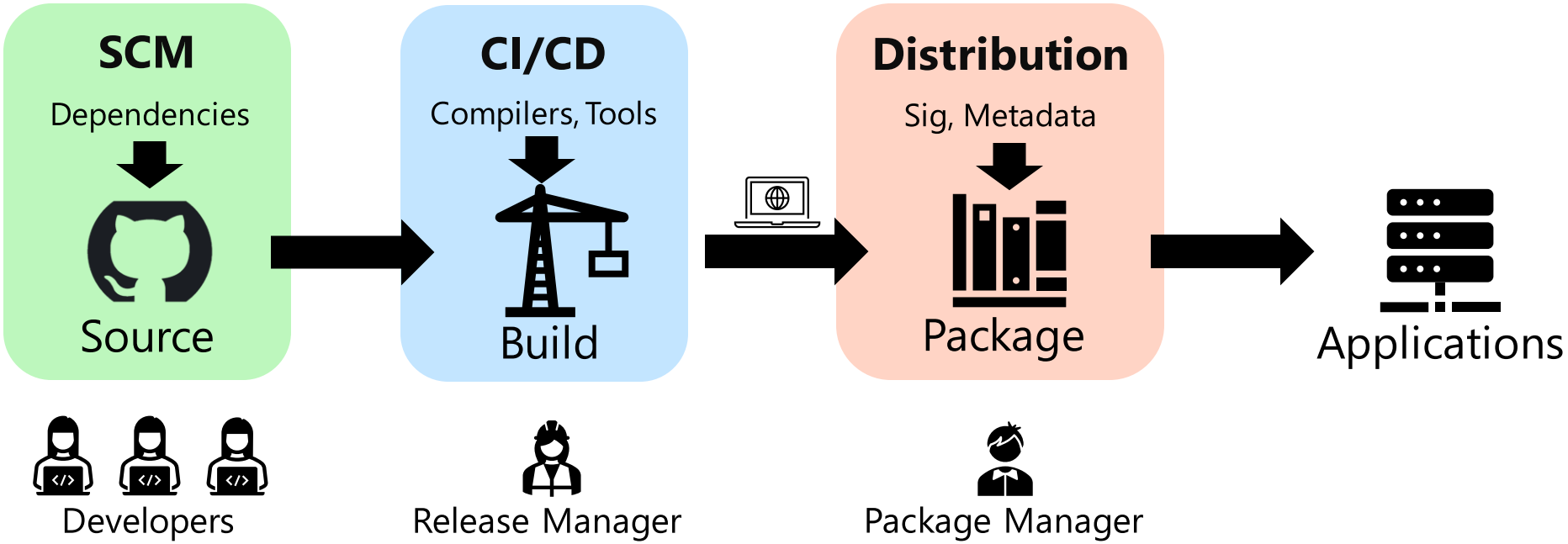
SECDISPATCH @ IETF 113, Tue March 22nd, 2022

