Trusted IoT Device Network-Layer Onboarding and Lifecycle Management

Build 5

BRSK over WIFI

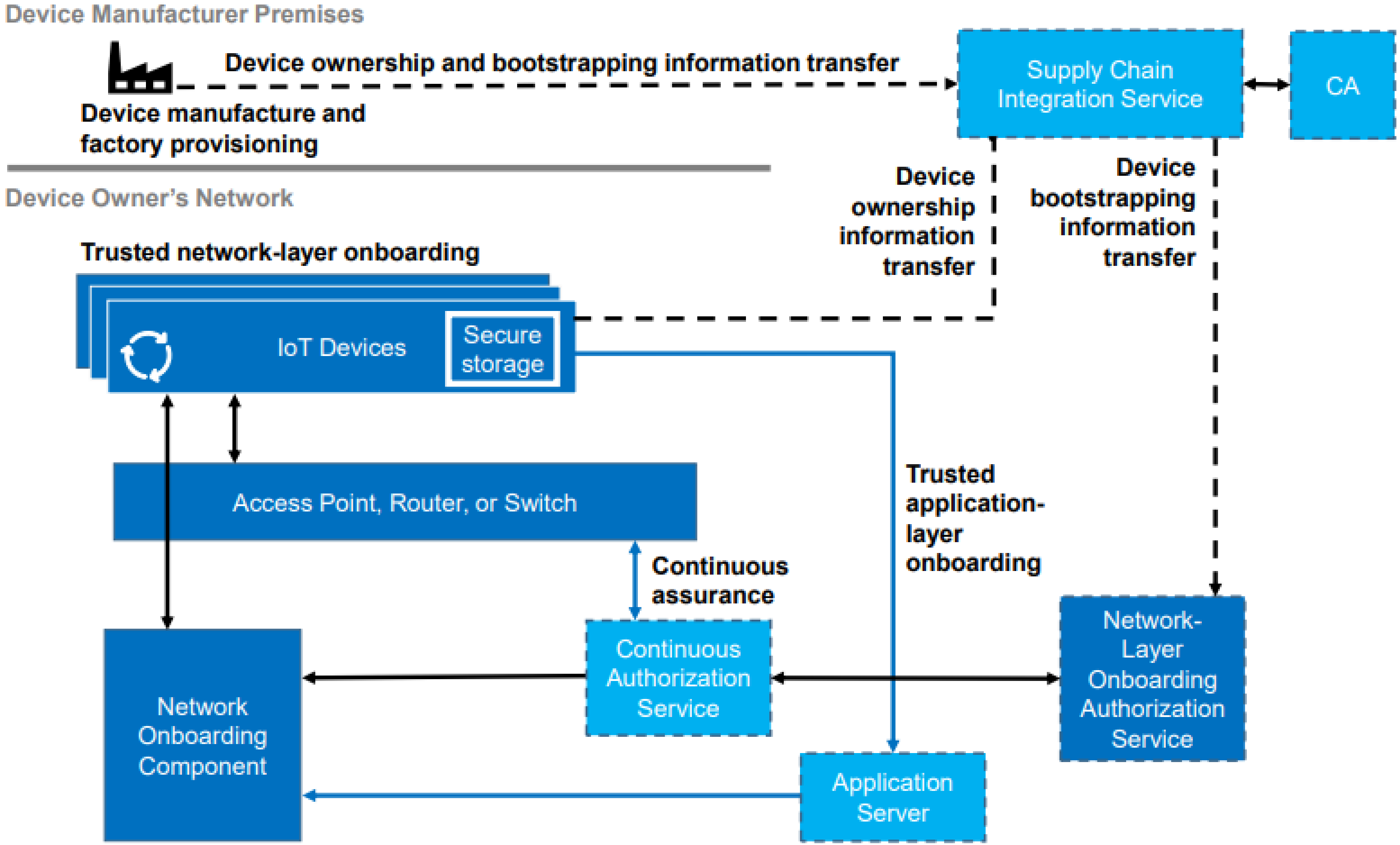
nquringminds

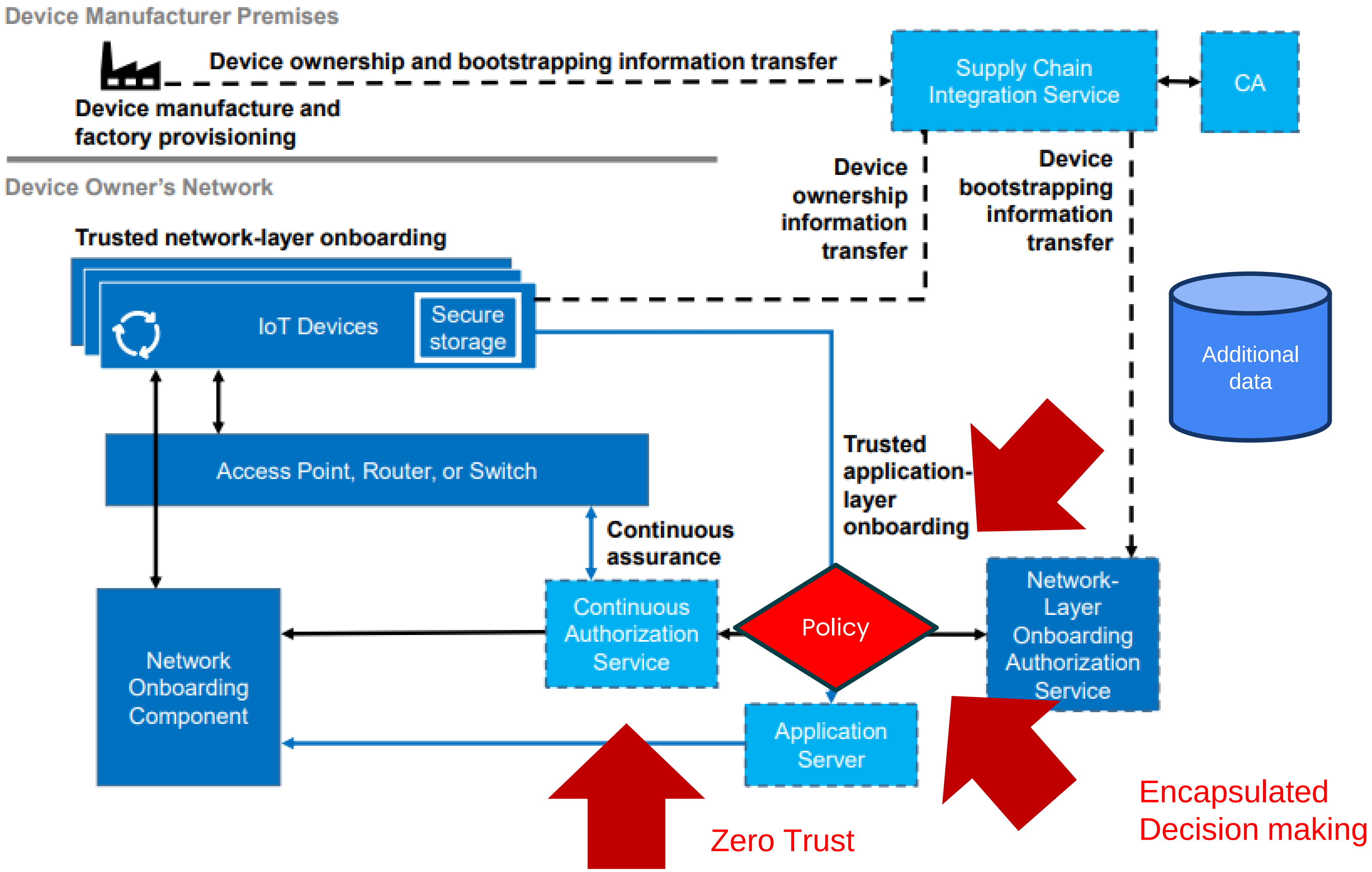
Trusted IOT Lifecycle

nquiringminds

Big Picture – why is this important

- No shared passwords (security)
- Low touch provisioning (usability)
- Policy encapsulation (flexibility)
- Supply chain integration (business + security)


Device Manufacturer Premises
Device ownership and bootstrapping information transfer
Device manufacture and factory provisioning
Supply Chain Integration Service
CA
Device Owner's Network
Device ownership information transfer
Device bootstrapping information transfer
Trusted network-layer onboarding
IoT Devices
Secure storage
Access Point, Router, or Switch
Trusted application-layer onboarding
Continuous assurance
Continuous Authorization Service
Network-Layer Onboarding Authorization Service
Network Onboarding Component
Application Server



