Siemens Corporate Technology | May 2015

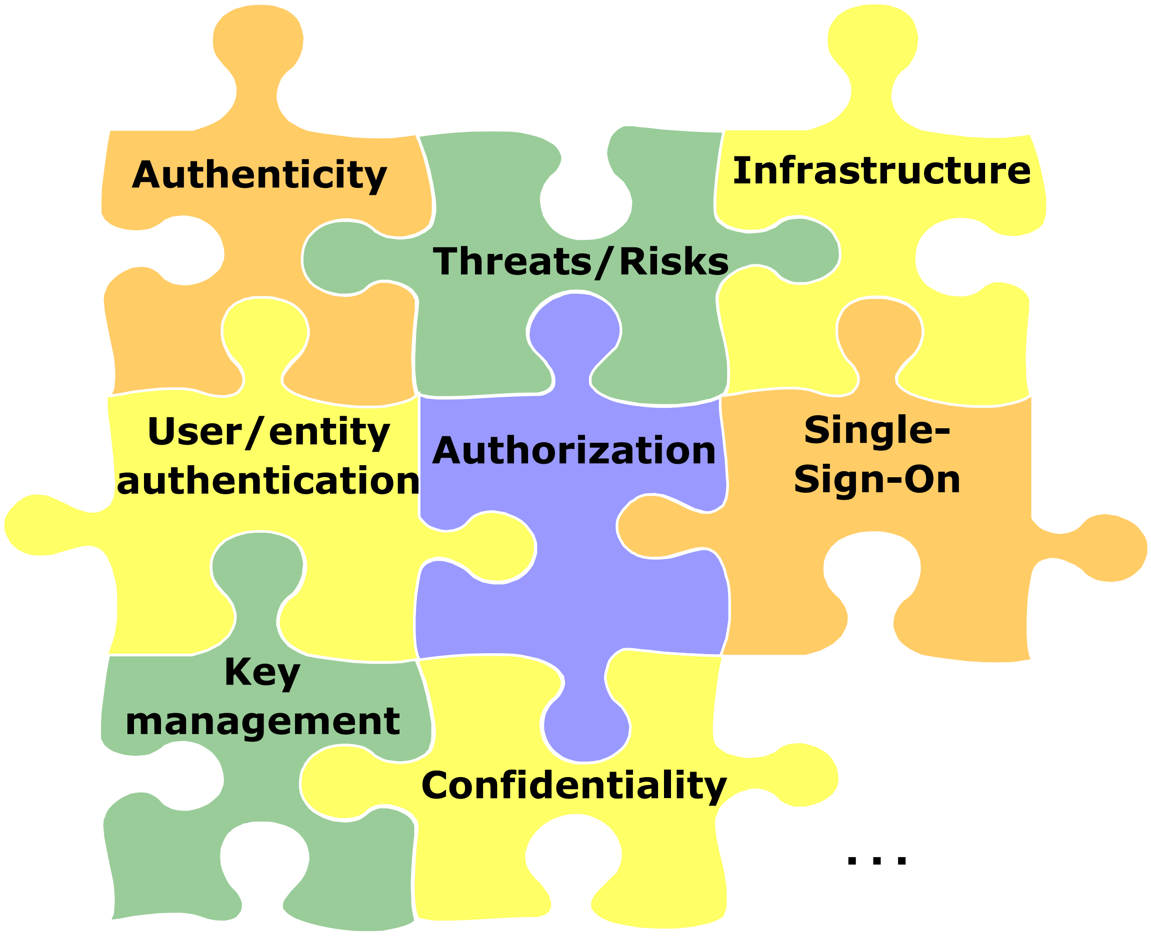
Security for the Industrial Internet – *A New Challenge!*

VDE-Ringvorlesung *Datensicherheit in der EDV-Welt*, Saarbrücken

Contents

- **Setting-the-scene**
  - IT-security
  - Industrial Internet
- **Challenges**
  - Constrained devices and networks
  - Connectivity, de-perimeterization
  - Not only human users
  - Not only IT-applications
  - Rethink access control
  - Accommodate physical goods
- **Conclusions**