Supply Chains

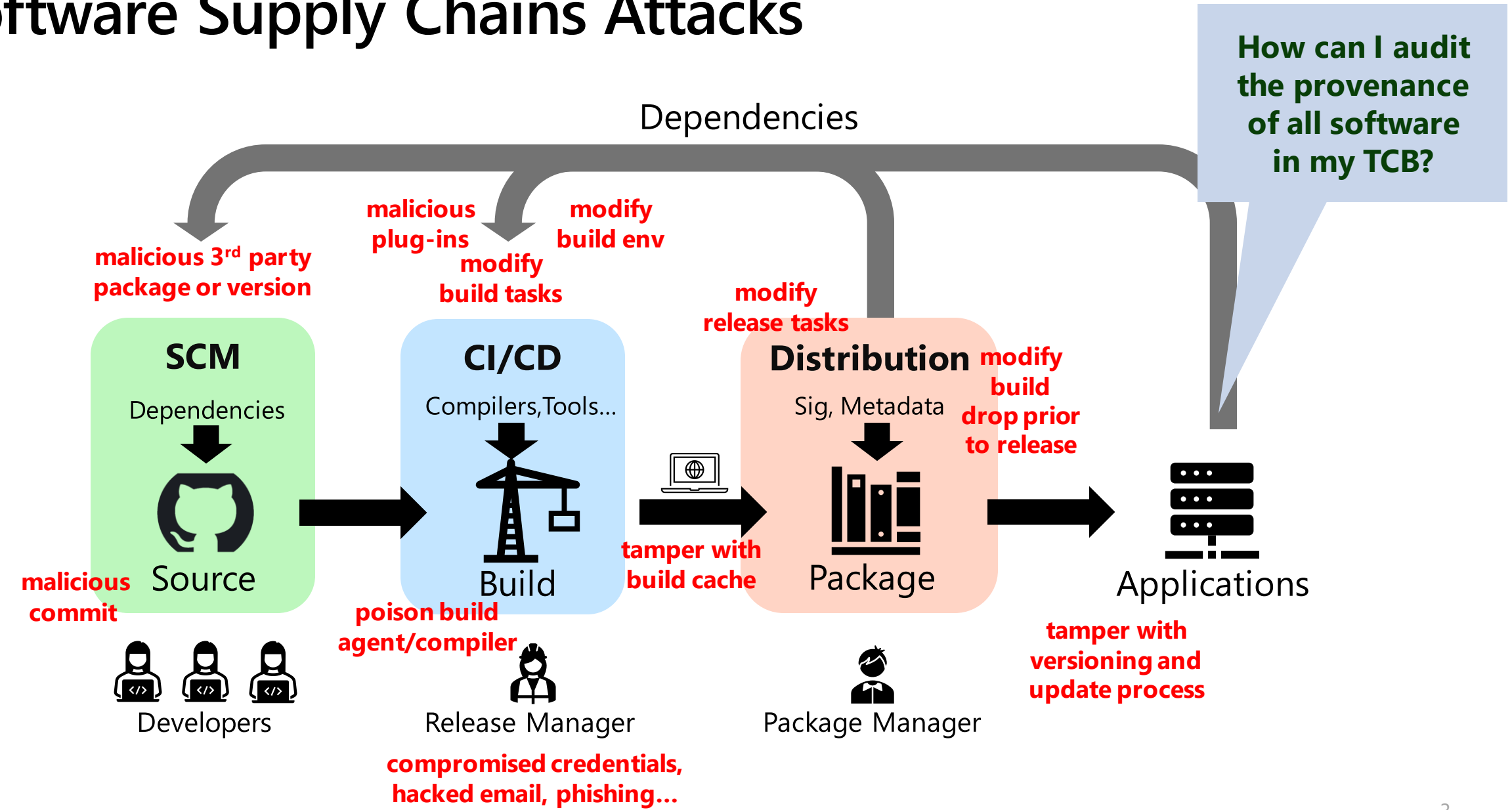
Generic Supply Chain



Software Supply Chain



Software Supply Chains Attacks



Software Supply Chains Attacks

ars TECHNICA — BIZ & IT | TECH | SCIENCE | POLICY | CARS | GAMING & CULTURE

BIZ & IT —

Devs unknowingly use "malicious" modules snuck into official Python repository

BIG HACK ATTACK —

The year-long rash of supply chain attacks against open source is getting worse

Backdoors snuck into 12 OSS packages were downloaded hundreds of thousands of times.

DAN GOODIN - 8/21/2019, 12:35 PM

BIZ & IT —

Rage-quit: Coder unpublished 17 lines of JavaScript and "broke the Internet"

Dispute over module name in npm registry became giant headache for developers.

SEAN GALLAGHER - 3/25/2016, 2:10 AM

POISONING THE WELL —

Two new supply-chain attacks come to light in less than a week

As drive-by attacks get harder, hackers exploit the trust we have in software providers.

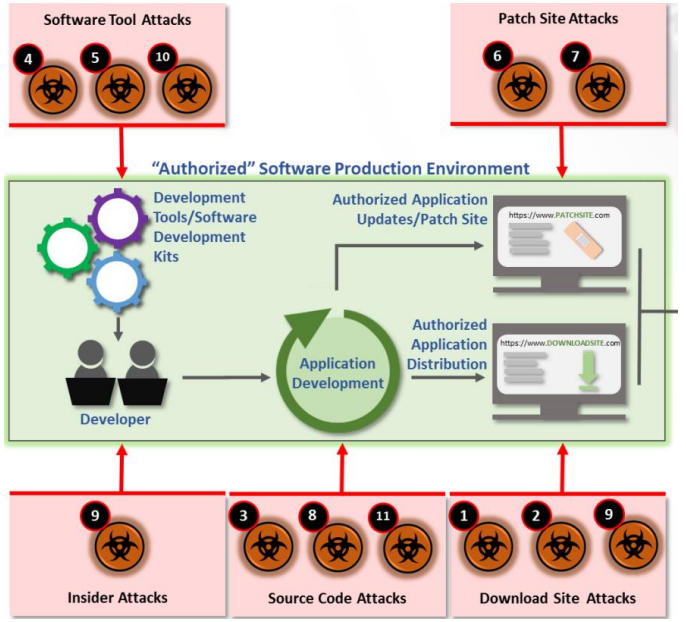
DAN GOODIN - 10/23/2018, 10:45 PM

POISONING THE WELL —

Widely used open source software contained bitcoin-stealing backdoor

Malicious code that crept into event-stream JavaScript library went undetected for weeks.

