PLATTFORM INDUSTRIE 4.0

DISCUSSION PAPER

Secure implementation of OPC UA for operators, integrators and manufacturers

Imprint

Publisher
Federal Ministry for Economic
Affairs and Energy (BMWi)
Public Relations
11019 Berlin
www.bmwi.de

Text and editing
Plattform Industrie 4.0
Bertolt-Brecht-Platz 3
10117 Berlin

Design and production
PRpetuum GmbH, Munich

Status
April 2018

Printed by
MKL Druck GmbH & Co. KG, Ostbevern

Photos and illustrations
BlackJack3D – gettyimages (title), sdecoret – Fotolia (p. 6), patpongstock – Fotolia (p. 8), zapp2photo – Fotolia (p. 9), NicoElNino – Fotolia (p. 13), Gorodenkoff – Fotolia (p. 21), Sikov – Fotolia (p. 23), kras99 – Fotolia (p. 24)

FSC®
www.fsc.org
MIX
Paper from
responsible sources
FSC® C108626

This brochure is published as part of the public relations work of the Federal Ministry for Economic Affairs and Energy. It is distributed free of charge and is not intended for sale. The distribution of this brochure at campaign events or at information stands run by political parties is prohibited. No political party-related information or advertising may be inserted in, printed on or affixed to this publication.

This and other publications are available from:
Federal Ministry for Economic Affairs and Energy
Public Relations
E-mail: publikationen@bundesregierung.de
www.bmwi.de

Central procurement service:
Telephone: 030 182722721
Fax: 030 18102722721

Content

Introduction

The innovative concepts and procedures of the Fourth Industrial Revolution (Industry 4.0) are creating entirely new possibilities for cooperation – especially on a technical level. Systems, machines and products interact, exchange data and information, communicating with each other all the while. It makes no difference whether communication takes place with a machine in the same factory hall or with a system in a plant on the other side of the world so that the boundaries for trust are exceeded. However, this can only work if *technical communication mechanisms* ensure that Industry 4.0 components (assets) can communicate in a secure and interoperable manner (1) and thereby enable trust across company boundaries.

The importance of OPC UA

OPC UA (OPC[1] Unified Architecture) is an architecture used to describe and exchange machine data. In this respect, OPC UA is more than just a communication protocol – the architecture also includes data models and interaction concepts. OPC has been successfully used in automation technology for some time now. The further development of OPC UA is today widely supported; it has been recommended as an important technology in the implementation strategy of the Industry 4.0 platform (2) and is part of the "Criteria for Industry 4.0 Products" of the ZVEI manufacturers' association (3). This paper focuses accordingly on OPC UA.

OPC UA in M2M communication

Work to standardise machine-to-machine communication has been underway for several years and at times parallel at various organisations, such as the OPC Foundation, IETF, oneM2M, OASIS, NIST, ITU, to mention but a few. This has resulted in different architectures, protocols and security concepts with different functionalities.

Consideration of security requirements

Software and network communications are already being extensively used today for tasks with increasingly more open security domains in automation. That's why security[2] aspects must also be taken into account in order to meet the resulting protection requirements. Although no new threats are generally expected, the now necessary opening of perimeter protection, i.e. protection on the perimeter of the security domain, means a larger attack area. When it comes to critical infrastructures, operators are therefore now required to take security measures to ensure provisioning. In industrial automation, security awareness also grows as network-based communication increases. With the goal of smart production in Industry 4.0 and the associated communication between IT, production, assets, components and products, security is an elementary part of concepts, as is also described in the implementation strategy (2).

Established standards and norms describe the technical and organisational measures that form the basis for secure use, see the section on "Security". The requirements of these standards and guidelines must be implemented in practice so that secure application of the OPC UA standard is possible. The OPC UA standard offers many solution concepts and ideas. What's important now is to describe how the individual aspects have to interact in order to achieve the goal of secure use.

Content and aim of this discussion paper

The aim of this discussion paper is to highlight the requirements for the secure use of OPC UA for communication in Industry 4.0 scenarios, to present implementation options and to identify points for discussion. The integration of a machine into an operator's infrastructure over its lifecycle will be used as an example.

[1] OPC: Originally "OLE for Process Control", today "Open Platform Communications"

[2] Always short for IT security in this document

The aim is to provide the stakeholders, manufacturers, integrators and operators involved with specific information about the necessary functions and measures and to describe best practices. At the same time, the analysis should show the extent to which the implementation of the security measures will lead to even more far-reaching requirements that call for additions to the OPC UA standard or to existing toolkits and products. The goal here is to cover as many aspects as possible with OPC UA only, so that no further requirements have to be fulfilled, for instance, by an additional interface such as web-based management. Therefore, both configuration and parameterisation must have a uniform design across corporate boundaries. This approach will improve consistency and interoperability.

This paper is based on the OPC UA standard, version 1.04, which offers significant further developments, especially for security. However, it can be assumed that the implementations and development tools available on the market are not yet at this level. One aim of this paper is to support providers and users during the transition.

This paper is meant for technically experienced readers, ideally with experience in the application of OPC UA.

Security

Security is a holistic issue that can only be achieved when all stakeholders work together. The relevant standards for industrial automation, IEC 62443 (4) and the German VDI 2182 (5), therefore always consider interaction between operators, integrators and manufacturers.

Information security management

The security requirements for secure operation must be based on the operational framework, see also "IT-Security in der Industrie 4.0 – Handlungsfelder für Betreiber" (6). Corresponding information security management systems (ISMS) are described in ISO 27000 (7) and IEC 62443-2-1 (4). Which threats exist for data and systems can only be determined specifically, depending on the application. To achieve a multi-stakeholder approach, as discussed later and shown in Figure 2, the requirements of all stakeholders must be taken into account and this can lead to conflicting goals.

Threat analysis

Identifying a company's assets to be protected forms the basis for the threat analysis. In the case of an operator, these typically include know-how, system availability, production efficiency and product quality. The document "Integrität von Daten, Systemen und Prozessen" (8) addresses the frequently underestimated importance of the integrity protection goal as a prerequisite. Once the company assets to be protected have been identified, threats, such as loss of know-how or production disruptions, can then be described.

Protection goals and guidelines

Once the threats have been identified, the protection goals are formulated accordingly and measures are taken based on the severity of impact and the assumed probability. The primary protection goals are:

- **Confidentiality:** Protection against unauthorised disclosure of information

- **Integrity:** Ensuring the correctness (integrity) of data and the correct functioning of systems

- **Availability:** Services, functions, information can always be used as intended

Further protection goals are formulated, for example, based on data protection aspects, such as the European General Data Protection Regulation (EU GDPR). In technical terms, these requirements can be mapped to the primary protection goals. This document restricts itself to the primary protection goals related to communication processes. It does not examine further implementation, for instance, in the form of data storage in a device.

When it comes to selecting and implementing measures, the options available often have to be weighed up. Encrypted communication protects against eavesdropping (protection goal: confidentiality), but makes troubleshooting (protection goal: availability) and monitoring of communication more difficult. Therefore, security measures should be selected according to the given requirements. In an internal automation network with less powerful components, encryption may not be necessary as integrity protection is technically possible independent of this. Confidentiality is additionally relevant when information is exchanged using unprotected networks. Secure communication protocols typically offer the "Integrity protection" (with OPC UA: "Sign") and "Confidentiality Protection + Integrity Protection" (with OPC UA: "Sign and Encrypt") options.

Detection and response

In security management, it can be generally assumed that 100% security is simply not possible. In addition, mechanisms must be in place to detect attacks, such as event logging and communication inspections, along with emergency and recovery concepts.

Component and system security

Various parts of IEC 62443 (4) describe the requirements for security functions, such as user management, integrity protection, secure storage of electronic keys and logging, as well as requirements for processes in the integration and development of components. The security functions are classified as “Security Levels”, from SL-1 to SL-4, which are designed to express the system’s strength of resistance. The security level is also determined in the threat analysis. It is important to bear in mind that security not only means the existence of functions, but also requires the application of development and integration processes that have been designed on the basis of security aspects.

