SIEMENS
Ingenuity for life

Security for IEC/IEEE 60802

Goals, Constraints and Use Cases

K. Fischer, A. Furch, L. Lindemann, O. Pfaff, T. Pössler, G. Steindl

Problem Statement

SIEMENS
Ingenuity for life

- Establish the **goals** for a security contribution to IEC/IEEE 60802 (see [15])
- Identify the given **constraints** for this contribution
- Agree on the applicable **use cases** for it
  - This slide-deck: dive into credential imprinting use case and sketch a solution blueprint for discussion (see slides in the sub-section “Use Cases - Imprinting HowTo”)
  - Otherwise, the security use case description was moved to a text document. This topic shall be followed-up in the Friday ad-hoc calls

Next Steps

1. Continue the consideration of security use cases during the **Friday ad-hoc calls**
2. Provide an initial contribution proposal for the **next draft D1.3** in text form using inputs from
   - Goals, constraints and use cases (this deck)
   - Selected information from the presentations for the
     - IEEE Interim Session 2021-05-05 (forthcoming)
     - IEEE Plenary Session 2021-03-03
     - Weekly TSN-IA working group call 2021-02-22

Scope

SIEMENS
Ingenuity for life

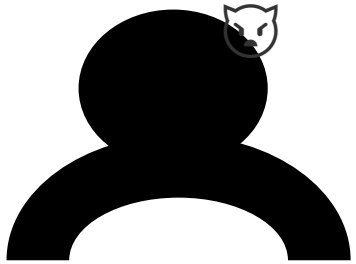
- **Security between stations**, in particular:
  - Discovering neighborhood relations
  - Provisioning of network configuration including TDMEs
  - Establishing streams including TDMEs
  - Synchronizing time
- **Shared security means,** considering the joint use for IEC/IEEE 60802 security and application/middleware security on a single station, in particular:
  - Profiling the set of cryptographic algorithms, their usage (e.g. TLS record layer [12] or 802.1AE [8]) and protocols for managing this usage (e.g. TLS handshake layer or 802.1X [3])
  - Using a single security resource, e.g. (HW) secure element upon a single station for this purpose
- **Securing-the-security**, in particular:
  - Supplying/managing initial keys/credentials/security configuration to individual stations in a secure manner

Guiding Principle

SIEMENS
Ingenuity for life

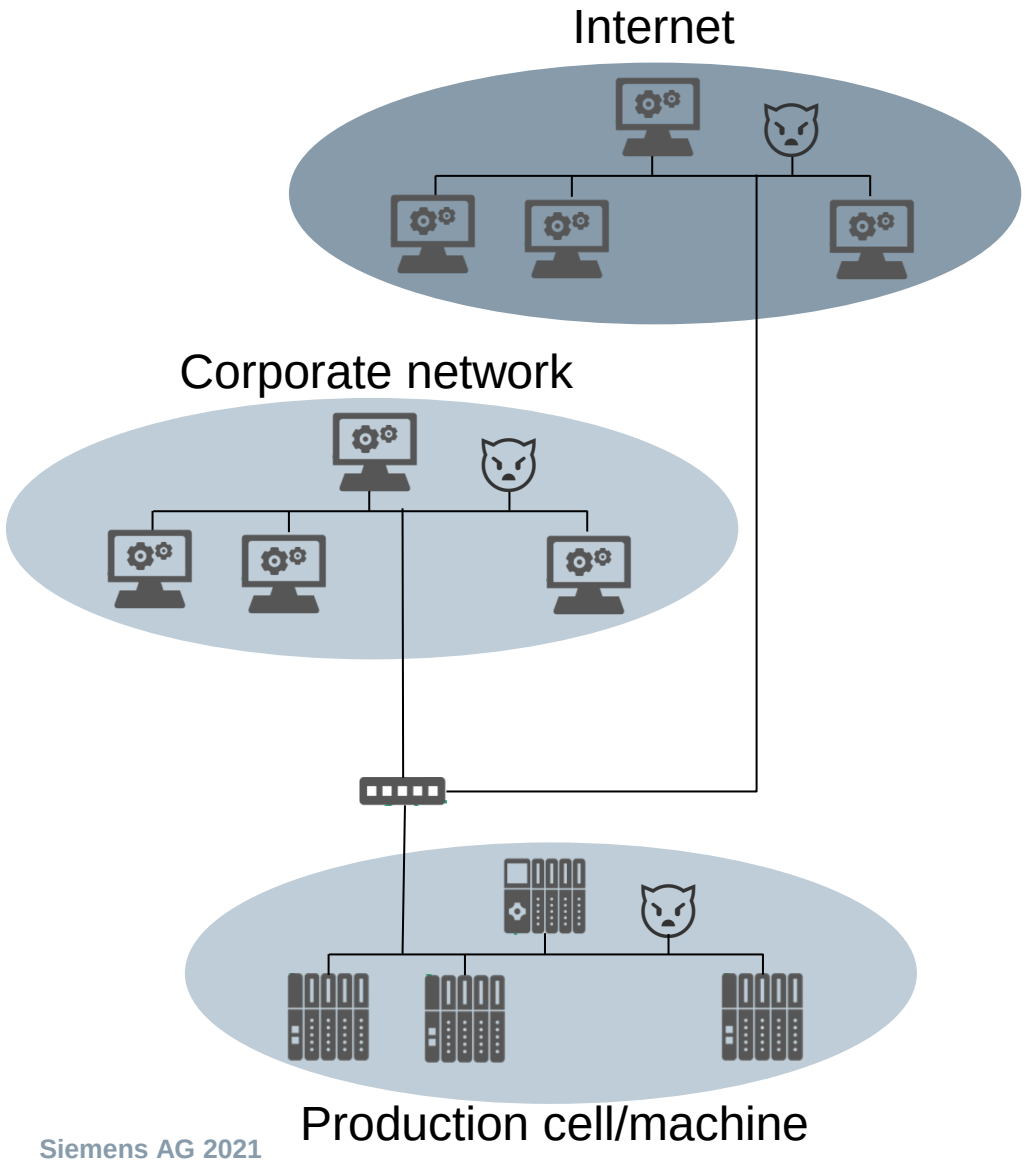
- Converged networks need a **‘converged security’** model
- Converged security means:
  i. An **interoperable solution** for IEC/IEEE 60802 security - covering the above identified scope, especially *security between stations*
  ii. **Co-existence** of IEC/IEEE 60802 security with the security for application/middleware as (sub)components on the same physical entity (station) – a part of the above identified scope, especially *shared security means* and *securing-the-security*