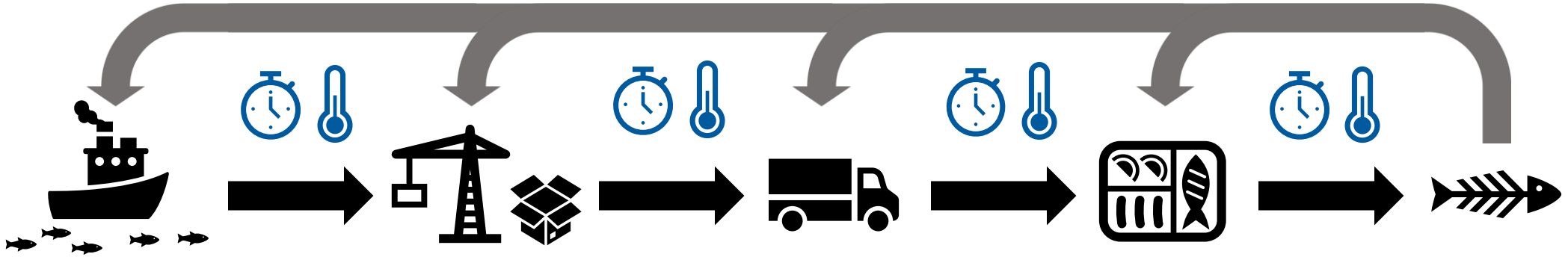
DAN GOODIN - 11/26/2018, 10:55 PM



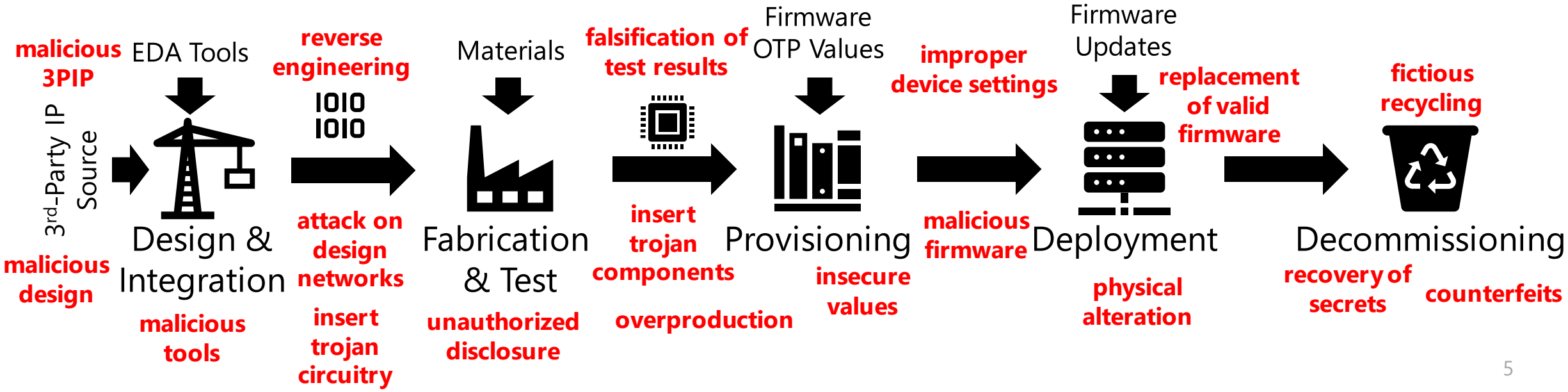
Legend	Date	Attack Name	Target Technology	Attack Vector	Attack Note
1	Jun 2014	Havex / Dragonfly	Industrial Control Systems	Download Site Attack	Watering hole attack
2	Apr 2015	KingSlayer	Network Logs and Event Monitor Tools	Download Site Attack	Subversion at distribution point by redirecting download request to malicious actor site
3	Dec 2015	Juniper Network Attack	Network Equipment Source Code	Source Code Attack	Unauthorized code added which created authentication bypass and ability to monitor and decrypt VPN traffic
4	Dec 2015	XcodeGhost	iOS	Software Development Tool Attack	Fake version of the developer tool distributed to site frequented by developers
5	Jan 2017	Expensive Wall / Shady SDK	Android	Software Development Tool Attack	Obfuscation used by malware developers to encrypt malicious code, allowing evasion of anti-malware protections
6	Jun 2017	Un-Named Attack	Python	Patch Site Attack	Typosquatting attack
7	Jun 2017	NotPetya	MeDoc	Patch Site Attack	Software infrastructure compromise to tamper with code
8	Jul 2017	Shadowpad	Network Manage Software Suite	Source Code Attack	Backdoor injected into a network management software suite then pushed through software update
9	Aug 2017	Floxif	CCleaner	Insider/Download Site Attack	Infiltration into development or distribution process before cryptographic signature for software occurred
10	Aug 2017	HackTask	JavaScript	Software Development Tool Attack	Typosquatting attack
11	October 2017	Un-Named North Korea Attack	Anti-Virus Code	Source Code Attack	Infiltrated network of a company providing computer anti-virus service