NAMUR recommendation NE 153 (9) concisely describes the four aspects of component and system security:

- **Security by Design:** Security must be included in the design stage
- **Security by Implementation:** Security as a feature by avoiding errors as far as possible
- **Security by Default:** A secure status should always be the basic setting, no retroactive hardening
- **Security in Deployment:** Secure operations through security documentation and product maintenance

Application scenario

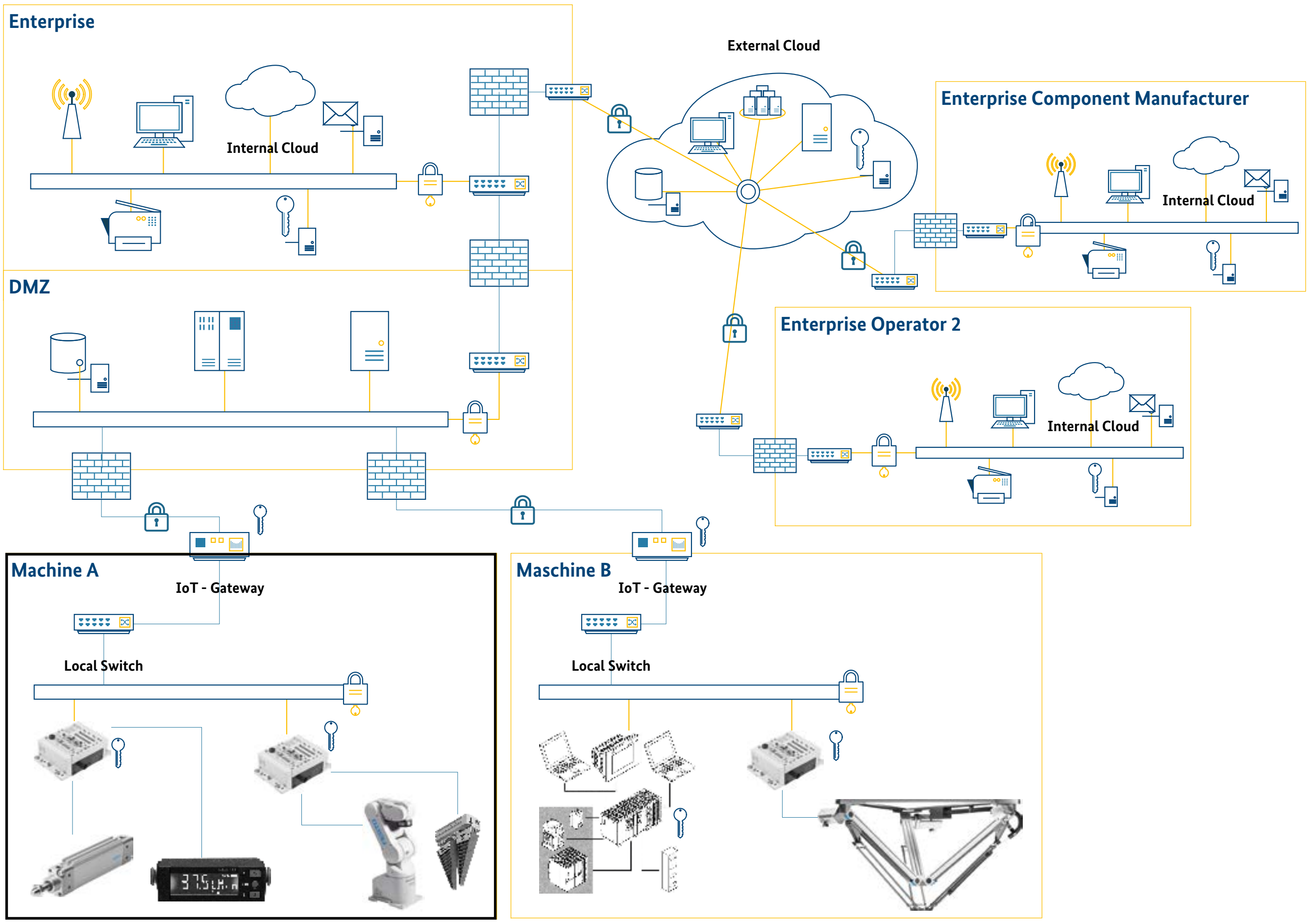
The “Collaborative Factory” scenario is used to illustrate such an application. This Industry 4.0 scenario shows the integration of different machines into a factory with connections to cloud solutions and other external companies (Figure 1). An overview of the application scenario can be found in the appendix. Communication within a company-wide network poses a multitude of demands for secure design. These requirements and approaches are already discussed in the technical overview entitled “Sichere unternehmensübergreifende Kommunikation” (10).

For the purposes of this document, only the integration of machine A into an operator’s infrastructure is to be examined, see box in Figure 1. The machine is seen as a unit that must interact with its environment. The machine design is irrelevant here.

Figure 2 shows the logical communication relationships of the machine. On the one hand, the machine must be integrated into the production process and hence interact with the operator’s systems. It must be possible to not only manage production orders but also to collect operating data and handle alarms. This communication relationship, marked with a green arrow in Figure 2, is the focus of this discussion paper.

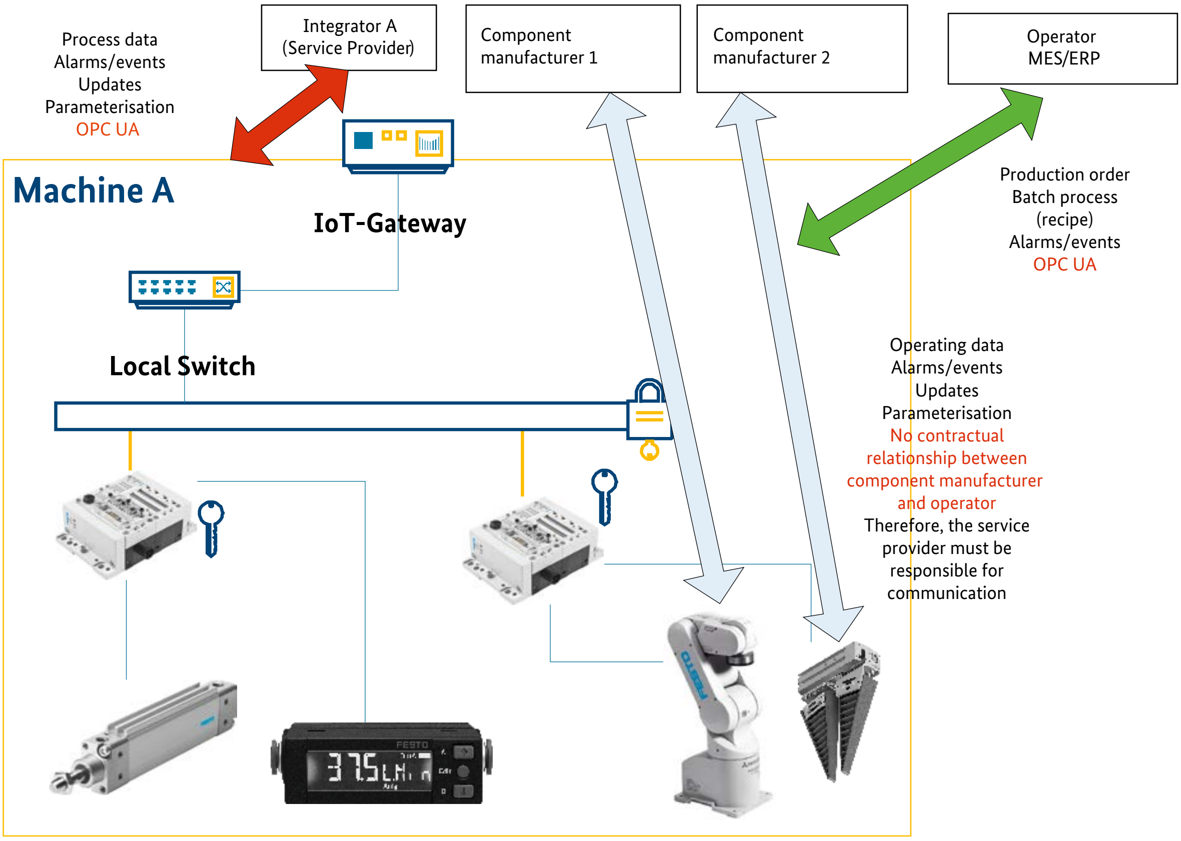
In an extended examination, interaction between the machine and an external service provider must be taken into account, which is indicated by a red arrow in Figure 2. This service provider could be the integrator or machine designer himself, who would only access the machine for remote maintenance purposes (“condition monitoring”) or, in the operator model, could even actively parameterise the machine.

Figure 1: Overall “Collaborative Factory” scenario



Source: Plattform Industrie 4.0

Figure 2: Logical interfaces of machine A



Source: Plattform Industrie 4.0

This document, however, does not examine the special features that result from this, such as possible requirements for monitoring of communications by the operator. Other options, such as individual communication between individual machine components and their manufacturers, are not examined either.

The discussion in this document refers to the logical communication relationships, i. e. information exchange. The underlying transmission technologies used (wired, wireless, short or long distance) were not considered.

Application of OPC UA in the scenario

It is widely expected that OPC UA will play a key role in the connected industry, since the exchange of parameters, operating data and alarms is one the main features of OPC UA.

Therefore, OPC UA is to be used in this paper to integrate a new machine into the operator's systems. The machine is integrated within a local network controlled by the operator. This document focuses on this local integration which targets the operator's security domain.

The logic connection to the external service provider is a remote connection. It is possible here that communication may run physically through the operator's network and then via the Internet to the service provider. This enables efficient use of existing resources. This design also allows the operator to monitor and influence communication. A dedicated connection via separate Internet connectivity is also conceivable, with would reduce interaction in the operator network. In any case, communication is established between the operator's security domain and that of the service provider. That's why the security measures of the stakeholders must be coordinated and taken into account in the respective security management system, see technical overview entitled “Sichere unternehmensübergreifende Kommunikation” (10). This communication relationship will be addressed in a future follow-up paper.

The following section restricts itself to discussing the secure use of OPC UA. The quality of the implementation, for example, through a secure development process (Security Development Lifecycle, SDL) and other security functions are not included here.