To-Be-Assessed Threats: Incarnations of Attackers



- Walk-in or network connected **human users**
  - Trying to exploit local interfaces of the designated victim e.g. USB port, JTAG port, serial interface by physically connecting (walk-in case)
  - Trying to exploit remote interfaces of the designated victim e.g. sending instructions to parameterization services (network connected case)
- Deployed **fake equipment** (pretending to be original)
  - Trying to exploit remote interfaces of the designated victim e.g. sending instructions to parameterization services (network connected case)
- Deployed **manipulated** (original) **equipment**
  - Trying to exploit remote interfaces of the designated victim e.g. sending instructions to parameterization services (network connected case)

To-Be-Assessed Threats: Scenarios



3. **Threats from the Internet:** attack vectors that apply when considering an Internet integration of production cells:
   - Exercised by (potentially) anyone
   - Exploiting remote access to the local network on layer 3-7

2. **Threats from local layer-3 networks**: attack vectors that apply when considering a production cell integration with corporate networks:
   - Exercised by e.g. staff employed by organizations that own/operate production cells
   - Exploiting remote access to the local network on layer 3-7

1. **Threats from the local layer-2 network**: attack vectors that apply when considering a production cell in isolation:
   - Exercised by e.g. staff with physical or local network access to the production cell e.g. service technicians
   - Exploiting physical access or remote access on layer 2

To-Be-Assessed Threats: Attack Vectors

SIEMENS
Ingenuity for life

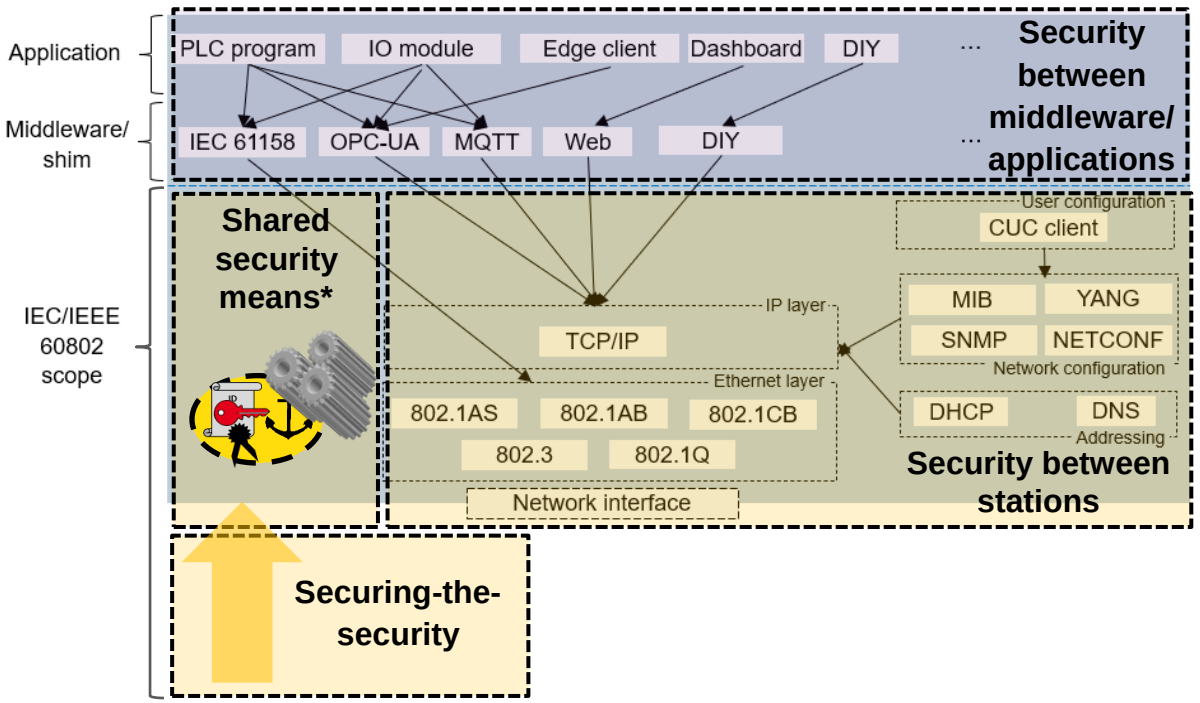
- Concerning individual physical entities (stations – as perceived in the network):
  a. **Impersonating/spoofing** of physical entities - e.g. forgery
  b. **Manipulating** of physical entities - e.g. corruption
- Concerning individual computing entities within a station (applications/middleware – as perceived in the network):
  c. **Impersonating/spoofing** of computing entities - e.g. forgery
  d. **Manipulating** computing entities - e.g. corruption
- Concerning communications between stations:
  e. **Insertion/manipulation** of individual messages or messaging contexts – e.g. tampering by manipulation or replaying, repudiation, denial-of-services
  f. **Reading** of message contents – e.g. eavesdropping
- Concerning accesses to (local) resources of a station by another (remote) station:
  g. **Abusing** (local) resources – e.g. accessing without privileges
  h. **Escalating** of privileges – e.g. accessing with inappropriate privileges

Functional Objectives

SIEMENS
Ingenuity for life

- **Message exchange protection** between identified stations:
  - Objective: protect communications against **forgery**, **tampering**, and **eavesdropping**
  - Features: (peer) entity identification and authentication, (data) integrity and confidentiality, replay protection
  - Scope: the system communications that are in-scope (see above for details)
- **Resource access authorization**:
  - Objective: protect system resources against **unauthorized access**
  - Features: coarse-grained authorization e.g. network isolation, fine-grained authorization e.g. application/middleware or network configuration resources
    - Prerequisite: message exchange protection, esp. (peer) entity identification and authentication
  - Scope: the system resources that are in-scope (see above for details)
- Note: message exchange protection plus resource access authorization allows to defeat/mitigate most attack vectors in the considered scenarios