Device Manufacturer Premises

Device ownership and bootstrapping information transfer

Device manufacture and factory provisioning

Supply Chain Integration Service

CA

Device Owner's Network

Device ownership information transfer

Device bootstrapping information transfer

Trusted network-layer onboarding

IoT Devices

Secure storage

Additional data

Access Point, Router, or Switch

Trusted application-layer onboarding

Continuous assurance

Network Onboarding Component

Continuous Authorization Service

Policy

Network-Layer Onboarding Authorization Service

Application Server

Zero Trust

Encapsulated Decision making

Incremental Challenge…..

nquiringminds

BRSKI WIFI mapping

- Full BRSKI mapping to WIFI onboarding
- Specifically, EAP-TLS certificate
- Full Secure element integration (changes to flow)
- Unique onboarding certificate issued to each device
- Allows for per device revocation (lifecycle management)

Incremental Challenge.....

nquiringminds

Flexible Policy

- Build 5 support generalised and extensible policy
- Implements the decision points implicit in BRSKI
- Extends them and drives from a generic policy engine
- Policy evidence is presented interoperable through W3C Verifiable credentials

Incremental Challenge…..

nquiringminds

Continuous assurance

- Lifecyle management!
- Reacts to changes in environment & information
- Reuses same policy framework
- Simple Radius server extension implementation
- Can be extended to other “Policy decisions

Demo

Policy dimensions

Build 5

Network owner: network owner defined preferences

Device owner: implicit and explicit device ownership is part of the decision (including receipts)

Manufacture: is the manufacture trusted? Is the manufacture trusted to be part of the decision

SBOM: supply chain and vulnerability process

MUD: device intent and dynamic behaviour

Verifiable Credentials

Why?

nquiringminds

- **Interoperable:** interworking standard for all aspects of the stack. More “explicit” than digital certs. Common expression for all crypto artefacts.
- **Data centric security** reduces API/ integration complexity. Integrity, provenance (identity) and revocation baked in.
- **Policy evidence:** better foundation for policy decisions. Explicit trust base. Easy to extend and integrate
- **Composable:** VCs can be combined and reasoned over

Learnings

Build 5

nquiringminds

Secure: EAP-TLS is a vast improvement, providing strong single device controls, but we need good protocols to provision.

DevID lifecycle: a single iDevID is fragile. We need more sophisticated identity lifecycle to manage SBOM and MUD at scale. Provisioning models!

Secure element: secure management of local credentials is critical. Code stacks need better support (e.g. EAP-TLS)

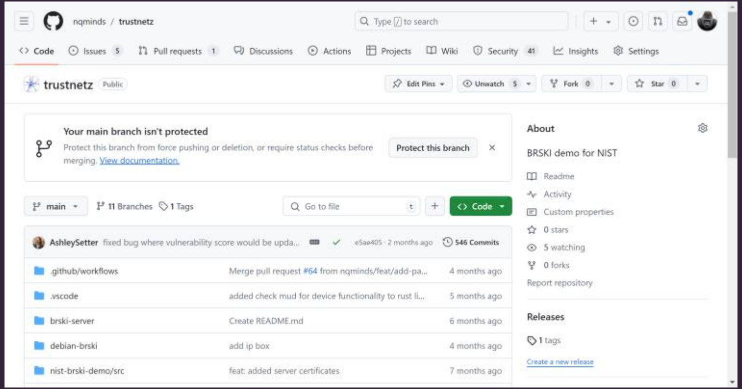
Flexible policy: expressing flexible policy with interoperable credentials, is invaluable to support different ecosystems and business models

Open source assets

https://trustnetz.org
https://github.com/nqminds/trustnetz

Questions

nick@nquiringminds.com
Nick Allott



Your main branch isn't protected

Protect this branch from force pushing or deletion, or require status checks before merging. View documentation.

BRSKI demo for NIST

AshleySetter fixed bug where vulnerability score would be upda... ✓ e5ae405 · 2 months ago 546 Commits

.github/workflows	Merge pull request #64 from nqminds/feat/add-pa...	4 months ago
.vscode	added check mud for device functionality to rust li...	5 months ago
brski-server	Create README.md	6 months ago
debian-brski	add ip box	4 months ago
nist-brski-demo/src	feat: added server certificates	7 months ago