Lifecycle

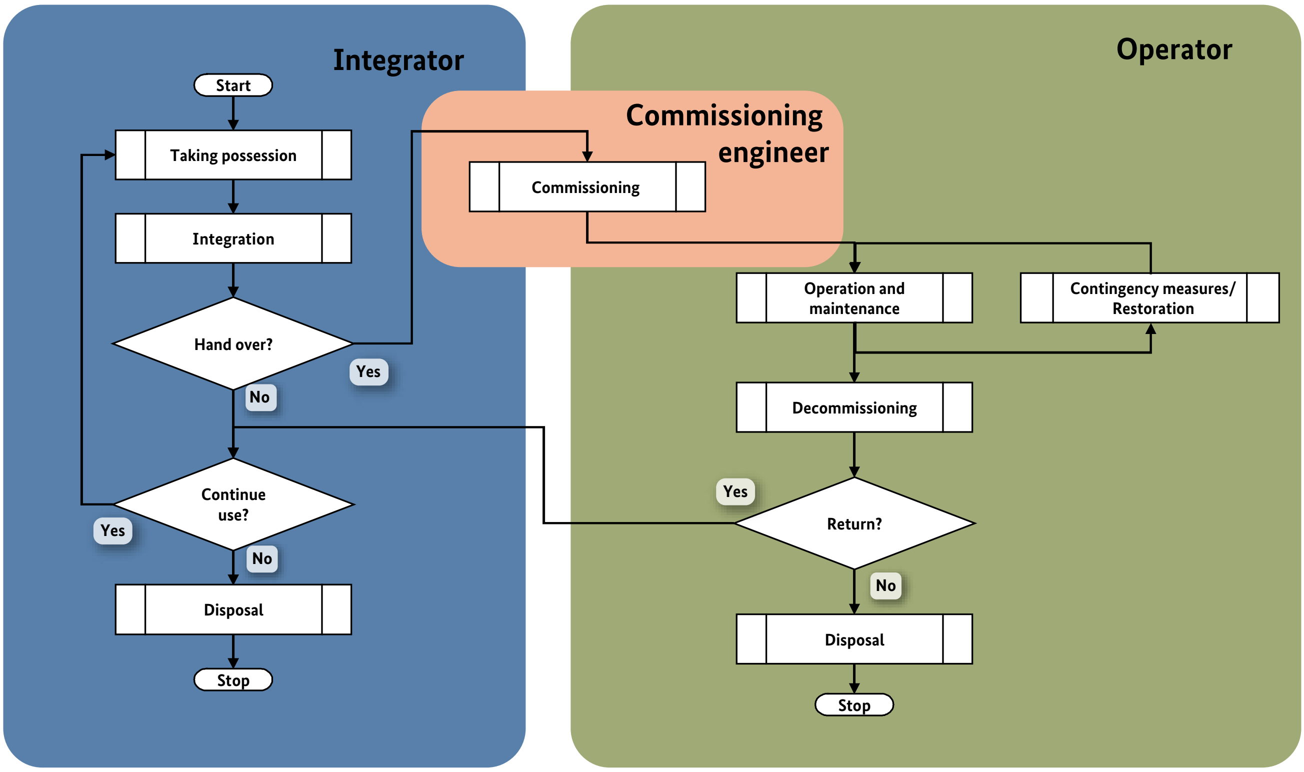
In order to be able to analyse the stakeholders’ security requirements for the machine to be integrated, the lifecycle is examined, beginning with the provision of the necessary components and ending with the decommissioning of the machine (see Figure 3). The phases explored here are:

- Provision and ownership of a component
- Integration of several components into a system (e.g. a machine or (sub-)system)
- Commissioning of the system by the operator
- Operation and maintenance
- Decommissioning of the system or system components
- Disposal after final decommissioning

In principle, the analysis is first generic, since the stakeholders’ specific protection goals are not known. All possible requirements must therefore be examined at the highest security level in order to ensure that all the necessary protection goals can be achieved. In addition, the requirement level is looked at independent of the technologies used.

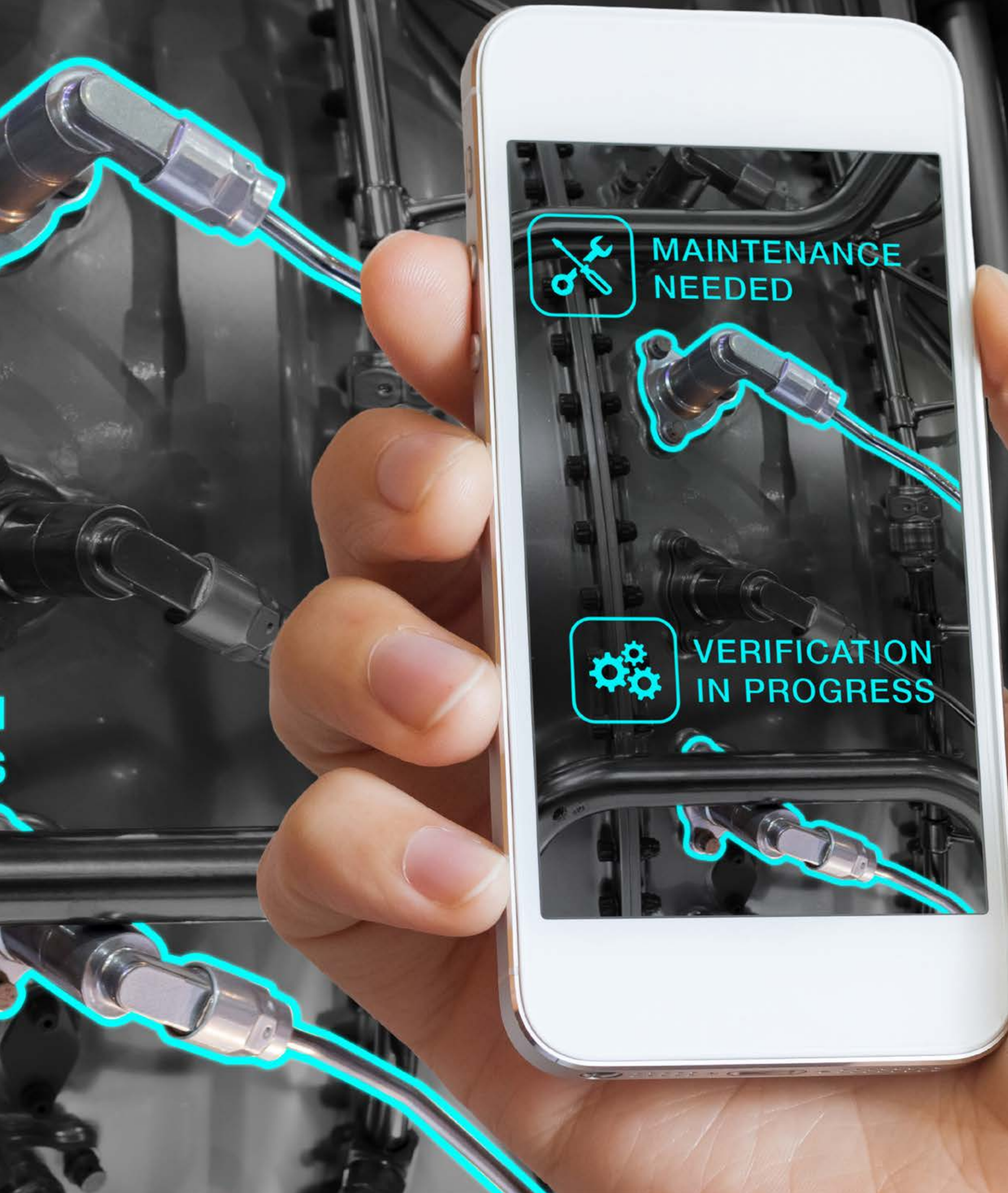
An analysis of the first part of the lifecycle, taking possession and integration of components into a system, can already be found in the paper entitled “Security der Verwaltungsschale” (11).

Figure 3: Lifecycle phases



Source: Plattform Industrie 4.0

Examination over the lifecycle



The security requirements for a component are examined on the basis of the lifecycle phases described above in addition to a section for contingency and restoration measures.

At the end of each phase, the security requirements identified in the document are summarised and repeated in abstract form.

A solution is subsequently sketched highlighting both the requirements that can already be fulfilled with means according to the OPC UA standard and/or sensibly using other customary measures as well as the areas where the need for discussion becomes apparent.

During the discussion on the lifecycle, the need for identities from different sources already becomes apparent. In order for them to be kept apart, the sources are described here first, i.e.:

1. identities issued by the component manufacturer (e.g. ZH_N manufacturer certificates),

2. identities assigned by the integrator (e.g. ZIN certificates),

3. optional identities valid in the system (e.g. ZA_N certificates) and

4. identities issued by the operator (e.g. ZB_N certificates).

Taking possession

Taking possession of a component is understood to be the first-time use by an integrator or directly by the operator. The integrator takes a generic component and configures it to gain exclusive control over it. This is carried out, for example, by replacing default passwords or uploading integrator certificates. In abstract terms, these are the identity information and authentication criteria used by the components to verify the identity of their communication partners.

First-time use of a component can refer to a brand new component or to a component that has been reset to factory settings (and is therefore equivalent to a brand new component in terms of its configuration).

First of all, it must be ensured that the component is an original component and that neither hardware nor software (including firmware) have been manipulated. While the authenticity of hardware can be verified on the basis of its physical features, such as holograms, software and firmware can be checked on the basis of code signatures.