Source: NIST CSRC

Cold Chains for Seafood



Microelectronics Supply Chain



Auditing Supply Chains

- EO14028
- SWID
- SPDX
- In-toto
- SLSA
- CycloneDX

Software BOM

Single pane Window 500mm x 500mm						
BoM level	Part #	Description	Qty	Units	Unit Cost	Cost
1	756	Window framing	1	4	$3.00	$12.00
1	95	Brackets	1	4	$0.75	$3.00
1	PRS045	Rubber seal	2	metre	$0.50	$1.00
2	342	Glass pane	1	1	$9.50	$9.50
2	LB8579	Safety label	1	1	$0.10	$0.10
3	GH098	Hinges	2	1	$2.25	$4.50
3	GS664	Screws	8	10	$4.95	$3.96
3	587	Latch	1	1	$2.20	$2.20
3	588	Latch hook	1	1	$0.88	$0.88
4	GS660	Screws for latch and hook	6	10	$4.95	$2.97
5	812	Protective wrap	1.5	metre	$0.65	$0.98
6	XYZ123	Cardboard box 600mm x 600mm	1	1	$1.00	$1.00
6	LB7487	Box label barcode	1	1	$0.10	$0.10
		Total number parts	27.5		Total costs	$42.19

Bill of Materials

In-toto statement

SLSA compliance levels

		Required at			
Requirement		SLSA 1	SLSA 2	SLSA 3	SLSA 4
Source	Version Controlled		✓	✓	✓
	Verified History			✓	✓
	Retained Indefinitely			18 mo.	✓
	Two-Person Reviewed				✓
Build	Scripted	✓	✓	✓	✓
	Build Service		✓	✓	✓
	Ephemeral Environment			✓	✓
	Isolated			✓	✓
	Parameterless				✓
	Hermetic				✓
	Reproducible				○
Provenance	Available	✓	✓	✓	✓
	Authenticated		✓	✓	✓
	Service Generated		✓	✓	✓
	Non-Falsifiable			✓	✓
	Dependencies Complete				✓
Common	Security				✓
	Access				✓
	Superusers				✓

○ = *required unless there is a justification*

VERIFICATION OF COMPLIANCE

Attestation Number : CRXZ181204007-02
Date of Issue: 2018-12-04
Product: Block Camera
Model(s): BC-80
Brand: datavideo
Manufacturer & Address: Datavideo Technologies Co., Ltd
10F., No.176, Jian 1st Rd., Chung Ho District, New Taipei City 235, Taiwan

Bay Area Compliance Laboratories Corp. (Taiwan) hereby declares that the submitted sample(s) of the above equipment has been tested for CE-marking and in accordance with the following European Directives and Standards:

EMC Directive 2014/30/EU

Harmonized Standards	Test Report Number
EN55032:2015/AC: 2016-07	RTWA170508001-01-M1
EN55024:2010/A1: 2015	
EN61000-3-2: 2014	
EN61000-3-3: 2013	

* Note: Harmonized Standards not yet cited in OJ

CE Mark is permitted only after all applicable requirements are met in accordance with the European Union Rules, including the manufacturer's issuance of a "Declaration of Conformity. The Declaration of Conformity is issued under sole responsibility of manufacturer. This attestation is specific to the standard(s) stated above and compliance with additional standards and/or European directives are applicable.

Attestation by: Jerry Chang
Lab Manager
Signature

Compliance Certificate

[Company Name]
[Company Slogan]
[web address]

[Stress Address]
[City, ST ZIP]
Phone: [000-000-0000]
Fax: [000-000-0000]

PACKING SLIP

DATE 1/26/2010
CUSTOMER ID [12345]

BILL TO:	SHIP TO:
[Name]	[Name]
[Company Name]	[Company Name]
[Stress Address]	[Stress Address]
[City, ST ZIP]	[City, ST ZIP]
[Phone]	[Phone]

ORDER DATE	ORDER #	PURCHASE ORDER #	CUSTOMER CONTACT
1/26/2010	[123456]	[123456]	Purchasing Dept.

ITEM #	DESCRIPTION	ORDER QTY	SHIP QTY
[23423423]	Product XYZ	15	13
[45645645]	Product ABC	1	1

Shipment records

Department of Commerce
The Minimum Elements for an SBOM

Data Field	Description
Supplier Name	The name of an entity that creates, defines, and identifies components.
Component Name	Designation assigned to a unit of software defined by the original supplier.
Version of the Component	Identifier used by the supplier to specify a change in software from a previously identified version.
Other Unique Identifiers	Other identifiers that are used to identify a component, or serve as a look-up key for relevant databases.
Dependency Relationship	Characterizing the relationship that an upstream component X is included in software Y.
Author of SBOM Data	The name of the entity that creates the SBOM data for this component.
Timestamp	Record of the date and time of the SBOM data assembly.