Building Blocks



- In-scope (see Scope for details):
  i. **Security between stations**
  ii. **Shared security means**
  iii. **Securing-the-security**
- Out-of-scope for IEC/IEEE 60802: security between and at middleware/application components:
  - Protecting their message exchanges e.g. IEC 61158 communications between PLC programs and IO modules
  - Authorizing their resource accesses e.g. providing or changing instructions for the operation of an IO module

*: joined usage by application/middleware security is perceived but not shown here

Respecting Industrial Automation

SIEMENS
Ingenuity for life

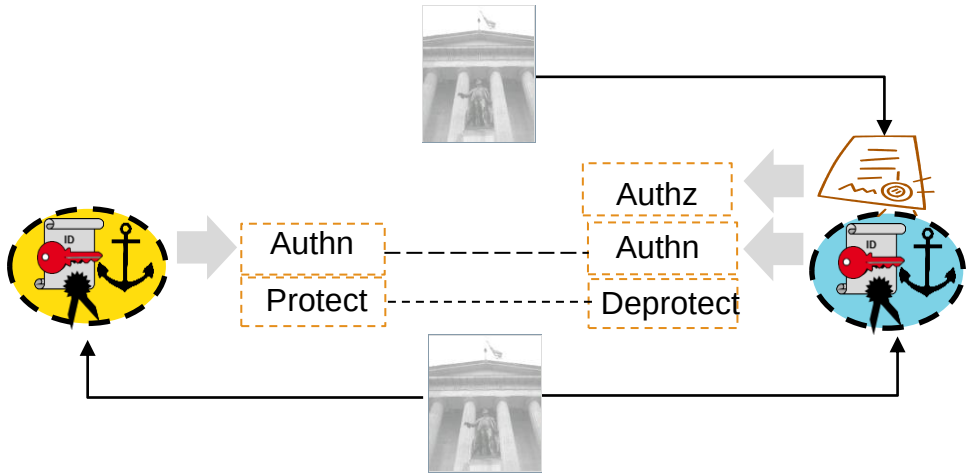
- IEC/IEEE 60802 security shall respect the essential characteristics/properties of industrial automation components/systems
- In particular characteristics/properties that differentiate industrial automation from IT must be addressed adequately. Differentiators from IT include:
  - Dedicated set of use cases (see [11]), e.g. ‘IA device replacement without engineering’, ‘machine cloning’
  - Embedded and constrained system components (lacking local user interfaces, limited computing power and memory, …)
  - System components that present physical entities and computing entities at the same time
  - Unattended operations
  - Undisturbed operations, e.g. bumpless key updates
  - Autonomy of production cells (with external cell control)
  - Deterministic communications particularly for time-aware streams
  - Physical world impacts, e.g. functional safety

Covering IEC 62443

SIEMENS
Ingenuity for life

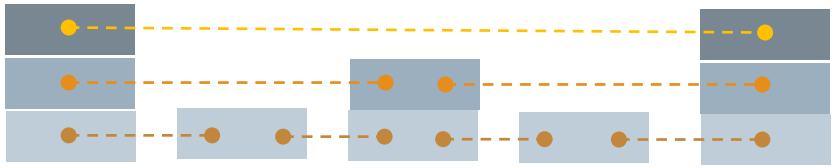
- IEC 62443-3-3 “System Security Requirements and Security Levels” (see [6]) and IEC 62443-4-2 “Technical Security Requirements for IACS Components” (see [14]) will be considered when creating the security contribution for IEC/IEEE 60802:
  - IEC 62443-3-3 and IEC 62443-4-2 **provide requirements** (from system and component perspective) to be addressed/considered by IEC/IEEE 60802 security (foundational requirements and security levels)
  - They do **not provide solutions** for their requirements, esp. do not provide a solution for IEC/IEEE 60802 security
- Applicability of further IEC 62443 documents for IEC/IEEE 60802 security needs further consideration
- Intended relationship with respect to IEC 62443:
  - IEC/IEEE 60802 security is **one building block** for the security of IEC 62443 IACS components
  - IEC/IEEE 60802 security shall **not conflict** with IEC 62443

Assigning Ownership for Keys/Authorizations



- Message exchange protection requires to assign authority for **keying control** e.g. keys for NETCONF message protection
- Resource access authorization requires to assign authority for **authorization control** e.g. for access control for NETCONF resources
- This aspect needs to be **well-balanced** by IEC/IEEE 60802 security esp. for long-living keys, considering:
  - **Actors** in **industrial automation**: manufacturers, machine builders, distributors, system integrators, owners, operators…
  - **Lifecycle** of an individual **station**: manufacturing, bootstrapping, operating, maintenance, off-boarding (according [13])
  - **Key/credential supply modes:** key/credential supply by nothing or by already assigned keys/credential e.g. LDevID-by-IDevID

Placing Security in the Protocol Stack



- Message exchange protection and resource access authorization require the allocation of **security mechanisms** in the **protocol stacks**
- This aspect needs to be **well-balanced** by IEC/IEEE 60802 security, considering:
  - **Going up/down**: there is no single, one-fits-all placement that can cover all demands:
    - Going up a stack allows to granularly protect entities/ objects and lengthens the achievable security span but tends to increase the overall complexity
    - Vice versa for going down the stack
  - **Footprint**: multiple simultaneous stack placements of security can improve functionality/flexibility but also increases its price-tag

Given Rules for Security Contribution

SIEMENS
Ingenuity for life

- **Re-use** existing security mechanisms specified by IEC, IEEE and IETF
- **Identify** possible white-spots
- **Not invent** solutions for possible white-spots - if such need arises dedicated projects shall be considered