In addition or alternatively, the authenticity of the components and their firmware can also be verified via the network. This enables work processes in which different persons are responsible for physical assembly and taking possession. Each component should have an individual certificate issued by the manufacturer. By using the pertinent private key, the device then confirms that its state is the state that has been defined as authentic by the manufacturer. There are two ways to verify whether this certificate is valid: 1.) If the certificate issued by the manufacturer is issued by a root Certification Authority (root CA), the entire certificate chain must be checked. 2.) If the certificate is individually self-signed, the manufacturer must provide the certificate fingerprint separately for comparison. This fingerprint must reach the customer via a different channel than the component itself, e.g. by publishing the fingerprint on the manufacturer's HTTPS secured website. Sending the fingerprint in the instructions for the component or in test software on a CD included in the scope of delivery is not secure enough and an attacker could change both (device and documentation/test software) during delivery.

Hybrid forms of these two variants are conceivable for verifying the certificate issued by the manufacturer. For example, to save costs, the manufacturer himself can operate the root CA for the device certificates issued by him. In this case and using a separate channel, the manufacturer only needs to provide the fingerprint for the root certificate created by him. The fingerprint is then the same for many devices, for example, all the devices of a series or even all of the manufacturer's devices.

The private key that belongs to the public key of the certificate must be kept locked. Ideally, this is carried out using secure hardware, a so-called "secure element". All operations on the private key then take place in the secure environment of the "secure element".

To ensure that no manipulation has taken place here, the integrity of the "secure element" should be checked, if possible, when it is taken into possession. There are also procedures for this that can be carried out via the network.

Even if there are reasons not to use a secure element, the use of asymmetric key pairs of public and private keys offers added security. If a component does not even have the resources to use asymmetric key pairs, other components can be used as relays. In a small, relatively separate area, for instance, a relaying component can use asymmetric key pairs and communicate on behalf of the other components that have no key pairs. In this case, communication with the other components beyond this point should be carried out via a medium with suitably restricted access.

An important part of a component's security is its secure configuration. When a component is delivered, its default configuration should be parameterised as securely as possible in order to prevent attacks during taking possession or insecure subsequent configurations due to operating errors. This is also known as "Security by Default". For OPC UA communication, for example, the None security policy would have to be excluded, i.e. it should not be offered. If the highest security level is not required for an application, the component should ensure that the change in configuration is subject to authorisation which in turn requires a graded rights access control concept. It must also be possible to reset the configuration of the component to (secure) factory settings.

The security requirements for a component are therefore as follows for taking possession of a component:

1. It should be possible to verify the authenticity of the component on the basis of a certificate from the manufacturer.

2. It should also be possible to reset all component settings to the manufacturer's factory settings.

3. The basic configuration should have a secure design and the device should be delivered with a secure configuration, making an attack during commissioning unlikely.

4. The integrator should be able to define all of the authentication criteria used by the component to verify other identities. These include all passwords and all certificates that the component trusts, except for a few certificates that are intended by the manufacturer for special purposes, such as authentication of firmware updates.

Integration

After possession has been taken of the individual components, the integrator combines them to produce an overall function. The integrator establishes both a logical and a physical relationship between the components and models the behaviour of the individual components required for the overall function. The processes in this step are especially relevant for security, since both the relationships between the individual components and the function of the individual components are decisive for the resultant secure and correct operation of the machine or system. In particular, the following particularly security-critical processes can be identified during the integration phase:

a. Definition of the digital identities of the respective individual components.

b. Modelling and implementation of the relationships between the component and other entities, such as

    1. other devices and software processes inside and outside the machine or system and

    2. persons acting in a certain function.

c. Setting (parameterisation) of access control mechanisms according to the model of relationships between the component and other entities.

d. Control of access to internal device features and functions with the aim of protecting business secrets which the integrator brings to the machine or system.

When relationships are established between individual components and other entities, both the functions of the devices and the resulting access rights of individual devices and software components must be defined in relation to each other. Unambiguous and reliable component identification and referencing is an important basis for modelling component relationships. This can be based on secure digital identities (e.g. digital certificates and hardware support). A digital identity can be assigned to a component either via a public key infrastructure (Certification Authority and issuance of certificates) or by manually configuring certificate-based identities on the respective components. It should be noted that the identities are assigned either within the system itself and/or by the integrator. If the identity of the manufacturer were used to establish the relationship, this might enable an attacker to later infiltrate the system by replacing the device with a device purchased and parameterised by the attacker that also has a valid manufacturer's certificate.

Once the devices have been given a secure identity (ZI_N certificate including public/private key), trust relationships between these devices can be modelled. This is carried out by including the identity of one device in the trust list of another device. Alternatively, the identity of a Certification Authority (CA), which can authenticate other digital identities, can be included in a trust list of a component. In the latter case, all identities issued by the CA are trusted and this reduces the configuration effort needed because the certificates to be checked no longer have to be explicitly included by other devices. Direct inclusion of identities in the corresponding trust lists or inclusion of a Certification Authority determines the components which the integrator basically provides for interaction. Each component should be configured to interact with a minimum number of other components. This reduces the possibilities for an attacker after a single component has been compromised.

After the permitted communication relationships have been configured, the access rights of the respective components must be modelled further. The integrator will restrict access to individual values of a component for other components or other users in order to prevent attacks, misuse or disclosure of the integrator's company secrets. Access to component functions and data should be designed in such a way that each user (other component or user) only receives the minimum level of access rights necessary to perform their function. The integrator sets these access rights and in many cases will reserve the exclusive right for himself to adjust access rights and restrictions and to assign access rights (to a component or person). This is necessary because otherwise the integrator cannot impose any restrictions to protect the function of a machine or to protect his own company secrets (e.g. the exact configuration of components and their interaction). The integrator can grant further rights to the future operator, so that he can operate the machine as intended or integrate it into his own company infrastructure. In this scenario, this means that the integrator gives some of the operator's users or components read access to certain values and alarms, so that the machine can be monitored. In addition, the operator can have write access to certain machine parameters, so that they can be adapted to the products to be manufactured.

With more complex systems and machines where a large number of access rights must be configured similarly or identically for other different components or persons, role-based or attribute-based assignment of rights makes it easier to manage these access rights. This means that rights can be defined for a role (e.g. maintenance staff, operator, monitoring, etc.) or for the owner of attributes (e.g. affiliation to the company and responsibility for maintenance). What's special here is the implementation of the rights specification without already defining the specific digital identity in the integration phase. Role affiliation or attributes must then be assigned to an identity using suitable measures during operation (e.g. based on a central authentication system in the operator's environment). The integrator can then provide for certain interaction patterns which the operator later assigns to specific individual identities.

The security requirements for integration of a component are therefore as follows:

1. It should be possible to assign an integrator-specific identity with a pertinent ZI_N certificate to the component.

2. For the purpose of communication in the system, the component should:

   2.1 be given and be able to use a system-specific identity including a ZA_N certificate (which may be, but does not need to be the integrator-specific identity) and

   2.2 be given and be able to use a system-specific trust list.

3. For the purpose of communication with the integrator's processes and staff, the component should be given and able to use the integrator-specific identity, including the ZIN certificate, as well as an integrator-specific trust list.

4. The component should support an access control mechanism that can be used to define rights independent of specific identities.

5. The rights in the component should be set so that certain rights are required in order to change rights.

6. The rights in the component should be set so that certain rights are required to set the rules for assigning rights to identities.

Commissioning

A transfer of risk takes place during commissioning. The system created by the integrator moves from the integrator's sphere of responsibility to that of the operator. Commissioning can be divided into two phases: In the first phase, the integrator prepares the takeover by the operator. In the second phase, the operator takes over the system. This part of the takeover by the operator is often also accompanied by the integrator's staff. In this case, staff performing the takeover by, for or with the operator are referred to as "commissioning engineers." This happens irrespective of whether these staff are now assigned to the operator, are contracted by the operator or whether they are the integrator's staff accompanying the operator's staff. The integrator prepares the commissioning of the system and takeover is carried out by the commissioning engineer.

It is not just the responsibility for the system that changes after commissioning. This often also means a change in possession (in this scenario, however, with no change in ownership). This primarily changes the integration of the system into the security domains. It ultimately determines to a certain extent who or what can accesses the components and vice versa, and who or what the component can communicate with. Other parts of these rights determine the access control mechanisms already set during integration along with options for assigning rights.

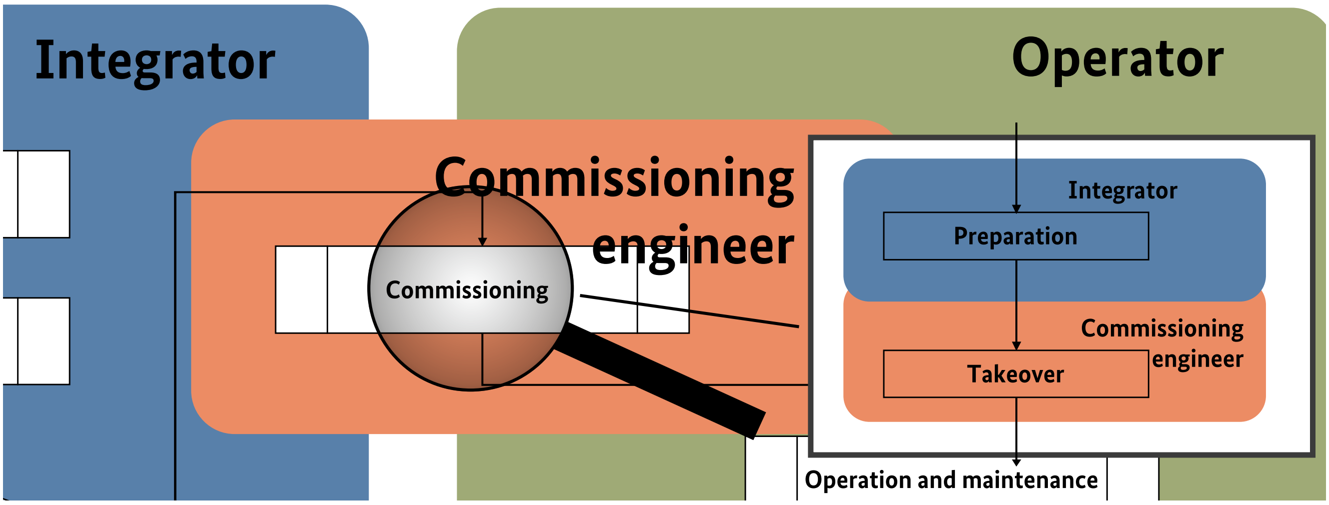
Integrator's preparation of handover

Preparation for handover to the operator is carried out by the integrator.