Transparency: Core Intuitions & Prior Work

We cannot stop supply chain actors from making false claims, but we can make them accountable by requiring their claims to be registered in verifiable **Transparency Ledgers**.

This ensures that malicious actors who make contradictory claims to different entities (customers, auditors, regulators) can be detected and held accountable.

All consumers of claims must first verify the proof of ledger registration to ensure a claim is auditable; this verification is cheap and can be done offline.

Examples of Transparency Systems

Certificate Transparency [RRC 6962] Adam Langley, Emilia Kasper, Ben Laurie (Google)

CONIKS: bringing key transparency to end users , M. S. Melara, A. Blankstein, J. Bonneau, E. W. Felten, and M. J. Freedman (USENIX Security'15).

Keeping authorities "honest or bust" based on large-scale decentralized witness cosigning (IEEE S&P '16)

CHAINIAC: Proactive Software-Update Transparency via Collectively Signed Skipchains and Verified Builds (Usenix'17, EPFL)

Contour: A practical system for binary transparency logging on bitcoin the latest authorized binary version.
M. Al-Bassam, S. Meiklejohn (Data Privacy Management, Cryptocurrencies and Blockchain Technology, 2018).

Transparency: Terminology

A **Ledger** is a consistent, append-only and distributed data store

A **Receipt** is a compact, offline verifiable cryptographic proof that a claim is stored in a ledger (example: Signed Cert Timestamp TLS extension in RFC 6962)

Claims are statements signed by **Issuers**, using keys they distribute through DID.
Transparent Claims are countersigned with receipts of ledger registration.

- Incorrect or inconsistent claims can be discovered by auditing the ledger

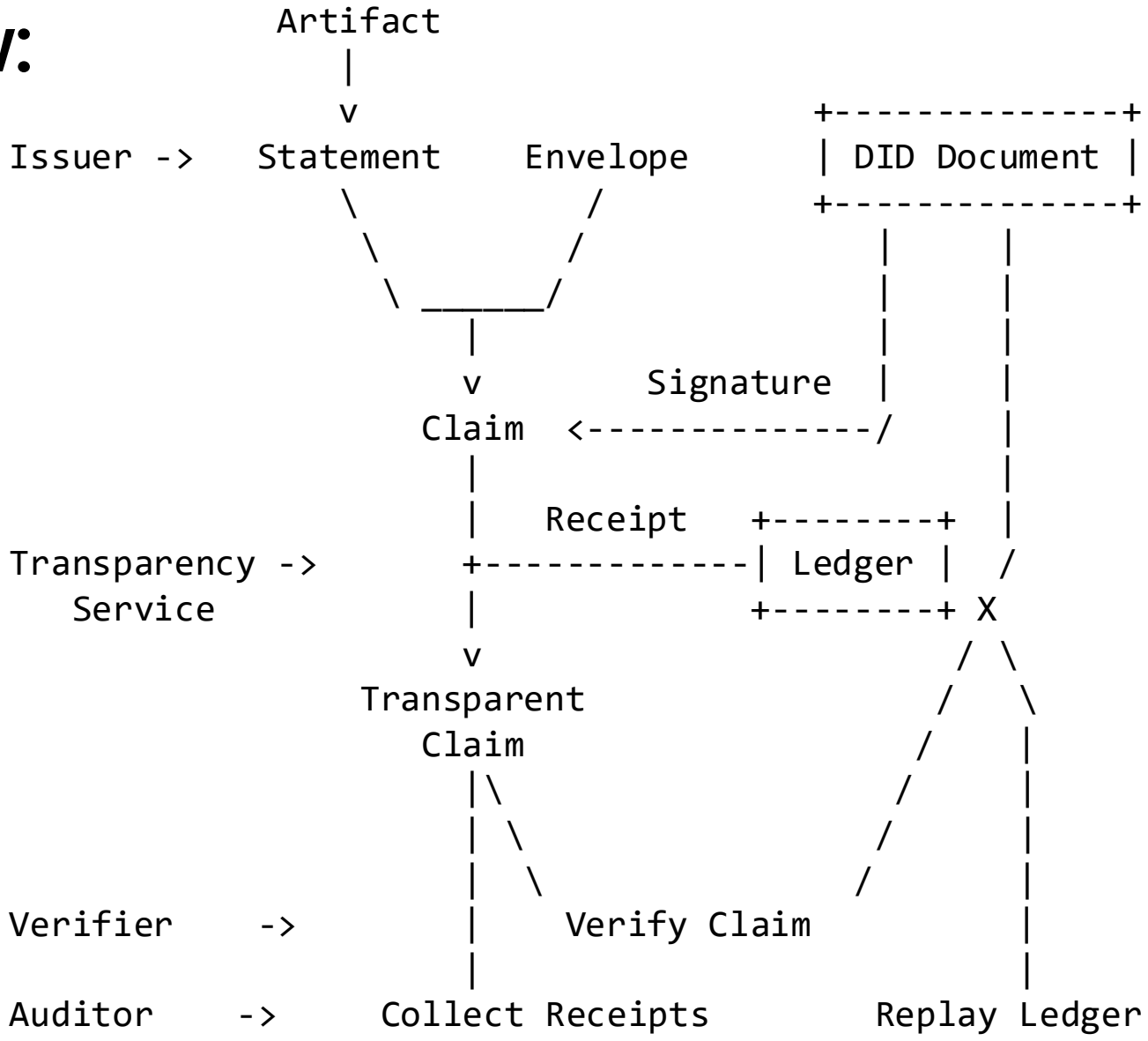
An artifact is **Transparent** if it comes with valid **Transparent Claims**.