Overview

SIEMENS
Ingenuity for life

- **Automation-domain use cases** - that shall be matched by IEC/IEEE 60802 security:
  - Are provided in the document ‘Use Cases IEC/IEEE 60802 V1.3’ (see [11])
  - Action item: the security contribution to IEC/IEEE 60802 shall comment on the automation use cases. Note: comments for the use case 30 ‘Security’ in [11] are already provided below
- **Security-domain use cases** - that shall be identified and elaborated in the security contribution for IEC/IEEE 60802:
  - Candidates are proposed below
  - Action item: the security use cases shall be elaborated in a working group document
- Constraint: **security** shall be added in a way that allows all **automation use cases** to still be **fulfilled**

Comments on Use Case 30 “Security”

SIEMENS
Ingenuity for life

- *Topping-up*: by adopting cryptographic security a whole set of **new security use cases** emerges and shall be considered (see below for candidates)
- *White-spot*: **authorization** is a fundamental goal of security and should be added
- *Refinements*: apart from spelling-out ‘*authorized system entities*’ (see Glossary), **availability** is a constraint for security and should be referred to as such
- *Scope*: albeit **protection against rogue/fake applications (running on authenticated stations)** is no deliverable of IEC/IEEE 60802 security, it should be addressed as security threat/attack vector. May result in separation requirements for application/middleware entities and the network interface upon an end station
- *Differentiation*: differentiate „**rogue**“ (*right entity behaving wrong*) and „**fake**“ (*wrong entity behaving right or wrong*)

Proposals for Security Use Cases (1)

SIEMENS
Ingenuity for life

1. **Checking the equipment under control**: serves to motivate/explain the checking (actual vs. expected) of IA components as prerequisite before imprinting keys/credentials for security (see [7] for an elaboration of this concern in case of natural persons)

2. **Imprinting during bootstrapping/commissioning**: serves to motivate/explain the supply of (initial) keys/credentials and differentiates according several credentialing modes e.g. human-operated/assisted vs. unattended/automated credentialing (see [1] for a basic model)
   - See slides in the subsection “Use Cases – Imprinting HowTo” for a solution blueprint (for discussion)

3. **Instructing equipment about security**: serves to motivate/explain the security instructions towards IA components:
   - Per owner/operator: security-only by default
   - Per individual IA component: security always-on or on/off, enabled ciphers….
   - Per application/communication relation: security on/off, authentication-only/authenticated encryption…

4. **Peer entity authentication:** serves to motivate/explain (peer) entity authentication and its inherited features e.g. key agreement or authorization including checking actual/expected (see [4] for an elaboration of this concern in case of IT)

Proposals for Security Use Cases (2)

SIEMENS
Ingenuity for life

5. **Message exchange protection:** serves to motivate/explain the protection of communications between stations including its prerequisites, in particular peer entity authentication (including identification)
   - Note: this security use case overlaps with use case 30 “Security” in [11]. It is proposed to serve a transfer
6. **Proving self-asserted information:** serves to motivate/explain binding between peer entity authentication and self-asserted information - applicable in case of such feature e.g. identification and maintenance data or topology discovery data
7. **Resource access authorization:** serves to motivate/explain access control and its prerequisites, in particular peer entity authentication (including identification)
8. **Credential/key update during operation**: serves to motivate/explain the handling of key aging, considers/establishes bumpless-ness
9. **Credential/key revocation/invalidation during operation**: serves to motivate/explain the handling of premature termination of key/credential lifetime
10. **Crypto algorithm-expiry/agility:** serves to motivate/explain the handling of the dawn of crypto algorithms
11. **Robust supply of security core function**: random number generation, key protection…

Security Precondition

SIEMENS
Ingenuity for life

- Security, for example TLS, needs **credentials**, important examples for TSN-IA components are:
  - Initial credential aka **IDevID** (see [9])
    - Identifies a component based on **manufacturer knowledge** about an instance e.g. *product serial number, product type, place and time of production, MAC address(es)…* and allows its authentication
    - Supplying IDevID credentials is a **manufacturer responsibility**
  - Local credential aka **LDevID** (see [9], also called **LDevID-generic** in the following)
    - Identifies a component based on **owner/operator (or system integrator) knowledge** about an instance and in **application-independent fashion** e.g. *IP address, DNS name, place of deployment…* and allows its authentication
    - Supplying LDevID credentials is an owner/operator (or system integrator) responsibility
  - NETCONF-specific credential (see [5], called **LDevID-NETCONF** in the following)
    - Identifies a component based on **owner/operator knowledge** about an instance and in **application dependent fashion** e.g. *application name/identifier/role…* and allows its authentication
    - Supplying LDevID-NETCONF credentials is an owner/operator (or system integrator) responsibility
  - ABC … XYZ application-specific credentials (also called **LDevID-ABC … LDevID-XYZ)**
    - Similar as above, usage of described model is at the discretion of the individual application

Security Precondition–NETCONF-over-TLS Illustration

SIEMENS
Ingenuity for life

TDME

LDevID-NETCONF

IA component

LDevID-NETCONF*

Connect NETCONF-over-TLS

Trigger TLS handshake**

Send **LDevID-NETCONF* EE** certificate

Validate **LDevID-NETCONF* EE** certificate
(requires **LDevID-NETCONF CA** certificate aka trust anchor)

Check server identity

Send **LDevID-NETCONF EE** certificate

Send PoP for **LDevID-NETCONF EE** certificate
(requires **private key** matching the EE certificate)

Validate **LDevID-NETCONF EE** certificate
(requires **LDevID-NETCONF CA** certificate aka trust anchor)

Check client identity

Check PoP for **LDevID-NETCONF EE certificate**

Send PoP for **LDevID-NETCONF* EE** certificate
(requires **private key** matching the EE certificate)

Check PoP for **LDevID-NETCONF* EE certificate**

Send first NETCONF <*> APDU
protected with TLS record layer

Send Protected request

Reply with Protected response

*: another EE certificate object instance (naming, public key…) of the same type (X.509 certificate), another private key
** not all TLS handshake options and messages are shown here

Converged Security Precondition

SIEMENS
Ingenuity for life

- Converged security needs means that allow **owners/operators** to **supply LDevID-* credentials** to IA components in a **manufacturer-independent way**