IT-Security: A Jigsaw Puzzle with Many Pieces



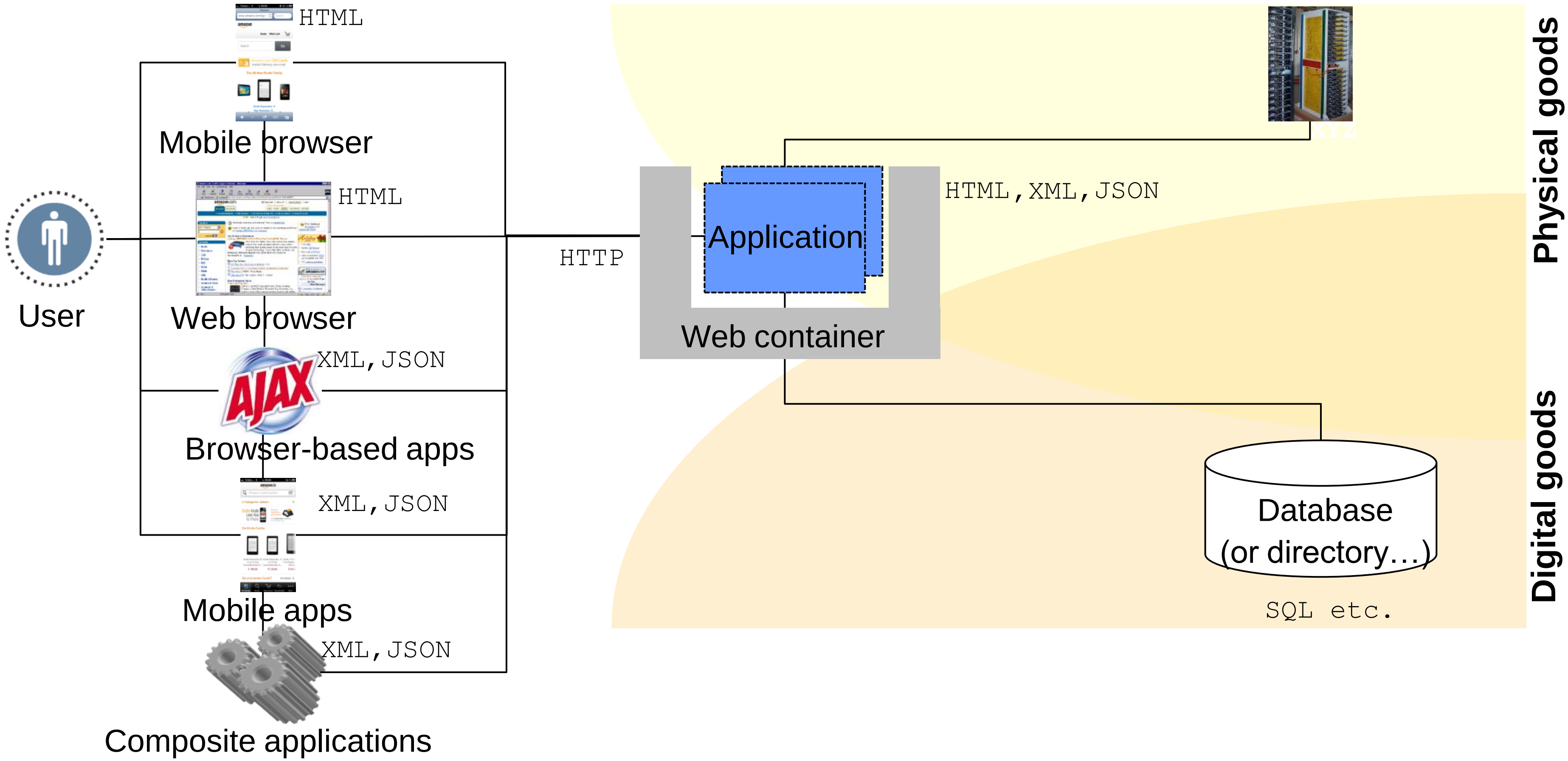
Evolution of the Web

- *Web* – since ca.1995:
  - **Digital goods** - *reproduction, relocation of item instances at almost no cost*
  - Examples: Web pages, messages, contact/mapping information, mp3 files…
  - Use cases: bulletin boards, data sharing, publishing, team collaboration, commerce…
  - Aspects:
    - Static vs. dynamic objects
    - Human vs. machine-readable
- *Web-of-systems* – from 2015, adding:
  - **Physical goods** - *reproduction, relocation of item instances at cost*
  - Examples: cars, lighting devices, smoke sensors, thermostats, wind turbines…
  - Use cases: building/industry automation, connected car, healthcare, smart home…
  - Aspects:
    - Consumer vs. investment goods
    - Individually vs. legal entity-owned

Setting-the-Scene

Evolution of Web Technologies

SIEMENS



1995 Database-backed applications, desktop browsers, read-only

2000 Read/write aka Web 2.0, AJAX clients

2005 Mobile browsers/apps, Composite applications

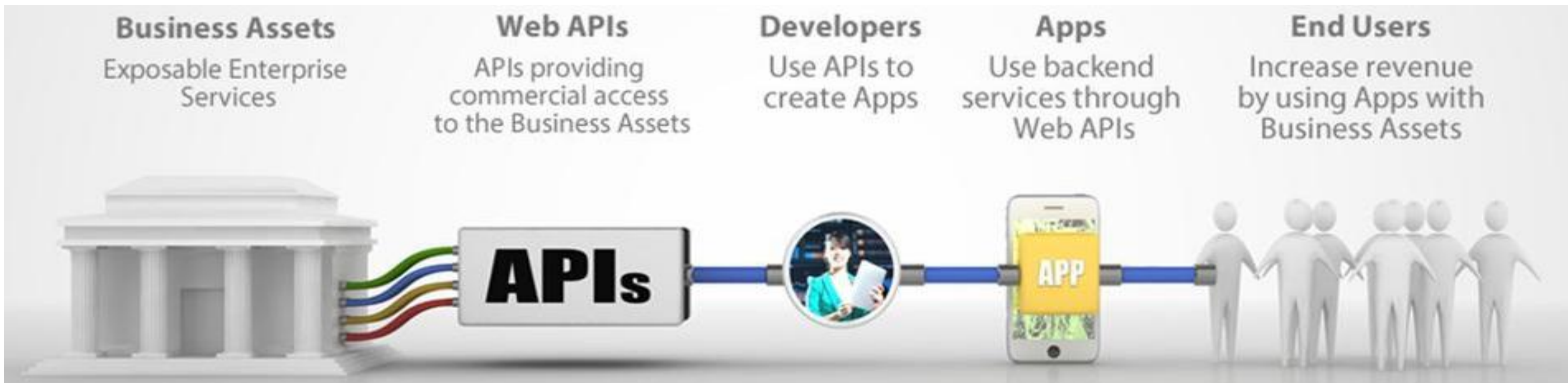
2010 Cloud, advent of applications backed by physical goods (self-provision CPU, RAM…)