For the example scenario, handover is prepared in the following steps and in the following order:

- The integrator defines access rights for maintenance staff and maintenance processes in the form of additional rights and assignment rules. Specific assignments of rights are also activated, for example, by assigning sets of identities with concrete rules to roles. In some applications, it is necessary that these rights also include the right to update the system's software and firmware at the beginning of commissioning. Delivery may have taken weeks, so that in the meantime updates may be available for software and firmware. The resulting complexity of rights – and how their definition survives a software and firmware renewal – is not discussed further in this paper, but remains a task for a successor paper.
- The integrator stores and activates authentication criteria and rights in the components of the system for the commissioning engineer, so that he is recognised by the system during takeover.

Figure 4: Breakdown of commissioning into two phases



Source: Plattform Industrie 4.0

  - In many cases, it must be possible for the system to verify these authentication criteria without the need for functioning integration into an IT network, because the system will not yet have been integrated into such a network at its new location.

  - Furthermore, the work of the commissioning engineer is once-off and temporary. This work is completed when the system has been taken over. The integrator rather than the commissioning engineer is responsible for maintenance according to the model here.

    This means that, according to the principle of assigning need-to-know rights, the special rights assignments for the commissioning engineer can be deactivated once the work has been completed. His authentication criteria and rights assignments should therefore only exist temporarily in the system.

- Finally, the integrator removes unnecessary authentication criteria as well as access rights and options which he no longer needs after handover.

- If necessary, the integrator has granted the system and/or its components access possibilities and rights in the integrator's security domain which were necessary during component integration, for example, for a test operation. The integrator will therefore block or delete unnecessary access paths and rights for the system in his security domain. The integrator will not delete the identities and pertinent ZIN certificates issued by him, nor will he revoke them, instead, their access options will be reduced. These identities could be useful for remote maintenance.

Preparation for commissioning therefore results in the following security requirements for a system with Industry 4.0 components:

1. It should be possible in the system to adjust the authentication criteria, rights and rights assignments (to roles or access rules) defined by the integrator for maintenance access.

2. The integrator should be able to activate authentication criteria and rights in the system that are temporarily needed for the commissioning engineer, so that the system can authenticate the commissioning engineer, if required even without a network connection, and it should be possible to remove these rights assignments and authentication criteria again for the commissioning engineer.

3. The integrator should be able to delete unnecessary access paths, authentication criteria and rights from the system.

Takeover during commissioning

While preparation for commissioning can take place before shipment of a system, actual takeover in this example will take place after shipment and physical installation of the system on site.

The commissioning engineer's task is to take over the system from the integrator and to commission it at the operator's site in such a manner that the operator can subsequently operate the system for his benefit in regular operating mode (only to be interrupted by maintenance, if any). For this purpose, the commissioning engineer connects the system to the local environment at the operator's site. In addition to physical connections, this also includes connection to the operator's IT, and more importantly, inclusion in certain security domains:

- First of all, the commissioning engineer checks the authenticity and integrity of the system in order to determine whether the system is from the integrator and whether it is in the state in which it was prepared by the integrator. This is particularly necessary if the system or parts of the system were under the control of third parties in the time between preparation and takeover, for example, by a forwarder and its contractor.

- If the system is trusted, the commissioning engineer brings identities created by the operator together with ZB_N certificates into the system to enable secure interaction with the operator's security domain.

- The system and its components are incorporated in two stages into the security domain of the operator's IT.

  - Authentication criteria provided by the operator are incorporated into the system and its components.

  - The identities of the system and its components are made known in the operator's infrastructure by activating the associated certificates. The system is now set up so that it can renew the ZB_N certificates if needed, for instance, in good time before validity expires. The system can also receive up-to-date versions of the authentication criteria issued by the operator. From time to time, the operator may have to introduce new trusted root certificates (root CA certificates) into operations or distribute revocation information if other components are taken out of operation before their certificates expire.

- The commissioning engineer sets the authentication criteria in the system for the operator's personnel and processes. The operator must especially remember that roles, attributes and their characteristics may have completely different names or designations in the operator's security domain than anticipated by the integrator. A series machine manufacturer, for instance, may not find any common denominator for the designations used by his customers. This means that it must be possible to change the designations provided by the integrator within the access control mechanisms of the system to the actual designations. Mapping rules from actual (external) to logical (internal) designations are useful here (external means outside the system and internal means inside the system).

- The commissioning engineer tests remote maintenance access together with the operator and the maintenance staff working remotely. In doing so, he also explains to the operator the access path and the rights that can be exercised via this path. This does not mean that a permanent and unobserved maintenance option is activated here, but that personnel can be authenticated and authorised in the case of maintenance. How maintenance access will in fact have to be later enabled by the operator, for instance, using a key switch, depends on the purpose of the system. For some systems, permanent monitoring by the integrator or a service provider may be desired, for example, as part of predictive maintenance or condition monitoring. As already mentioned above in the explanation of the application scenario, the auditability of actual maintenance access is an issue for some operators. But its safe implementation will be the subject of discussion in a later paper.

- Once the system has been accepted by the operator, the commissioning engineer deletes his access options, which are now no longer required.

Takeover results in the following security requirements for a system with Industry 4.0 components:

1. It should be possible to check that the system comes from the integrator and is in the state defined by the integrator.

2. The operator's identities together with the ZB_N certificates issued by the operator can be introduced into the system, while the previous identities and the pertinent certificates (e.g. certificates of the ZI_N integrator and certificates of the ZH_N component manufacturers) remain in the system.

3. It should be possible to store authentication criteria for identities of different security domains separately and simultaneously in the system.

4. It should be possible to both set and at the same time activate rights with reference to the authentication criteria of the identities of a security domain, so that the system can distinguish between the operator and the integrator (for maintenance) and enforce the corresponding rights.

5. It should be possible to map the descriptions actually used by the operator (external roles, external attributes and their characteristics) for the identities of his personnel and his processes to designations defined by the integrator (internal designations).

6. It must be possible to delete the access rights and authentication criteria temporarily set in the system regarding the identities for access by the commissioning engineer.

Operation and security maintenance

During the operating phase, the private keys and certificates for authentication for OPC UA must be changed at regular intervals in a secure manner, for instance, if technical progress or attacks constantly impair the security of cryptographic algorithms. A change can be planned when cryptographic ageing of methods and algorithms is foreseeable, whereas an attack in which private keys, for instance, were stolen is reason for a fast and unscheduled exchange.

A fundamental distinction must be made when changing private keys and certificates:

- A component/user receives a new private key and a new certificate must be generated.
- A component/user has received a new certificate and the certificate is to be authorised (for example, to communicate via OPC UA). In the best-case scenario, the authentication criteria (trust lists) of the other components do not have to be updated. In some cases, it is necessary to store the new certificate in repository services or even to distribute an associated new issuer certificate (sub-CA certificate) to other components via a distribution mechanism.
- The issuing Certification Authority is changed and all components must be informed of this. This means that certificates from several certification bodies may exist during the period of transition. The components must support and accept this.

In the case of certificates for components, a distinction must also be made between two types of holders. The certificates can come either from the operator or from the integrator. Both are responsible for their respective certificates and must exchange them on the basis of their validity. Access rights must be defined for the exchange. For this purpose, the component must be able to assign rights for certificate renewal to different user groups. The same applies to the renewal of the key pairs that belong to the certificates. Responsibility for the certificate determines who can initiate the renewal of the key pair.

User authorisations can change over time. That's why it should be possible to change the authorisations used by the components or to change authentication servers. Again, a distinction must be made between the different user groups of the operator and the integrator, because access authorisation may not be mutually overwritten. It should even be possible to change user group assignments over time, because changes in personnel responsibilities due to changes in organisational structures or changes in the operator or integrator's technical infrastructure will necessitate changes in processes.

When private keys and certificates are exchanged, it must be taken into account that some systems or machines have only limited maintenance windows. This should therefore be carried out early or it should be possible without interrupting system operation.

If a connection to the system or its components is established from the operator's area or vice versa, the system should prove its affiliation to the operator with a ZB_N certificate. The system should also check the certificate of the remote station using the operator's criteria, for example, a trust list containing certificates of the operator. A secure connection is not established until mutual verification has taken place. If the system or its components establish a connection to the integrator or vice versa, for example, for maintenance purposes, the integrator's certificates and criteria must be used analogously. These rules apply especially to connections in which the keys and/or certificates are regularly renewed.