- Relying parties verify claims before use and/or audit claims later
- Receipts prove the existence of claims in the ledger to offline 3rd parties
- Transparency is known to scale well in practice (e.g., CT: ~6B certs in ledger)

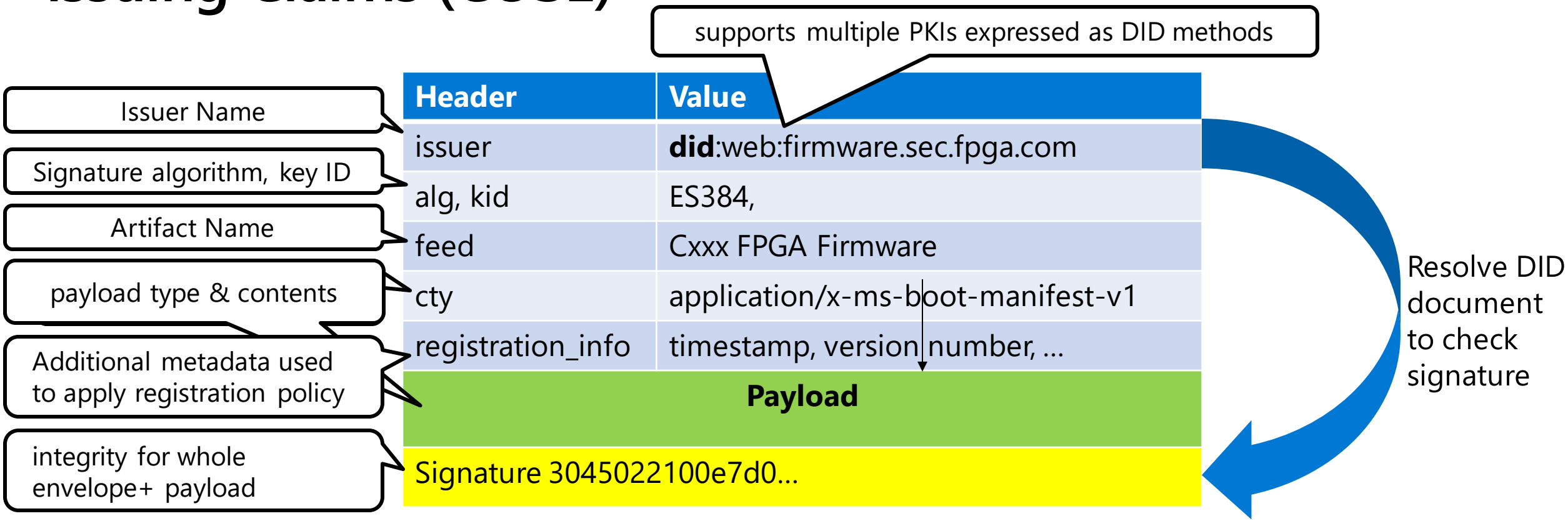
Architecture Overview:

Supply Chain Integrity, Transparency, and Trust

SCITT



Issuing Claims (COSE)



COSE as Universal Signing Envelope Format

- Standardized (RFC 7049, 8152)
- Efficient (for resource constrained devices)
- Direct payload encoding
- Extensible & crypto agile
- Not tied to X.509

```
COSE_Sign1 = [
    protected : bstr .cbor { * label => values },
    unprotected : { * label => values },
    payload : bstr / nil,
    signature : bstr
]
label = int / tstr
values = any
```