An Industrial Internet Example

SIEMENS

The **API/app** pattern - empowering the Web evolution:

Source: API Economy Drivers



Predix, an instance of the API/app pattern and GE's software platform for industrial Internet:

Source: Industrial Internet & Advanced Manufacturing (M. Annunziata, GE)

Serves
private resources

API

API

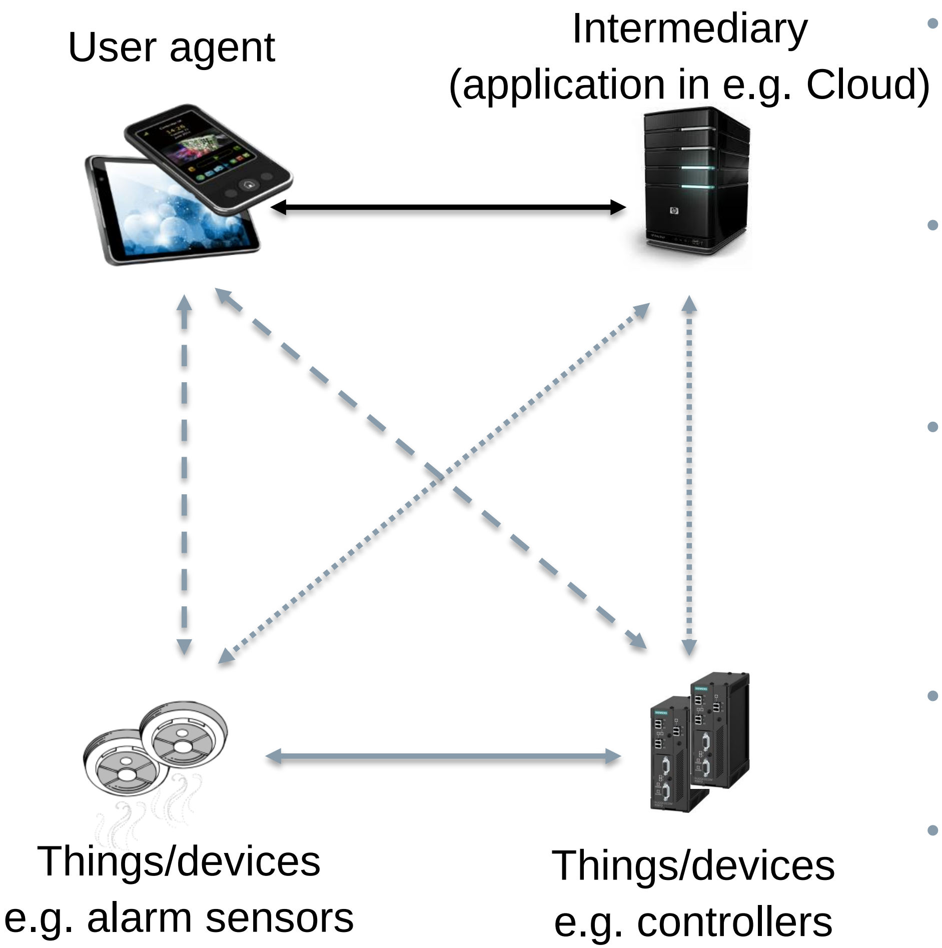
Exposes **public** resp.
partner-facing endpoints

User agents, clients

Developed by 3rd parties

Users

With More IoT/WoT Added



- *Allow things/devices to be engaged/engage*
- Variety of topologies
  - Direct interactions between things
  - Mediated interactions
- Variety of connectivity styles
  - Near field…wide-area
  - Intermittent…undisturbed
- Variety of communication patterns
  - Request/response
  - Publish/subscribe
  - One-way
- Variety of protocols
  - AMQP, CoAP, HTTP, MQTT, XMPP…
- Variety of constraints on things and networks
  - RFC 7228 device classes 0/1/2

What We'll Be Talking About?

- To meet industrial Internet (resp. I4.0, IoT/WoT) needs, IT-security will fundamentally change from what we know today
- Drivers behind this change:
  - ***Constrained devices and networks****:* require new security mechanisms
  - ***Connectivity, de-perimeterization***: demand new risk-management approaches
  - ***Not only human users***: things appear as callers that have to be identified/authenticated
  - ***Not only IT-applications***: and things also appear as callees
  - ***Rethink access control***: device-friendly authorization approaches are needed
  - ***Accommodate physical goods****:* representing and handling ownership relations much more complex than for digital goods