One principle of security practice is to minimise risks by using different key pairs for different tasks. For example, if a key pair with a certificate is used for confidential (encrypted) communication with a component, the same key pair should not be used to authenticate the component or for signatures to be generated by the component, see also Table 7, No. 4 in "Sicherheitsanalyse OPC UA" issued by the Federal Office for Information Security (12). Various risks are thereby reduced, which are justified both in security procedures and in organisational applications. For example, if the same key pair is used for confidential communication and authentication, some authentication methods can cause the key holder, i. e. the component, to decrypt for others confidential material intended for the key holder. Some authentication procedures require the component to be able to decrypt a random number that is unknown to the component but encrypted with its public key. However, if the attacker does not reinvent an encrypted random number but selects another confidential and encrypted message intended as a task for the component, decryption will then be delivered to the attacker virtually free of charge during the authentication process.

In security practice, using different key pairs for authentication and negotiation of symmetric keys for encrypting messages would also support the use of so-called middleboxes at trust boundaries in such a way that communications can be read by key deposit measures or sub-CA instances in a middlebox without this also compromising the authenticity of communications.

That's because middleboxes would only require keys or sub-CA instances which are stored for decryption. They would not have to be trained for authentication and hence falsification of messages. The auditability of data traffic across trust boundaries will be the subject of a future paper which will address this circumstance once again.

Regular operations with maintenance results in the following security requirements for a system with Industry 4.0 components:

1. In the case of identities and the pertinent certificates as well as key material,

   1.1 it should be possible to renew them without interruption and

   1.2 depending on the issuer, it should be possible to renew them on different paths (integrator certificate versus operator certificate)..

2. Authentication and authorisation criteria for identities should be

   2.1 renewed regularly and

   2.2 separately for each person responsible (integrator versus operator).

3. When establishing a connection to the system or its components or in the opposite direction, it should be possible to select

   3.1 which identity and which certificate are relevant (integrator versus operator) and

   3.2 which verification criteria are relevant for verifying the remote station and its users.

4. Different keys and certificates should be used for encryption and authentication/signing.

Decommissioning

Industry 4.0 components and systems contain sensitive data, such as keys, access data and confidential information in log data. Sensitive data therefore includes not only data relevant to security, but also data subject to data protection. If sensitive data falls into the wrong hands, this will pose a threat to all communication partners, because the data can be used to trigger actions and other confidential data may be captured and security settings may even be changed. If a device is compromised or stolen and if its sensitive data has not been protected by special hardware measures, this can be even more dangerous.

While the identities of lost components must be blocked in the event of sudden loss (theft), sensitive data must be deleted at the time of decommissioning. Both are described in general terms in a separate security policy: an end-of-life policy. This policy must be defined in line with the application. This means that there may be specific guidelines for specific types of systems and/or components.

Security guidelines and the security procedures derived from them must define the steps to be carried out in order to securely take a device out of service or disable it after it was lost. Decommissioning or disabling can be either permanent or temporary. It may be designed to securely erase all sensitive data from a device in order to recycle it in another context or to dispose of it in a secure manner. Furthermore, it must be defined how the device can be replaced in the event of a defect so that system functionality can be guaranteed and the information on the device deleted.

The ability of software to evade attacks decreases over its lifetime as new threats are discovered or emerge due to technological progress. When systems or devices are replaced, a new risk and hazard assessment should be carried out in order to check whether the security decisions previously made are still sufficient or whether stronger safeguards may be required.

Decommissioning results in the following security requirements:

1. Components should be able to securely delete sensitive data.

2. In both the operator and the integrator's infrastructure, it must be possible to disable access partially or completely for components or entire systems.

3. In the case of temporary disabling, it should be possible to activate and revoke disabling in the operator and/or integrator's infrastructure.

Disposal

During disposal, it must be ensured that all sensitive data has been effectively deleted from the system parts and components affected during prior decommissioning. This is important not just with a view to IT security, but also in terms of data protection, especially in light of the EU's General Data Protection Regulation (EU GDPR). If in doubt, the decommissioning procedures to delete sensitive data must be repeated. Alternatively, physical destruction of the memory containing sensitive data can be an option. No further abstract security requirements are laid down for this phase.

Contingency measures/restoration of operations

From a security perspective, an emergency exists if it is found that an operating system may not be behaving as usual due to unauthorised manipulation of the system. As a rule, this prevents the operator from using the system securely or efficiently. When dealing with security incidents at a production plant, this leads to a conflict of objectives: On the one hand, the cause must be investigated in order to determine the extent of damage and to be able to prevent such incidents in the future; on the other hand, operations must be restored quickly in order to minimise consequential damage (e.g. due to a loss of production). Since unauthorised manipulation is usually unforeseen and therefore initially goes undetected, the cause can often only be found on the manipulated object and the investigation requires time during which restoration of operations will have to wait. Quick restoration of operations often erases traces that can be used in the subsequent analysis of the root cause of such a security incident.

One proven approach in this case is to take a snapshot of data and the states of the affected system for later analysis and then to bring the data and states of the system back up to an operational state before the analysis is completed. This state can be restored from a backup copy of a (presumably) not yet manipulated state. To be able to do this, the operator must also be able to make backup copies and to reload these again. He must also be able to take a snapshot as quickly as possible with or without the integrator's assistance. Sensitive data must be protected both when making and restoring backup copies and when taking snapshots. The operator may not gain possession of sensitive integrator data (e.g. know-how of the plant application) and vice versa (e.g. private keys to operator's certificates or log data related to operator's personnel).

Simple restoration of a secure configuration is not always easily possible, since at this point in time the target system is in an insecure and perhaps even unknown state. Therefore, different measures must be selected from case to case (e.g. configuration reset, reload software or replace components completely).

Unfortunately, simply restoring a formerly safe state is often not enough. Unauthorised access could result in the seizure of private keys to certificates. In this case, it would be even easier than before for the attacker to repeatedly intervene and manipulate – this time virtually indistinguishable from authorised access – so that simple restoration would create a false impression of regular operations whereas in fact ongoing attacks are still successful. Therefore, if sensitive data material, such as private keys, is suspected of being compromised, new key pairs must be generated and new certificates issued as a precautionary measure. In cases like these, passwords are much more difficult to replace if their comparison values (the so-called password hashes) are stored on the devices.

That's why in the case of password-based authentication, the password should be verified using an authentication service, such as an LDAP server, an Active Directory or a Kerberos system where passwords for entire areas can be renewed.

To prevent unauthorised manipulation or copying of private keys, the keys should be safely kept in hardware security modules (Secure Elements). In cases like these, private keys and the pertinent certificates do not have to be renewed as long as the hardware, i.e. the component including the secure element, is still intact. That's because an attacker cannot gain possession of them. This also eliminates the above-described need to renew private keys and certificates after an attack. This paper does not take a more detailed look at the requirements for secure elements since they are not the focus of this discussion. It should be noted, however, that due to the use of several certificates and pertinent key pairs, which has already become apparent in the course of the discussion above, requirements for secure elements would certainly need to be discussed.

The following security requirements apply to support for restoring operations and contingency measures:

1. It should be possible to take snapshots of system and component data (log data, temporary data, etc.) for forensic purposes, so that they do not contain any sensitive information but still allow an analysis of security incidents.

2. It should be possible to make and restore backup copies of systems and components in such a way that sensitive data is still protected in the backup copies and restoration only transfers the sensitive data to components in a trusted state.

3. It should be possible to quickly exchange key pairs and pertinent certificates in the case of components that have probably been compromised.

4. For users, components should support certificate authentication and/or verify passwords using an authentication service.

Solution sketch/discussion

The aim of this section is to identify requirements and issues yet to be discussed in order to trigger further discussions on the secure application of the OPC UA standard. The aim is to stimulate discourse on the application of the OPC UA standard. Therefore, once a solution to a requirement is found and mentioned, it is deliberately not discussed further here. The document is based on both the published OPC UA standard (13) and subsequent versions of its parts. The corresponding status is referenced at the respective point. The OPC Foundation has also published a recent whitepaper (14) on the topic.

In the following, solutions are sketched with one table for each phase of the lifecycle and in each case related to the individual security requirements of the phase explained above. When describing the solution sketches, two colours are selected for the text which should provide a quick overview: Green text means that at least one solution is sketched based on established standards or common technologies. Blue text refers to points open for discussion.

Anticipating repetitive sketches

Before reference is made to the individual security requirements, an overview is used to explain some repetitive solution sketches.

The security requirements show that digital identities are needed from different sources. As a reminder, the terms used for the sources and identities are repeated here:

1. Identities issued by the component manufacturer (e.g. ZH_N manufacturer certificates)

2. Identities assigned by the integrator (e.g. ZI_N certificates)

3. Optional identities valid in the system (e.g. ZA_N certificates)

4. Identities issued by the operator (e.g. ZB_N certificates)

Security domains

The sources of identities correspond to the different security domains. A distinction is made here particularly between the integrator's security domain and that of the operator. Figure 5 on the next page shows the different domains and their location. For the sake of clarity, the manufacturer's security domain is not presented. The illustration shows an "SD-I" security domain for the integrator. "SD-A" is a security domain within the system. The domain for the operator is "SD-B".