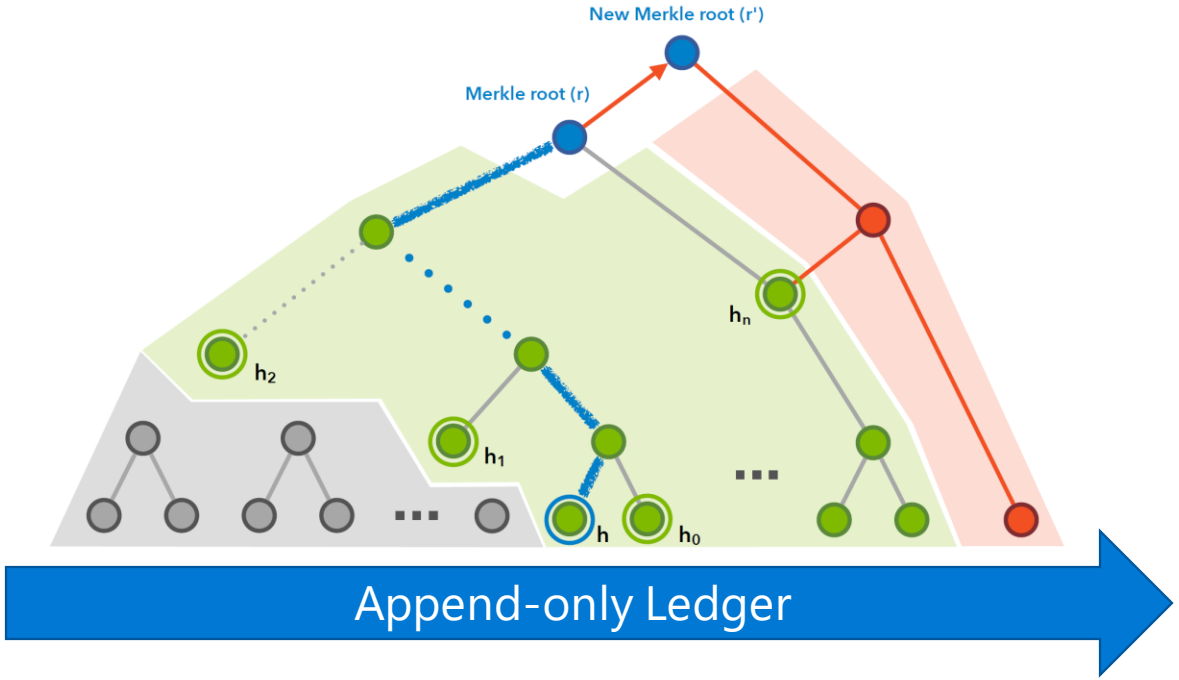
Countersigning Envelopes with Transparency Receipts

Receipts are implemented by signing the root of the binary Merkle Tree (root hash) over the whole ledger contents.

They can be issued efficiently:

- One hash per transaction
- One signature per transaction batch

The signing key is supported by attestation results and governance transactions, also recorded in the ledger.

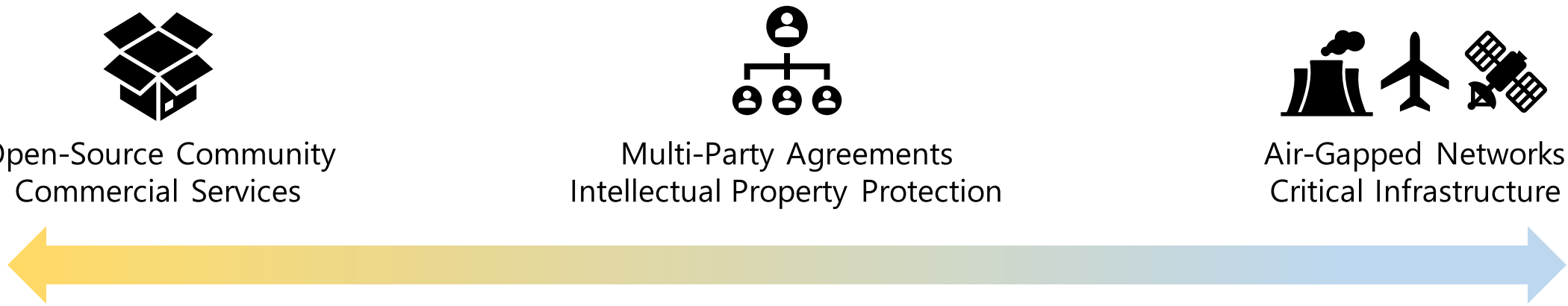



```
ReceiptContents = [
    signature: bstr             ; Signature over tree root
    node_certificate: tstr      ; Certificate of TS node that signed receipt
    proof: [+ ProofElement]     ; Intermediate hashes (Merkle path)
    leaf_info: LeafInfo         ; Extra data beyond claim stored in leaf
]
```

Federating Transparency Services

Multiple, independent transparency services can be governed and operated by different organizations

- Each transparency service enforces its own registration policy
- Relying parties may trust issuers and transparency services to different extents
- All Issuers, Transparency Services, Verifiers, and Auditors interoperate on claim envelopes and their interpretation using shared tools and formats