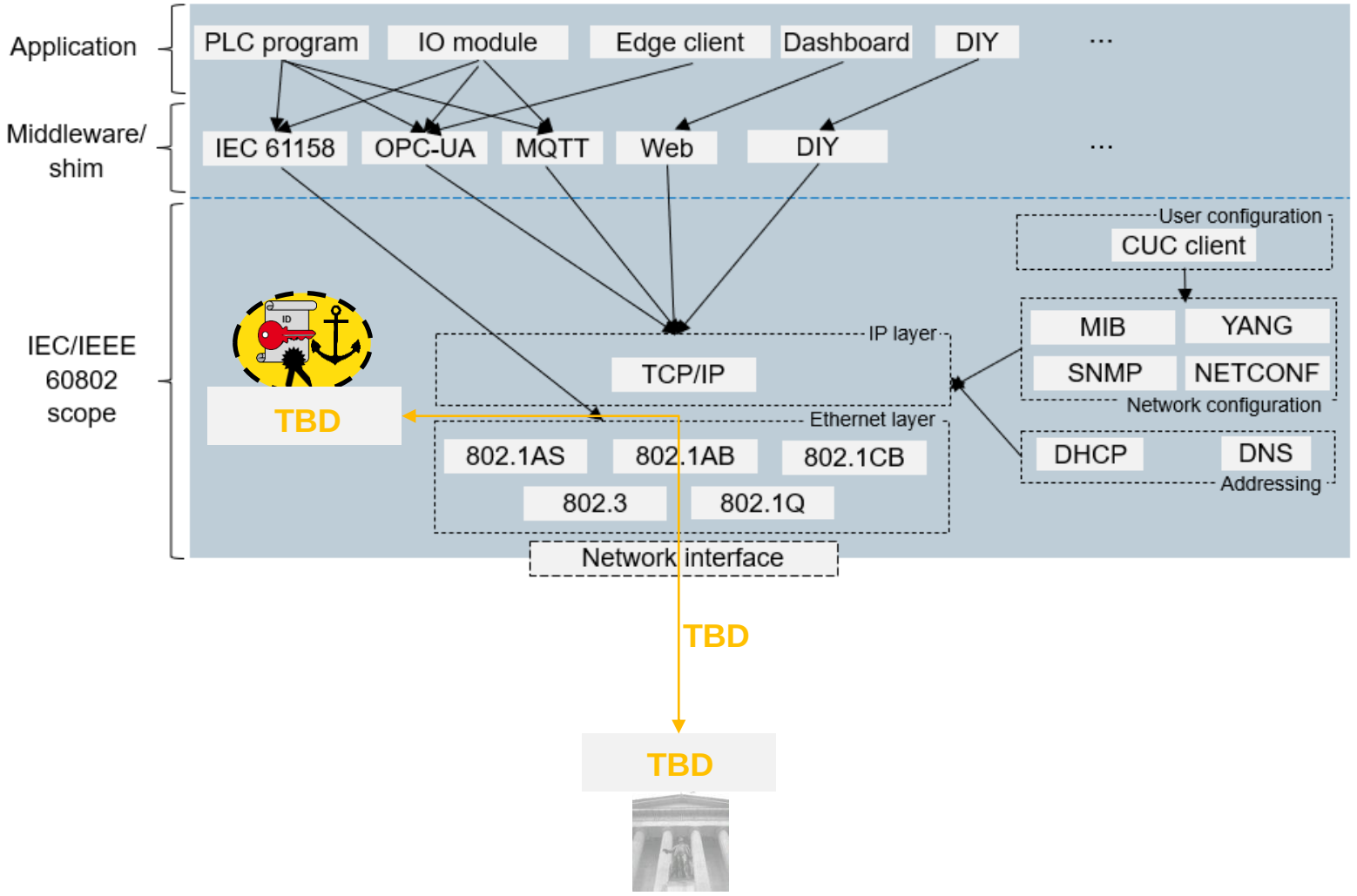
  a. Supplying LDevID-* credentials to standalone machines (without [connectivity to] security infrastructure):

     - According to **push** over a network
     - In protected fashion

  b. Supplying LDevID-* credentials to integrated machines (with connectivity to security infrastructure):

     - According to **push** or pull over a network
     - In protected fashion

Converged Security Precondition-Illustration



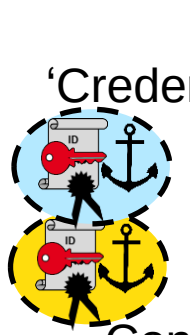
Solution Blueprint (for Discussion)

SIEMENS
Ingenuity for life

- Q: how do owners/operators push LDevID-* credentials into IA components?
  - A: use the NETCONF service upon the IA component as a responder
- Q: how to protect this supply?
  - A: use manufacturer-equipped IDevID credential to protect the LDevID-* supply
- Q: which security protocol to use for making this supply?
  - A: NETCONF-over-TLS according RFC 7589 [5]
- Q: how to address the storage of LDevID-* credential objects?
  - A: a dedicated YANG module (potential starting point: draft-ietf-netconf-keystore-21.txt, see [16])
- Q: how to persist the LDevID-* credential objects?
  - A: a local matter, to-be-illuminated in the “Shared security means” section
- Q: who is allowed to supply and manage LDevID-* credentials?
  - A: according a role-based security model; using role assignment (e.g. “CredentialManager” role name) in LDevID-generic credentials (residual challenge: controlling the supply of the LDevID-generic CA certificate)
- Q: which entity can supply and manage LDevID-* credentials?
  - A: a functional role “CredentialManager” that may be assigned to misc. system components

Solution Blueprint-Illustration

SIEMENS
Ingenuity for life



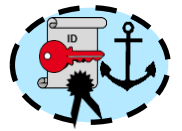
TDME with
'CredentialManager' functional role**

IDevID

LDevID-NETCONF

IA component

IDevID*



Connect NETCONF-over-TLS

TLS handshake protected with IDevID/IDevID* (see above)

Send NETCONF <edit-config(LDevID)> APDUs protected with TLS record layer

LDevID-NETCONF*

Happens once (for LDevID-NETCONF)

TLS handshake protected with LDevID-NETCONF/LDevID-NETCONF* (see above)

Send NETCONF <*> APDUs protected with TLS record layer

Happens often (network configuration)