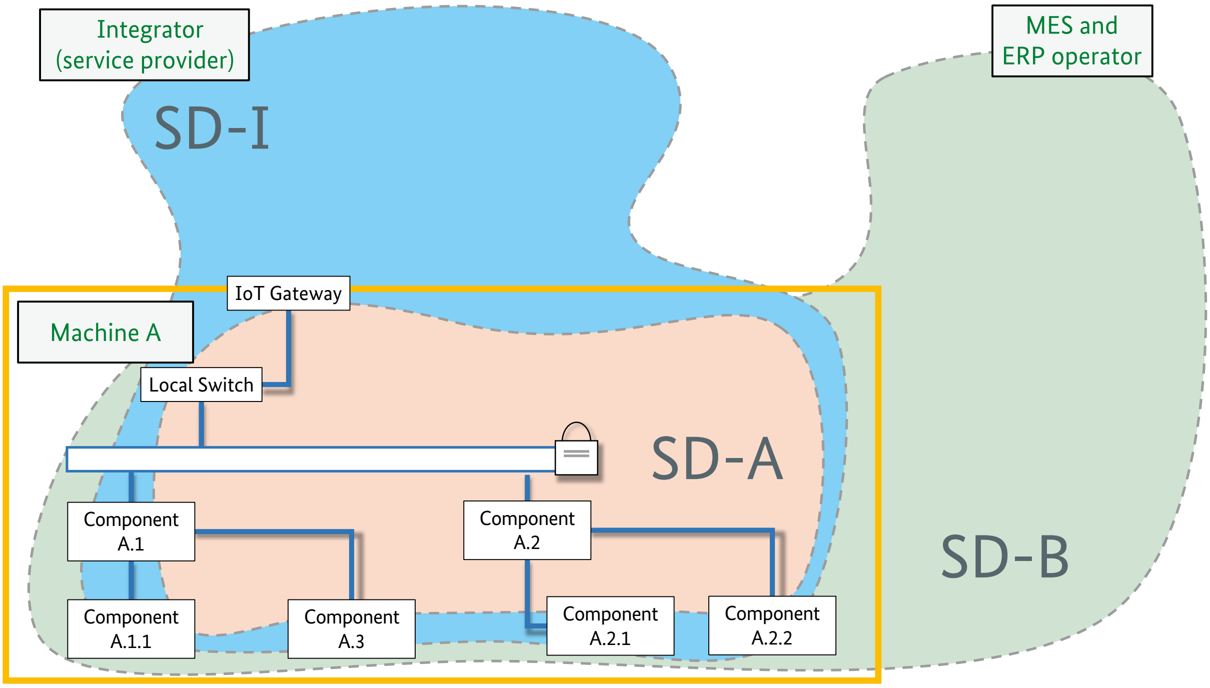
Provision of authentication criteria (e.g. trust lists)

Part 12 of the OPC UA standard (15) defines two mechanisms for the automatic management of certificates (certificate management):

- With "pull management", an OPC UA application can regularly obtain various trust lists from an OPC UA server, which is referred to in the standard as the Global Discovery Server (GDS). An information model (collection of objects, their types and methods) is defined in the standard that describes an interface in the GDS via which OPC UA applications can perform "pull management". This means that they regularly copy the trust lists.

- The second mechanism is called "push management" and defines an information model for OPC UA applications that are an OPC UA server. Via this interface, a management application can update the trust lists from the GDS according to a schedule in the target server, i.e. copy them there. The management application works in both directions as an OPC UA client, both to the GDS and to the target server. The standard does not define where the management application runs. It is conceivable for this application to be located near the GDS, providing several OPC UA servers with new information for trust lists. For a server with "push management" capability, the standard defines more precisely that the target server must have an object called "ServerConfiguration" under which various certificate groups are referenced with each group representing a trust list.

Figure 5: Localisation of the security domains using machine A as an example – without the manufacturer's domain



Source: Plattform Industrie 4.0

With OPC UA, a trust list contains precisely the information needed to check certificates of a security domain in the form of trusted certificates, optionally additional certificates to complete certificate chains (issuer certificates) and optionally revocation information. OPC UA defines two types of trust lists in the standard (via the certificate groups), i.e. those for checking application certificates and those for checking user certificates. This distinction matches the above security requirements.

In the solution sketches described below, the two methods of certificate management via OPC UA and Global Discovery Server (GDS) are used, so that all of the components or relevant plant parts in each security domain with a GDS and the underlying Certification Authorities (CAs) receive copies of the trust lists. Several CAs can play a role here:

- one CA for devices and software processes and
- optionally one CA for user identities.

The trust lists must be managed accordingly

- one trust list for devices and software processes issued by the pertinent CA, and
- optionally one trust list for user identities, issued by the pertinent CA.

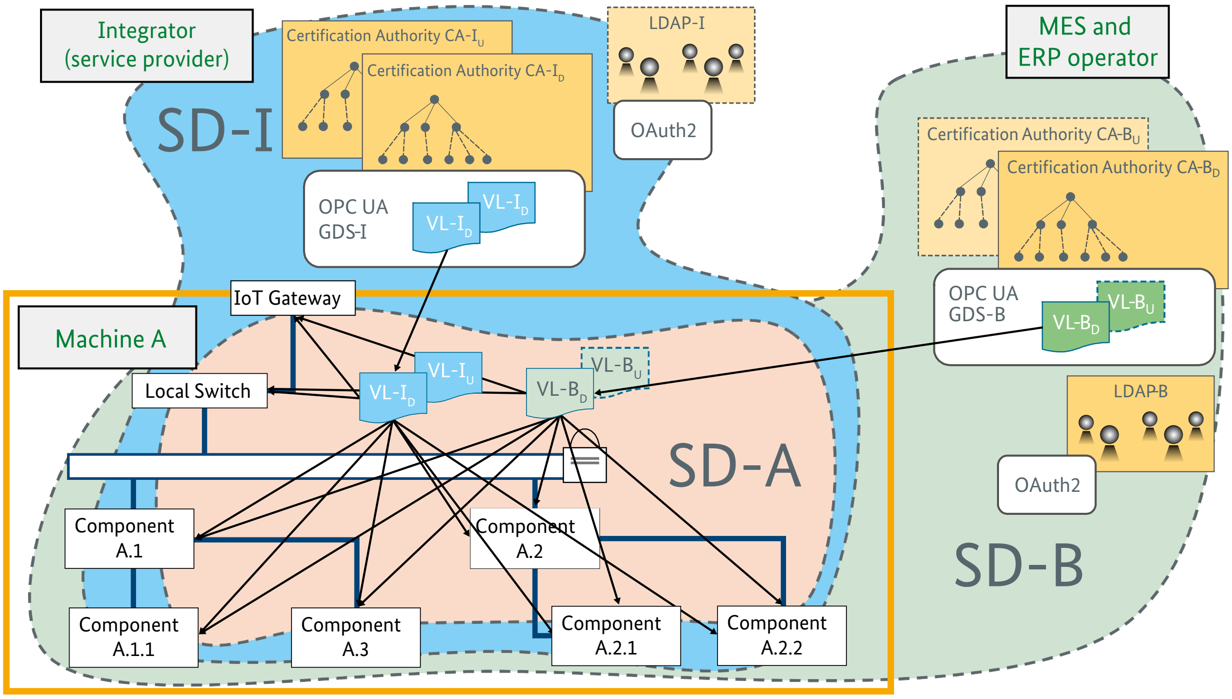
It is quite possible for this information to also be distributed in other ways. This approach takes into account the requirement, i.e. to propose as little as possible in addition to OPC UA, which was already explained in the introduction with regard to the significance of OPC UA.

Instead of using a set of trust lists for each security domain, it is also conceivable to provide the same set of trust lists for different domains within a GDS. However, this would raise the question as to who is responsible for maintaining the security domain. As soon as other security domains are included in a scenario, for instance, by suppliers, it becomes clear that, instead of helping, this kind of mixed managed trust list leads to even greater complexity. Similarly, it is possible to include only parts of a certification hierarchy of another Certification Authority in a trust list, for example, in order to declare certain plant parts of another security domain to be trusted. It is also necessary once again to weigh up the complexity of responsibility issues with the technical simplification in the form of a lower number of trust lists to be distributed. This paper does not examine any further the more complex organisational procedures but instead the solutions describe in simple terms the approach with strictly separated trust lists for each security domain.

Since the release of version 1.04 in November 2017, the OPC UA standard describes in the "Services" (16) and "Mappings" (17) sub-documents new possibilities for authenticating users, for instance, using OAuth2 with JSON Web Tokens to check passwords with check criteria that can be stored in an LDAP server.