Device Classes – IETF RFC 7228

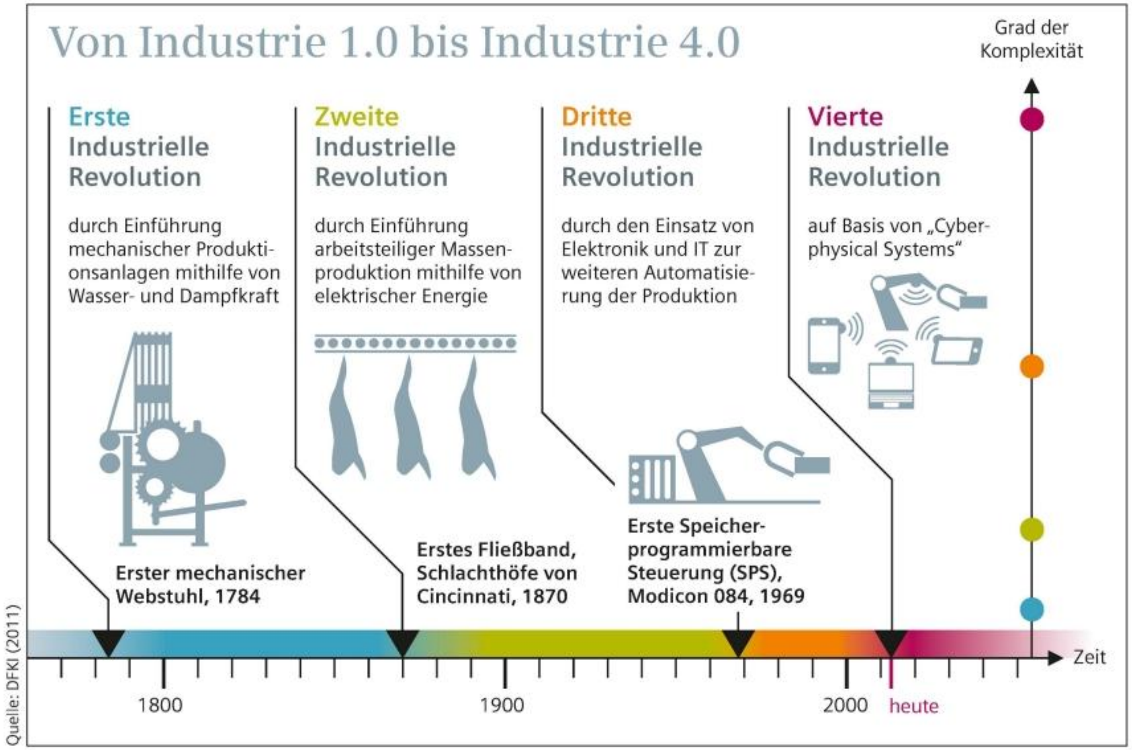
- **Class 2**:
  - Data size (memory): 50 KB
  - Code size (flash, disk): 250 KB
  - Can interact with Internet nodes. Example protocol: HTTP-over-SSL/TLS
- **Class 1**:
  - Data size (memory):10 KB
  - Code size (flash, disk): 100 KB
  - May interact with Internet nodes. Example protocol: CoAP-over-DTLS
- **Class 0**:
  - Data size (memory): <<10 KB
  - Code size (flash, disk): <<100 KB
  - Depend on intermediaries (e.g. class 1 or 2 components) to interact with Internet nodes

Innovation Needs

- Common Internet/Web **security mechanisms** do **not match** class 1/0 devices
- Results in a need to tune security mechanisms
- Required measures include:
  - **Down-scaling** of security system implementations
  - **Lightweight security** mechanisms covering
    - *Cryptographic primitives*: algorithms to transform data
    - *Cryptographic objects*: representations of transformed data along with metadata e.g. JOSE
    - *Security tokens*: (cryptographic) objects to make assessments about system actors e.g. JWT
    - *Security protocols*: means to exchange cryptographic objects or security tokens e.g. DICE

	Cryptographic primitives		Cryptographic objects				Security tokens				Security protocols		
	Asymmetric	Symmetric	ASN.1	XML	JSON	CBOR	ASN.1	XML	JSON	CBOR	SSL/TLS	DTLS	DICE
Class 2													
Class 1													
Class 0													

IT-Network Utilization of Industrial Products



I1.0	I2.0	I3.0	I4.0

I1.0
- Resources: private
- Exposure: no IT

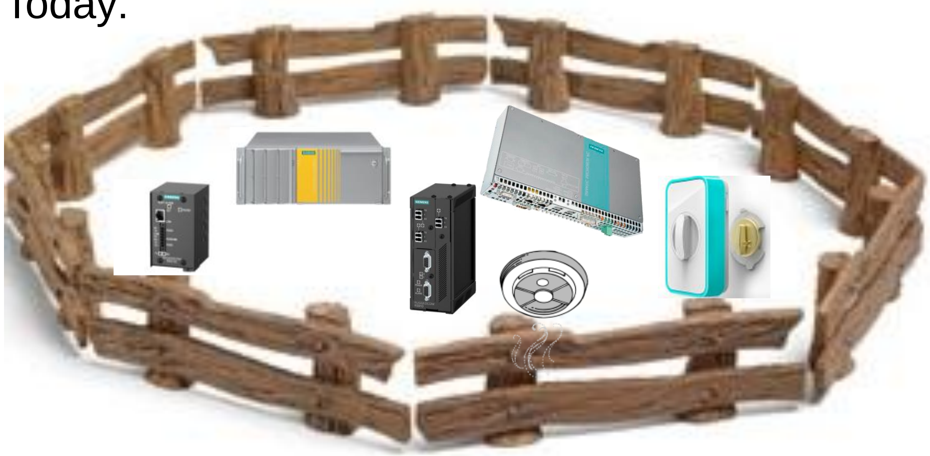
I2.0
- Resources: private
- Exposure: no IT

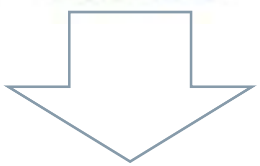
I3.0
- Resources: private
- Exposure: **private enclosures** (“*things in the garage*”)

I4.0
- Resources: private
- Exposure: **public/partner-facing** (“*stuff on the street*”)