*: another EE certificate object (naming, public key), another private key
** may also be allocated with another system component

Resulting Actions

SIEMENS
Ingenuity for life

- Profile NETCONF/YANG to support credential management in a manufacturer-independent and secure way
  - Profile NETCONF-over-TLS based on IDevID and LDevID
  - Profile YANG module for credential management
  - Profile EE certificate contents for NETCONF-over-TLS
  - Profile role model for resource access authorization
  - …

Abbreviations

SIEMENS
Ingenuity for life

APDU	Application Protocol Data Unit
ASN.1	Abstract Syntax Notation 1
Authn	Authentication
Authz	Authorization
CA	Certification Authority
DIY	Do It Yourself
DLL	Data Link Layer
DN	Distinguished Name
E2E	End-to-End
EE	End Entity
EUC	Equipment Under Control
FW	FirmWare
HTTP	Hypertext Transfer Protocol
HW	HardWare
IA	Industrial Automation
IACS	Industrial Automation and Control Systems
ID	Identifier
IDevID	Initial Device ID
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IT	Information Technology
JOSE	Javascript Object Signing and Encryption
LDevID	Locally significant Device ID
MAC	Medium Access Control (networking) or Message Authentication Code (security)
OAuth	Open Authorization
OID	Object ID
OPC	Open Platform Communications
OT	Operational Technology
PLC	Programmable Logic Controller
PoP	Proof-of-Possession
SW	SoftWare
TBD	To Be Defined
TDME	TSN Domain Management Entity
TLS	Transport Layer Security
TSN	Time-Sensitive Networking

Glossary (1)

SIEMENS
Ingenuity for life

Attack (RFC 4949, [2]): An intentional act by which an entity attempts to evade security services and violate the security policy of a system

Authorization (RFC 4949): An approval that is granted to a system entity to access a system resource

Availability (RFC 4949): The property of a system or a system resource being accessible, or usable or operational upon demand, by an authorized system entity, according to performance specifications for the system

Bridge (IEEE 802.3, [10]): A functional unit that interconnects two or more IEEE 802 networks that use the same data link layer (DLL) protocols above the medium access control (MAC) sublayer, but can use different MAC protocols. Forwarding and filtering decisions are made on the basis of layer 2 information

Certification path (RFC 5280): A chain of multiple (public-key) certificates comprising a certificate of the public key owner (the end entity) signed by one CA, and zero or more additional certificates of CAs signed by other CAs

Certification path validation (RFC 5380): The verification of the binding between the subject distinguished name and/or subject alternative name and subject public key

Computing entity (DIY): A system entity that has no own incarnation in the physical world e.g. PLC program/IO module

Credential (IEEE 802.1AR, [9]): Information that an entity (a person or device) possesses that allow it to make a verifiable claim of identity, i.e., to be authenticated

(Data) confidentiality (RFC 4949): The property that data is not disclosed to system entities unless they have been authorized to know the data

Glossary (2)

SIEMENS
Ingenuity for life

(Data) integrity (IEEE 802.1AE): A property whereby data has not been altered in an unauthorized manner since it was created, transmitted or stored

End station (IEEE 802.3): A functional unit in an IEEE 802 network that acts as a source of, and/or destination for, link layer data traffic carried on the network

Fake entity (DIY): A wrong system entity (incapable to authenticate itself – if that would be demanded) behaving right or wrong

Integrity (RFC 8446): Data sent over the channel after establishment cannot be modified by attackers without detection

Message authentication (IEEE 802.1AE): If the message arrives authenticated, the cryptographic guarantee is that the message was not modified in transit and that the message originated from an entity with the proper cryptographic credentials

(Peer) entity authentication (RFC 4949): The process of verifying a claim that a system entity or system resource has a certain attribute value. An authentication process consists of two basic steps:

- *Identification step*: Presenting the claimed attribute value (e.g., a user identifier) to the authentication subsystem
- *Verification step*: Presenting or generating authentication information (e.g., a value signed with a private key) that acts as evidence to prove the binding between the attribute and that for which it is claimed

Physical entity (DIY): A system entity that has an incarnation in the physical world e.g. station

Glossary (3)

SIEMENS
Ingenuity for life

Risk (RFC 4949): An expectation of loss expressed as the probability that a particular threat will exploit a particular vulnerability with a particular harmful result

Rogue entity (DIY): A right system entity (authenticated or capable of authenticating itself) behaving wrong (because of e.g. being corrupted internally)

Secure element: (Global Platform): A tamper-resistant platform (typically a one chip secure microcontroller) capable of securely hosting applications and their confidential and cryptographic data (for example cryptographic keys) in accordance with the rules and security requirements set by well-identified trusted authorities

Spoofing (IEEE 802.1AE): Claiming a fraudulent identity for purposes of mounting an attack

Station (IEEE 802.3): An end station or bridge

(System) entity (RFC 4949): An active part of a system -- a person, a set of persons (e.g., some kind of organization), an automated process, or a set of processes (see: subsystem) -- that has a specific set of capabilities

Threat (RFC 4949): A potential for violation of security, which exists when there is an entity, circumstance, capability, action, or event that could cause harm

Trust anchor (RFC 5280): A CA certificate that serves as a trust anchor for the certification path validation

References, Chronologically Ordered (1)

SIEMENS
Ingenuity for life

1. Stajano, F.; Anderson, R: The Resurrecting Duckling: Security Issues for Ad-hoc Wireless Networks, 1999
2. IETF RFC 4949: Internet Security Glossary, Version 2, 2007
3. IEEE 802.1X-2010: IEEE Standard for Local and Metropolitan Area Networks – Port-Based Network Access Control, 2010
4. IETF RFC 6125: Representation and Verification of Domain-Based Application Service Identity within Internet Public Key Infrastructure Using X.509 (PKIX) Certificates in the Context of Transport Layer Security (TLS), 2011
5. IETF RFC 7589: Using the NETCONF Protocol over Transport Layer Security (TLS) with Mutual X.509 Authentication, 2015
6. IEC 62443-3-3: System Security Requirements and Security Levels, 2015
7. NIST SP 800-63: Digital Identity Guidelines, 2017
8. IEEE 802.1AE-2018: IEEE Standard for Local and Metropolitan Area Networks – Media Access Control (MAC) Security – Revision D 1.3, 2018
9. IEEE 802.1AR-2018: IEEE Standard for Local and Metropolitan Area Networks–Secure Device Identity, 2018
10. IEEE 802.3-2018: IEEE Standard for Ethernet, 2018
11. IEC/IEEE 60802: Use Cases IEC/IEEE 60802 V1.3, Draft (work-in-progress), 2018
12. IETF RFC 8446: The Transport Layer Security (TLS) Protocol Version 1.3, 2018