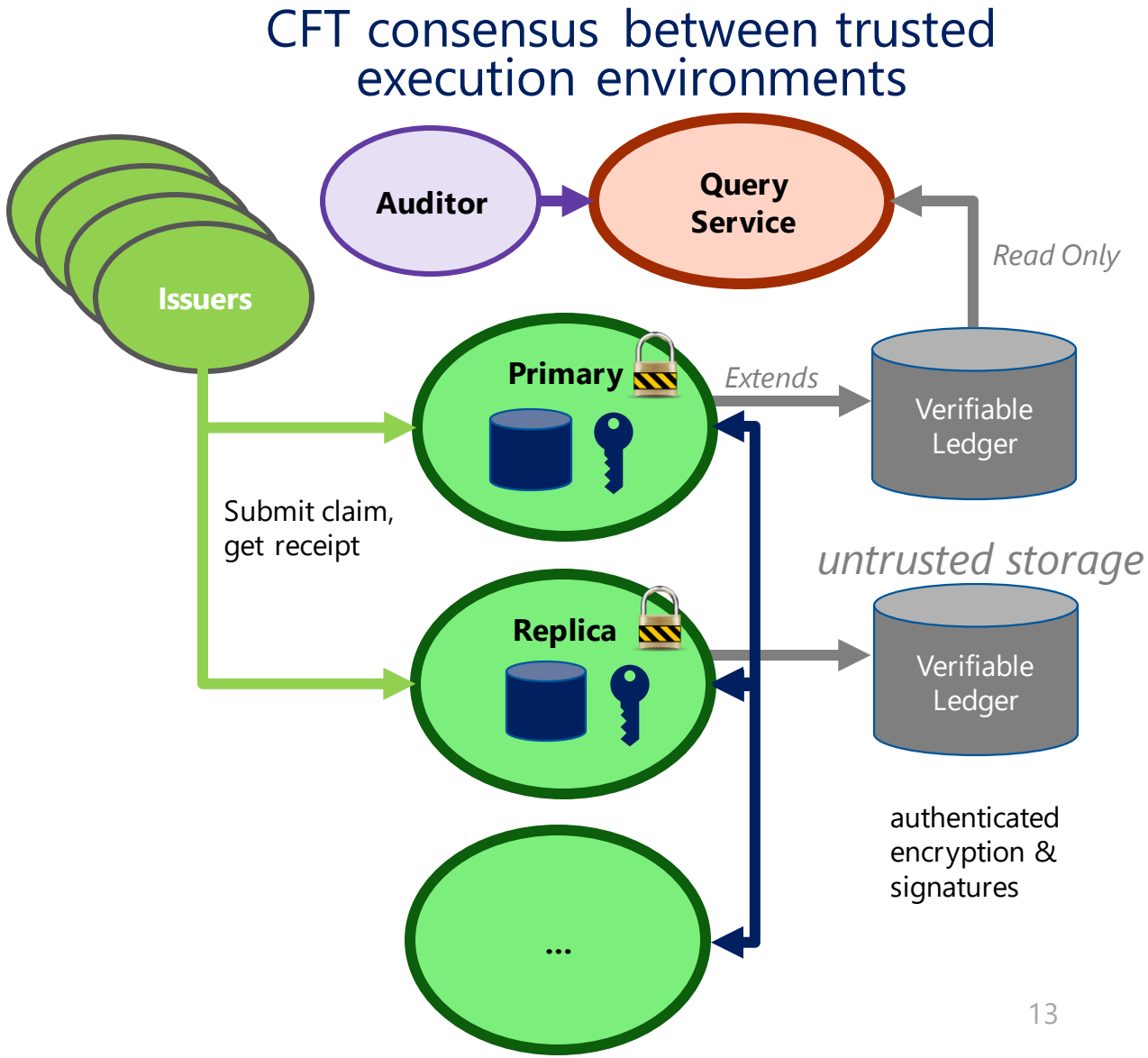
Prototype of SCITT Transparency Service & Verifier

Prototype based on Confidential Consortium Framework (CCF) framework: https://ccf.dev &

https://github.com/Microsoft/ccf

The ledger implementation is a chronological Merkle Tree, distributed in a CFT network of SGX protected enclaves.

The ledger implements draft-00 COSE claims and receipts, including a client library for checking receipts, as a basis for discussion.



Related Work and Working Groups in the IETF

- Envelopes & Receipts are based on COSE WG output
- Transparency service operations trustworthiness involves RATS WG output
- Transparency services borrow concepts and terms from the concept of Certificate Transparency defined in RC 6962

(Bar) BoF onsite

- 1700 CEST (30 min after the SECDISPATCH WG meeting)
- Meeting Point: Yard and Park Ensemble (Park Pavilion)
- Hybrid "Bar" BoF today come with online meeting links:
- https://teams.microsoft.com/l/meetup-join/19%3ameeting_OWUwMDhiZjEtYjkwNS00NDA0LTlmMTgtNGZhOGE0NmU3ZTcz%40thread.v2/0?context=%7b%22Tid%22%3a%272f988bf-86f1-41af-91ab-2d7cd011db47%22%2c%22Oid%22%3a%22bced92fe-7c20-456e-9afd-5b18c383de81%22%7d
- If you are spontaneous and/or have the time, join in. It's soon and (physically) in the same building