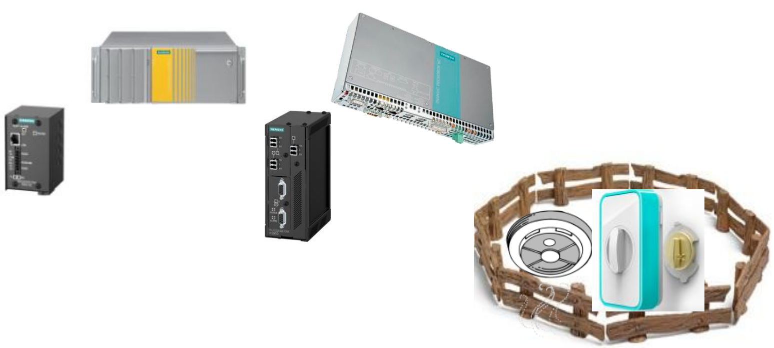
Innovation Needs

Today:





Tomorrow:



- The **premise disappears**
  - *Drivers*: opening-up is needed to enable new ecosystems
  - *Obstacles*: invalidates the old security approach “*we are safe - we live on an own island and rely on own technologies*”
- Results in a need to adapt mindsets and priorities in industrial product development
- Required features include:
  - **Intrusion detection/prevention**
    - Block suspicious traffic
  - **Throttling**
    - Enforce rate-limits, dynamically determined
  - **Risk-based authentication**
    - Determine authentication schemes in a context-aware, adaptive way
    - Include step-up and re-authentication

SIEMENS

Challenges - Not Only Human Users

On the Internet Nobody Knows You’re a Dog

“On the Internet, nobody knows you’re a dog.”

- Callers resp. requestors **need to be authenticated**
  - Before providing access to protected resources, accepting critical inputs
  - Examples:
    - *Protected resources*: mail or bank account contents, identity/location data, ordering systems, team shares etc.
    - *Public resources*: Wikipedia, Internet search, maps etc.
- Current practice is to authenticate human users against IT-applications or networks
- Current caller authentication practices are:
  - Username/static passwords resp. API client identifiers/secret (ubiquitous)
  - Username/one-time-passwords (some)
  - Public/private key credentials (sporadically)

Innovation Needs

Things/devices as callers (classes 0/1/2)

- The set of actors **increases by 1 order of magnitude** (approx. 7''' users, 50''' devices).
- New actors have **new characteristics**:
  - Lack of user interfaces and displays
  - Unattended operation
  - Difficulties in keeping secrets secret (*human users might have them too*): scrutinization
- The current practices (= username/password) rely on an **anti-pattern**:
  - Users or providers may leak credentials
  - Users forget credentials
  - Credentials get overexposed (HTTP Basic)
  - 3rd parties that ask users for shared secrets
- Results in a need to re-think mechanisms for the authentication of callers. Required features include:
  - Device **identity bootstrapping**, **credentialing**
  - Device **authentication**

Million Dollar Question

SIEMENS

- User space: 7’’’ users on this planet. Contenders of the ‘user authentication’ race are – all starting with a vast coverage of this space:
  - Governments: birth certificates, passports, ID-cards, driver licenses…
    - *DNF*: governmental authentication does not propagate into IT – have no relevant market share in ‘user authentication events in IT’
  - Telco’s: IMSIs, SIM cards, PINs
    - *Lost*: network access authentication does not propagate into applications – have no relevant market share in ‘user authentication events in IT-applications’
  - Enterprises: Windows domain credentials, employee cards…
    - *Other race*: IdPs have no own incentive to extend user base, have an incentive to accommodate external relying parties → no real chance to drive their market share in ‘user authentication events’, face a minor threat (BYOI)
  - Web giants: usernames (mail addresses), static passwords, security questions
    - *Lead*: the current leader in the ‘user authentication in IT’ market – number of users, number of authentication events, relevance for users, openness for relying parties, security features
- Device space: 50’’’ devices projected to have Internet connectivity by 2020
  - ➢ ***Who will be the kings-of-the-hill in terms of ‘device authentication’ market share?***

Whom Do I Talk To?

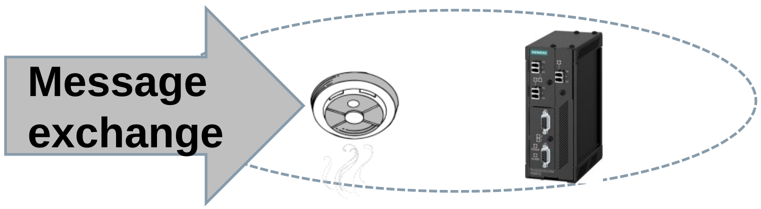
If you speak standard protocols

…nobody knows you are fake

- Callees resp. responders also **need to be authenticated**
  - Before sending confidential information e.g. credit card numbers, passwords to them
  - Before getting sensitive data from them e.g. personal mails or other information that can trigger actions on caller side
- Current practice is to authenticate applications and hosts in networks
- The best current practices technologies are:
  - Kerberos in case of applications in Windows domains e.g. Exchange servers
  - SSL/TLS in case of Web applications, mail servers etc.
  - SSH in case of remote hosts