References, Chronologically Ordered (2)

SIEMENS
Ingenuity for life

13. IETF RFC 8576: Internet of Things (IoT) Security: State of the Art and Challenges, 2019
14. IEC 62443-4-2: Technical Security Requirements for IACS Components, 2019
15. IEC/IEEE 60802: Time-Sensitive Networking Profile for Industrial Automation, Draft 1.2, 2020
16. IETF NETCONF: A YANG Data Model for a Keystore, Draft (draft-ietf-netconf-keystore-21.txt, work-in-progress). 2021

Authors

Kai Fischer, Siemens AG, T RDA CST SES-DE,
kai.fischer@siemens.com

Andreas Furch, Siemens AG, T RDA CST SES-DE,
andreas.furch@siemens.com

Lars Lindemann, Siemens AG, DI FA CTR ICO ARC,
lars.Lindemann@siemens.com

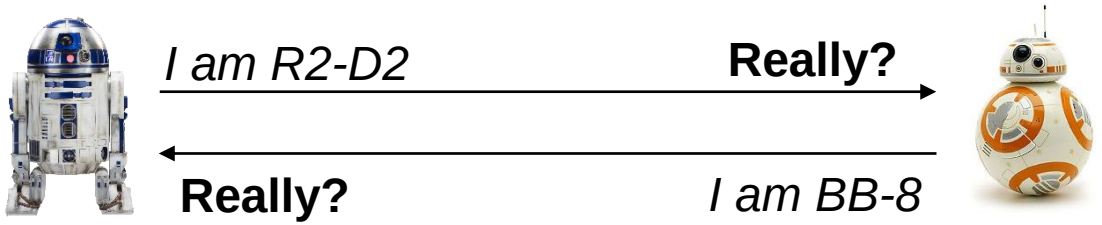
Oliver Pfaff, Siemens AG, T RDA CST,
oliver.pfaff@siemens.com

Thomas Pössler, Siemens AG, RC-AT DI FA DH-GRAZ SAS,
thomas.poessler@siemens.com

Günter Steindl, Siemens AG, DI FA TI ART EA,
guenter.steindl@siemens.com

What Are Credentials – In General?

- Objects to authenticate oneself as well as to authenticate others in a distributed system



- Best current practices (IT domain):

Username credential

Ubiquitous in IT - **Web user authentication**

Authenticate oneself (the human user): username and (static resp. one-time) password

Authenticate others (at Web server/application): hashed password resp. seeding info

Shared secret key credential

Ubiquitous in IT - **enterprise IT**

Authenticate oneself: ticket, session key (e.g. Kerberos)

Authenticate others: pre-shared key (e.g. Kerberos)

Public key certificate credential

Ubiquitous in IT - **Web server authentication**

Authenticate oneself (the TLS server): EE certificate (X.509), private key

*Authenticate others (*at TLS clients*)*: trust anchor i.e. trusted CA certificate (X.509)

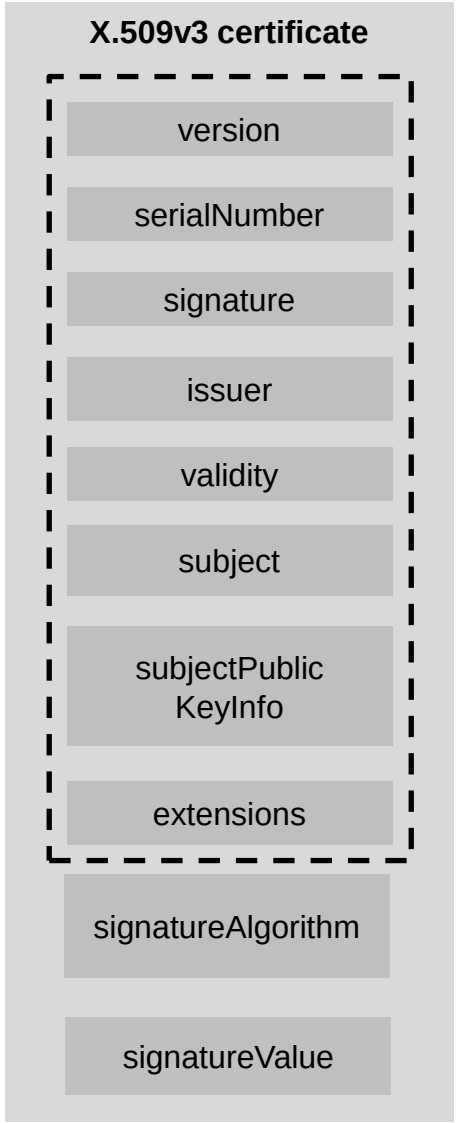
Which Credential Type Is Needed for NETCONF?

SIEMENS
Ingenuity for life

- NETCONF-over-TLS servers *must* be equipped with a **public key certificate credential** according IETF RFC 7589, see [5] (the term “LDevID-NETCONF” is used for notational convenience). This comprises:
  i. **EE certificate** (plus intermediate sub-CA certificates) to identify oneself
  ii. **Private key** (matching the public key in the EE certificate) to provide proof for the own identification
  iii. **CA certificate(s)** aka **trust anchors** to validate the identification of others
- NETCONF clients *may* be equipped with a **public key certificate credential**. This comprises following items
  i. **EE certificate** (plus intermediate sub-CA certificates) to identify oneself
     - Same object type as above but another object instance (other naming attributes, public key…)
  ii. **Private key** (matching the public key in the EE certificate) to provide proof for the own identification
     - Same object type as above but another object instance (other value, possibly also other algorithm type…)
  iii. **CA certificate(s)** aka **trust anchors** to validate the identification of others
     - Same object (type and instance) as above
  - Note: there are alternatives to public key certificate credentials in the “*NETCONF client=human user*” arena (username credentials) but no actual alternatives in the “*NETCONF client=IA component arena*”

How Do EE Certificates for NETCONF Servers Look*?