Challenges - Not Only IT-Applications

Innovation Needs

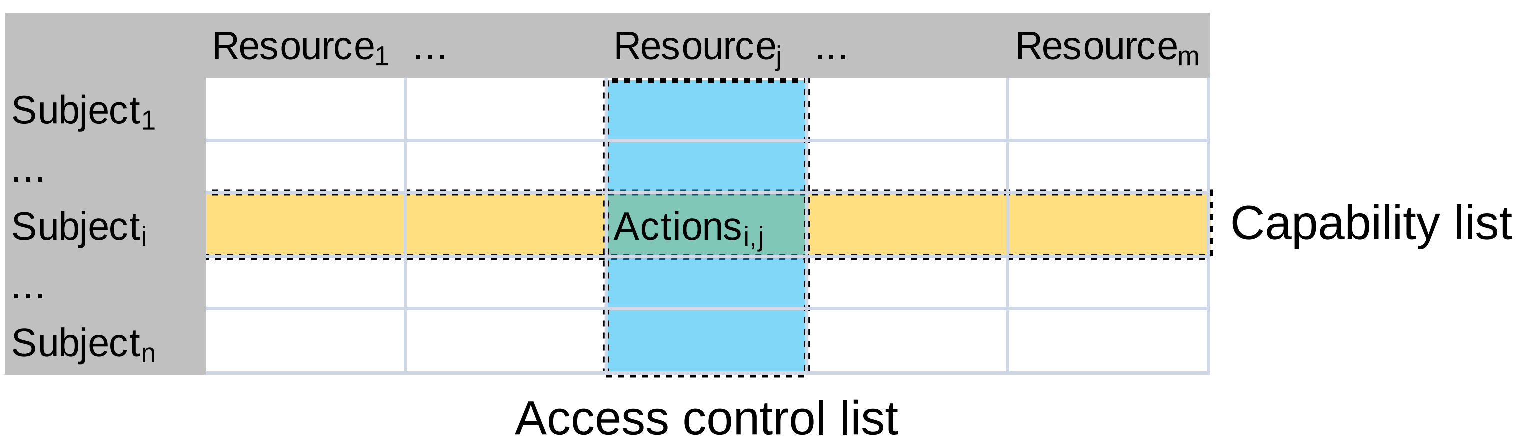


Things/devices as callees
(classes 0/1/2)

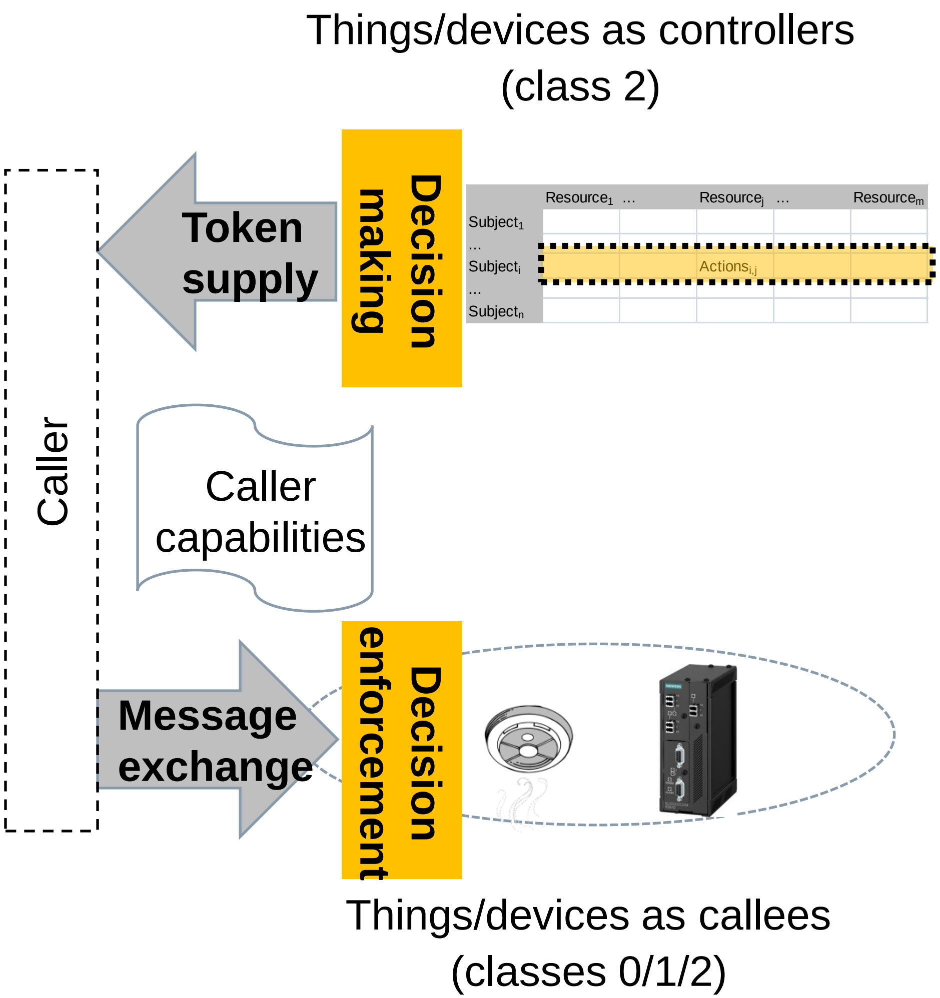
- The current practices **do not match**
  - Kerberos: confined to Windows domains i.e. office/enterprise IT
  - SSL/TLS (PKI-based): ca. 5’’ SSL/TLS server (leaf) certificates exist worldwide but 50’’’ devices projected to have Internet connectivity by 2020 - a factor of 10’ for a technology (PKI) that is known to be tedious
  - SSH (public key cryptography with no/lightweight infrastructure): tailored according specific use cases in IT
- Results in a need to re-think mechanisms for the authentication of callees
- Required features: as for caller authentication
  - Device **identity bootstrapping**, **credentialing**
  - Device **authentication**

What May A Caller Do?

- Callers resp. requestors **need to be authorized**
  - Before providing access to protected resources (caller authentication is necessary but not sufficient)
- Current practice is to implement an authorization technology that incarnates an access control matrix
- Best current practices approaches are:
  - Web (CMS): URL-level authorization enforcement by Web containers
  - Web (OAuth 2.0): O-to-O authorization for individually-owned Web resources
  - Web (UMA): O-to-* authorization for individually-owned Web resources
  - Operating systems: access control lists in Windows/Linux (controlling file system objects)

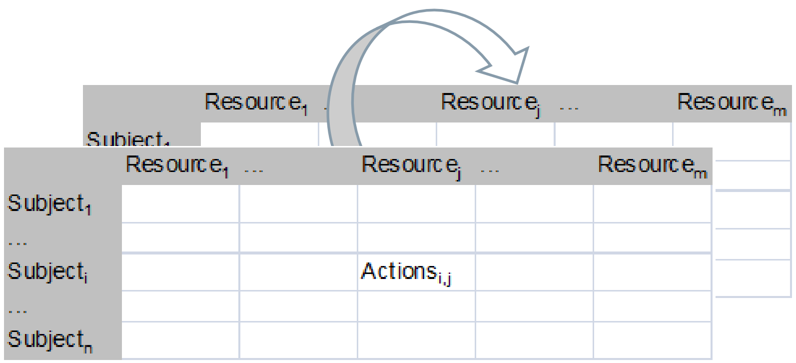


Innovation Needs



- **Decision enforcement** needs to happen **close to the resource**. It can typically not be offloaded from constrained things/devices
- **Decision making** is complex (implements the access control matrix in some way) and needs to be **offloaded**
- Externalization of decision making prefers a **push** mode
  - Pull adds backchannel roundtrips per request
- This requires security tokens capable of describing **capabilities** of the requesting subject along with protocols to acquire, supply and evtl. validate, revoke such objects
- These means have to be embedded with the protocol stack used to interact with the device
  - Corresponding means recently appeared in the HTTP stack (class 2)
  - Corresponding means for class 1/0 emerge just now

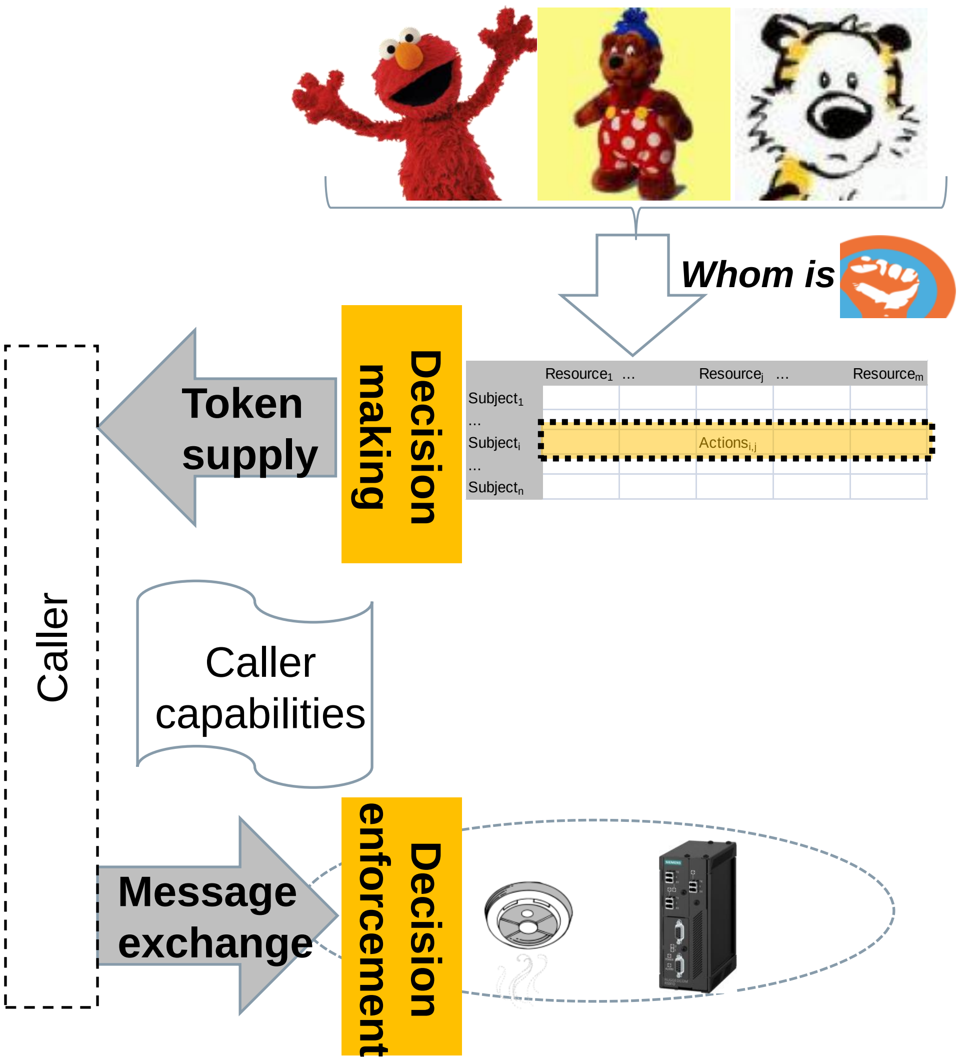
Who Is the Authority of Authorization?