SIEMENS
Ingenuity for life

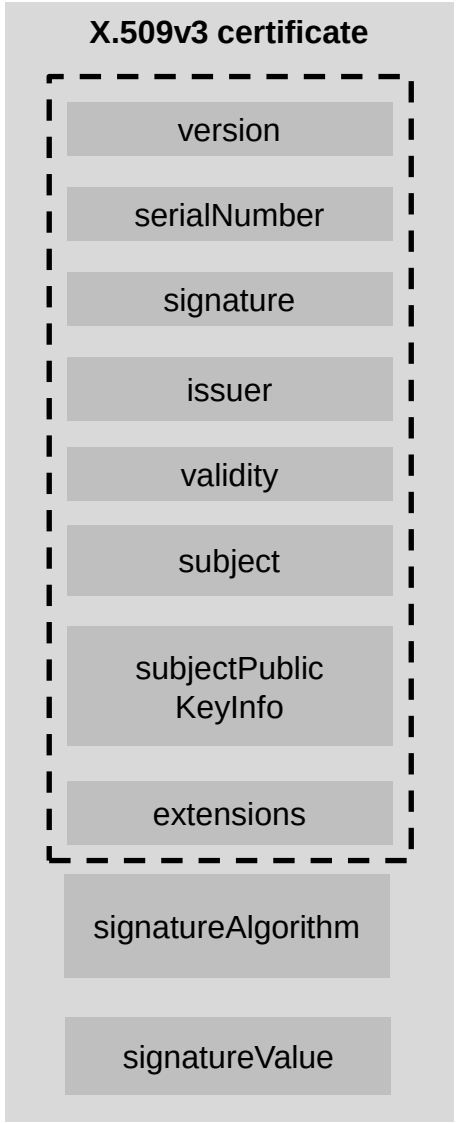


- **version** (asn1:INTEGER): 2 (X.509v3)
- **serialNumber** (asn1:INTEGER): any integer
- **signature** (asn1:AlgorithmIdentifier): any algorithm identifier (OID)
- **issuer** (asn1:Name): any DN value (name of issuing CA)
- **validity**:
  - notBefore (asn1:Time): any time value
  - notAfter (asn1:Time): any time value
- **subject** (asn1:Name): empty according RFC 6125, see [4]
- **subjectPublicKeyInfo**:
  - algorithm (asn1:AlgorithmIdentifier): any algorithm identifier (OID)
  - subjectPublicKey (asn1:BIT STRING): any bitstring
- **extensions** (asn1:Extensions): non-empty and containing
  - subjectAltName (asn1:Extension): any dNSName, or iPAddress, (partial) wildcards are possible, cf. Section 6 of RFC 6125, see [4] for further details
  - keyUsage (asn1:Extension): shall qualify for TLS server role
- **signatureAlgorithm** (asn1:AlgorithmIdentifier): any algorithm identifier (OID)
- **signatureValue** (asn1:BIT STRING): any bitstring

*: the exciting stuff is highlighted by yellow markup.
These items also explain why manufacturer credentials can not be used here

How Do EE Certificates for NETCONF Clients Look*?

SIEMENS
Ingenuity for life



- **version** (asn1:INTEGER): 2 (X.509v3) or 0 (X.509v1)
- **serialNumber** (asn1:INTEGER): any integer
- **signature** (asn1:AlgorithmIdentifier): any algorithm identifier (OID)
- **issuer** (asn1:Name): any DN value (name of issuing CA)
- **validity**:
  - notBefore (asn1:Time): any time value
  - notAfter (asn1:Time): any time value
- **subject** (asn1:Name): empty (RFC 6125, see [4]) or any DN value (subject name)
- **subjectPublicKeyInfo**:
  - algorithm (asn1:AlgorithmIdentifier): any algorithm identifier (OID)
  - subjectPublicKey (asn1:BIT STRING): any bitstring
- **extensions** (asn1:Extensions): possibly empty, if non-empty
  - subjectAltName (asn1:Extension): any rfc822Name, dNSName, or iPAddress, cf. Section 7 of RFC 7589, see [5] for further details
  - keyUsage (asn1:Extension): must qualify for TLS client role
- **signatureAlgorithm** (asn1:AlgorithmIdentifier): any algorithm identifier (OID)
- **signatureValue** (asn1:BIT STRING): any bitstring

*: the exciting stuff is highlighted by yellow markup.
These items also explain why manufacturer credentials can not be used here

What Happens with the NETCONF EE Certificates?

SIEMENS
Ingenuity for life

- **NETCONF servers**: actually the exact same thing as with Web server certificates in HTTP-over-TLS i.e. this facet is a clone of Web security best practices
- **NETCONF clients**: deviates from the Web security best practices. Following terms and conditions apply
  - Presenting EE certificates of NETCONF clients: clones Web security approach (same as in HTTP-over-TLS)
  - Cryptographically validating EE certificates of NETCONF clients: clones Web security too
  - Checking the PoP for EE certificates of NETCONF clients: clones Web security too
  - Post (success) validation assessment of EE certificates of NETCONF clients: deviates from Web security
    - HTTP-over-TLS: "*the server has no external knowledge of what the client's identity ought to be and so* ***checks*** *(…) are* ***not possible***" according Section 3.2 in RFC 2818 (leaving checks to the application)
    - NETCONF-over-TLS: "*The NETCONF server MUST verify the identity of the NETCONF client to ensure that the incoming request to establish a NETCONF session is legitimate before the NETCONF session is started"* according Section 7 in RFC 7589, see [5] (describing a EE certificate to NETCONF username mapping and using this for checking)

Illustrating IA Devices/Controllers

SIEMENS
Ingenuity for life