- The **owner(s)** of an object are its root **authority of authorization**
  - This authority controls the contents of an access control matrix resp. its representation in implementation according provided tools
- Current practice is to understand and manage such authority in the case of digital goods
- Digital good basics (reproduction and relocation at almost no cost) allow to address the management of ownership in a trivial way:
  - The resource owner is always known at digital good creation time
  - Ownership of a digital good never gets transferred to another actor
  - Rather objects are cloned (exploiting reproduction at almost no cost) and the new object is assigned to a new owner

Challenges - Accommodate Physical Goods

Innovation Needs



- The current approaches **do not reflect the needs of physical goods.**
  - Change of ownership is commonplace in industrial IT. Sample scenarios:
    - Produce for an unknown customer, sell it
    - Produce for known customer who later sells it (possibly without informing manufacturer)
  - The digital goods approach to reflect and manage ownership (clone the item) just does not do the trick for physical goods
- Support of this use case is mandatory. Its elaboration must address legal concepts:
  - Legal entity-owned goods: proxy actors (managers/admins…) are commonplace
  - Individually-owned goods: proxy actors are an exception

Conclusions

So Who May Champion the Industrial Internet?

- **Industry and industrial IT**:
  - *Come from*: closed ecosystems utilizing proprietary mechanisms
  - *Prefer*: closed standardization bodies (IEC, IEEE, ISO…)
  - *Advantages*: champion industrial IT domain know-how, components and functionality
  - *White spot*s: lack experience with the supply and management of private resources (legal entity-owned) at public or partner-facing endpoints
  - *Threats*: disruptive innovations from outside the industry and industrial IT ecosystem
- **Internet and Web giants**:
  - *Come from*: open ecosystems with standards-based mechanisms
  - *Prefer*: open standardization bodies (IETF, W3C, OASIS, OpenID Forum…)
  - *Advantages*: champion the management of private resources (individually-owned) at public-facing endpoints
  - *White spots*: manufacturing of industrial products and their integration into solutions, reflecting the specifics of physical goods in IT-processes
  - *Threats*: inability to enter the IoT/WoT domain in case of investment goods aka Industrial Internet (did already enter this domain in case of consumer goods e.g. Google nest)

Takeaways

- Security for the industrial Internet presents a **challenge** for
  - Industry and industrial IT players
  - Web and Cloud giants – assuming they would want to enter the industrial Internet
- There will be **no one-size-fits-all** security solution for the industrial Internet
  - Constraints do vary too broadly across industrial Internet scenarios
- Security for the industrial Internet (resp. IoT/WoT and I4.0) is **no done thing:**
  - Innovations are needed e.g. for inclusion of RFC 7228 class 1/0 devices or means to reflect and manage device ownership
  - Further elaboration is also needed e.g. means to manage device authorization as an end user

Abbreviations

AMQP	Advanced Message Queuing Protocol
ASN.1	Abstract Syntax Notation 1
BYOI	Bring Your Own Identity
CBOR	Concise Binary Object Representation
CMS	Container-Managed Security
CoAP	Constrained Application Protocol
DICE	DTLS In Constrained Environments
DNF	Did Not Finish
DTLS	Datagram TLS
HTTP	HyperText Transfer Protocol
I4.0	Industrie 4.0 (German term)
ID	IDentity
IdP	Identity Provider
IIC	Industrial Internet Consortium
IMSI	International Mobile Subscriber Identity
IoT	Internet-of-Things
JOSE	Javascript Object Signature and Encryption
JSON	JavaScript Object Notation
JWT	JSON Web Token
MQTT	Message Queue Telemetry Transport
OAuth	Open Authorization
OIDC	OpenID Connect
PIN	Personal Identity Number
PKI	Public Key Infrastructure
SIM	Subscriber Identity Module
SSH	Secure SHell
SSL	Secure Sockets Layer
TLS	Transport Layer Security
UMA	User-Managed Access
WoS	Web-of-Systems
WoT	Web-of-Things
XMPP	eXtensible Messaging and Presence Protocol

SIEMENS

Literature

- J. Arkko et al.: *Architectural Considerations in Smart Object Networking*, IETF RFC 7452, March 2015; http://tools.ietf.org/rfc/rfc7452.txt
- T. Berners-Lee, *Information Management: A Proposal*, March 1989; www.w3.org/History/1989/proposal.html
- C. Bormann et al., *Terminology for Constrained-Node Networks*, IETF RFC 7228, May 2014; https://tools.ietf.org/html/rfc7228
- R. Fielding: *Architectural Styles and the Design of Network-based Software Architectures*, PhD thesis, University of California, Irvine, 2000
- S. Gerdes et al.: *Delegated CoAP Authentication and Authorization Framework (DCAF)*, IETF Internet draft, work in progress, May 2015; https://tools.ietf.org/html/draft-gerdes-core-dcaf-authorize-02
- T. Hardjano et al. (eds.): *User-Managed Access (UMA) Profile of OAuth 2.0*, IETF Internet draft, work in progress, May 2015; https://tools.ietf.org/html/draft-hardjono-oauth-umacore-13
- D. Hardt: *The OAuth 2.0 Authorization Framework*, IETF RFC 6749, Oct. 2012; https://tools.ietf.org/html/rfc6749
- M. Jones: *A JSON-Based Identity Protocol Suite*, Information Standards Quarterly, vol. 26, no. 3, 2014, pp. 19–22
- S. Yegge: *Stevey's Google Platforms Rant*. Oct. 2011; https://plus.google.com/+RipRowan/posts/eVeouesvaVX

Author

Dr. Oliver Pfaff, oliver.pfaff@siemens.com, Siemens AG, CT RTC ITS