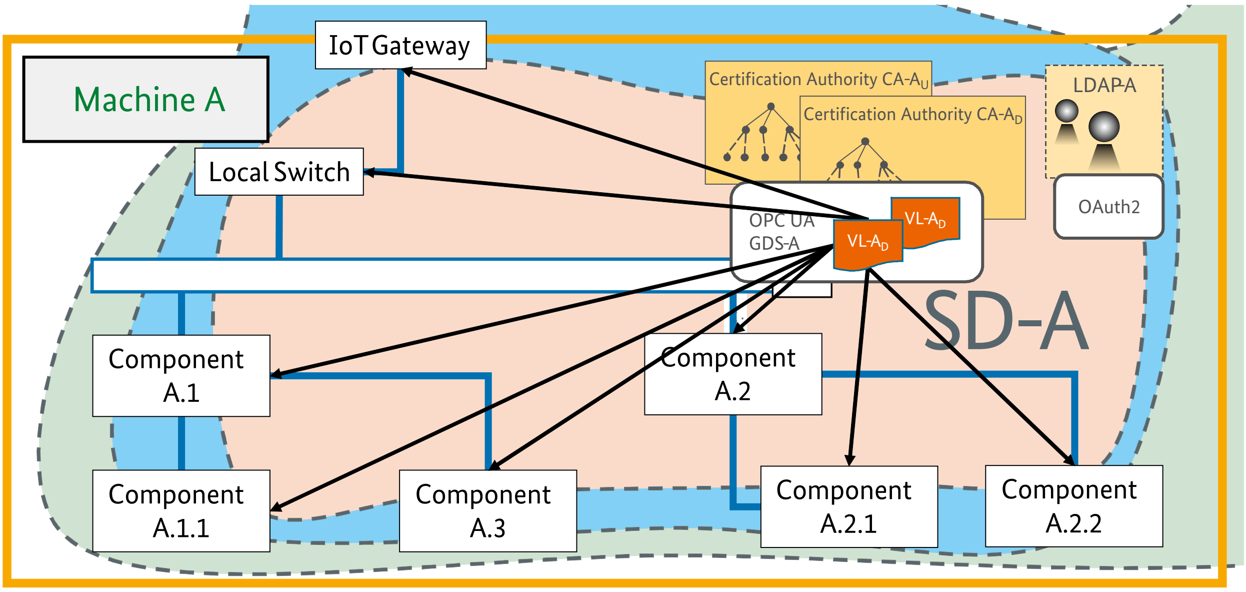
Providing the system with different trust lists, own certificates and password check criteria is depicted in Figure 6 below using the example of the SD-I and SD-B security domains. Every component communicating via OPC UA requires trust lists and own certificates. For more complex systems, it makes sense to regularly copy the trust lists once from one domain to the system and to distribute them from there, for example, with a local OPC UA server that serves as a GDS proxy with a cache for the components in the system. This concept is only outlined here and will not be discussed further. In Figure 6, the SD-I security domain is the Certification Authority (CA) for the devices and software processes shown as CA-ID and for the CAs for the users shown as CA-IU. The pertinent VL-ID and VL-IU trust lists are provided and distributed via the GDS called "GDS-I". For the SD-B domain, this is carried out in the same way via the GDS-B by the CAs called CA-BD and CA-BU with the VL-BD and VL-BU trust lists.

Figure 6: Provision of (copies of) trust lists and password verification information



Source: Plattform Industrie 4.0

Figure 7: Provision of (copies of) trust lists and password verification information



Source: Plattform Industrie 4.0

Figure 6 also shows that passwords can be used instead of certificates in order to authenticate users. Besides a conventional method, i.e. the use of passwords configured locally in components and the pertinent verification criteria (tables with user names and passwords), as described above, the OPC UA standard describes the use of OAuth2 as a network method for authenticating users. Locally configured password tables have disadvantages that have already been discussed in the examination of the lifecycle. There are also disadvantages to using OAuth2:

Besides OPC UA, another protocol is required when using OAuth2 and this causes user authentication to fail if the OAuth2 server(s) is/are not available.

Certificates for users, on the other hand, allow new authentication processes to be performed for an interim period without current replication of trust lists. In the main, obsolete revocation information forms the limit for the interim period. The Certification Authority usually defines how long revocation information can be used.

No security domain must (but can) support both methods of user authentication, i.e. with certificates and/or OAuth2. Figure 6 shows an example of an LDAP server called “LDAP-I” with an upstream OAuth2 mechanism in the SD-I domain and an LDAP server called “LDAP-B” in the SD-B domain. The password verification information is not replicated, however, communication takes place indirectly with the OAuth2 servers.

The fact that a Certification Authority can also distribute trust lists within a system in the SD-A security domain is shown in Figure 7 as an example. The certification authorities, the LDAP server and the trust lists are named in the same way as the previous examples.

Provision of identities (certificates and key pairs)

For pull management and push management, Part 12 of the OPC UA standard “Discovery” (15) explains not only the provision of trust list copies, but also the provision of digital identities in the form of asymmetric key pairs and the pertinent certificates. Using both methods, an OPC UA application can obtain (pull management) or provide (push management) the required number of certificates. It is up to the application whether it generates the corresponding key pair itself or whether it receives it from the providing end. This is ideal for supporting cryptographically weak devices that do not have sources of good random numbers for generating keys. At the same time, devices are supported that have a secure element for generating and protecting key pairs and especially the private key, such as devices with a Trusted Platform Module (TPM), as is already the case with most PC platforms today.

The OPC UA standard allows an OPC UA client to choose which certificate to use with the server as part of a secure connection setup. An OPC UA server can offer multiple endpoints and define a certificate and key pair for each endpoint to be used. If an OPC UA server uses several endpoints, it is still a single process within an operating system. Internally, the same address space and slightly or completely different address spaces can be displayed behind each endpoint, i.e. sets of nodes and their references. The OPC UA server remains a single OPC UA application. When a client connects to a server, the client indicates to the server, through the URL of the endpoint, the endpoint to which the client wishes to connect.

The solution sketches here assume that an OPC UA server offers precisely the same number of endpoints within a system as the number of communication relationships that it supports in security domains. A component with an OPC UA server, which is to be able to communicate both within the system, with the operator and the integrator, therefore has three endpoints, one each for the domains: SD-A, SD-I and SD-B. For each of these endpoints, it uses a certificate which it has received via the GDS of the respective domain.

Similarly, the OPC UA clients of all of the system components issue their certificates via the GDS of the respective domain. The communication relationships needed are already shown with illustrations in the previous discussion.

Authorisation of communication and interaction partners (partners)

In the above examination of the lifecycle, access control mechanisms and authorisation mechanisms, such as roles and rights, and alternatively attributes and rules, were addressed rather awkwardly with regard to user authorisation. This is due to the fact that different concepts can now be found in different standards.

Role-based access control (RBAC) mechanisms, for instance,

- are described in the OPC UA standard, version 1.04, released in November 2017, in the "Address Space Model" (18) and "Information Model" parts (19), as an option for OPC UA servers.
  - Server manufacturers are at liberty to implement a different procedure.
  - The OPC UA standard not only describes the effect of roles and rights in the OPC UA server, it also defines
    - their presentation to clients and users who are allowed to view this information,
    - extensions of the information model for changing the assignment of rights and roles to the individual nodes in the server address space,
    - a mechanism (IdentityMappingRuleType) for mapping identities of OPC UA clients and users using
      - attributes of users from their authentication tokens (e.g. attributes from their JSON Web Token, such as their group or role affiliation),
      - their certificates,
      - the endpoint selected for communication and
      - certain combinations thereof.
- Part 4-1 of IEC 62443 (4) defines for components with the CR 2.1 requirement and its RE2 extension that the enforcement of authorisation for human users must be based on roles and the assignment of roles to human users must be directly definable or modifiable, or via compensation mechanisms by IT security.
- The IEC/TS 62351-8 (20) standard, which is binding for the energy sector, stipulates that authorisation must be implemented on a role-based access control system for data exchange with devices for the energy sector.

Attribute-based access control systems (ABAC for short), on the other hand, are comparatively new and not yet widespread in industry. A technically sound definition can be found, for instance, in NIST SP 800-162, a "Guide to Attribute based Access Control ..." (21). ABAC systems can be seen as a superset of RBAC systems, where rule sets can be used instead of direct assignments between identities and roles in order to define complex mapping functions between attribute values assigned to an identity, objects or environment and the roles to which rights are assigned. To some extent, the OPC UA standard already goes in the direction of ABAC, because it already describes mapping rules between attributes from an authentication token and assigned roles.

Due to the spread of RBAC in the relevant industrial standards, the solution sketches here refer to role-based access control as described in the OPC UA standard.

Taking possession

Requirement	Can be fulfilled by/Point open for discussion
1. It should be possible to verify the authenticity of the component on the basis of a certificate from the manufacturer.	The Companion standard "Devices" (22) published by the OPC Foundation explains an information model for describing devices. It does not yet include device authentication. A dedicated, always available "endpoint" in the OPC UA server of a component, which is to be called the manufacturer endpoint here, could use a certificate from manufacturer ZHN directly to authenticate the server to clients. Such an endpoint alone cannot prove hardware "authenticity". However, OPC UA can support this as a communication protocol and information model with security features. The "manufacturer endpoint" could be extended using methods and objects that can be requested using OPC UA and provide proof of the integrity and authenticity of the component's hardware and software. In the field of "trusted computing", this is referred to as "remote attestation" and is usually based on hardware security modules (secure elements). Instead of complicated remote attestation, the device could, for instance, also use the private key for the above endpoint only in the case of trusted hardware and software, i.e. a secure element releases the private key only if a secure start procedure (trusted boot/secure boot) has taken place. For security reasons, the "manufacturer endpoint" should not allow access to the device's full functionality. Ideally, this access should be restricted to verifying authenticity and taking possession of the device, so that the owner is still required to issue a specific certificate.[3]
2. It should also be possible to reset all component settings to the manufacturer's factory settings.	A reset mechanism on the device can allow it to be reset to factory settings. An OPC UA standard could (additionally) define a reset method for devices. Secure components must delete sensitive data when reset to factory settings.
3. The basic configuration should have a secure design and the device should be supplied with a secure configuration.	By applying the "Security by Design" principle during product development, as explained, for instance, in Part 4-1 of IEC 62443 (4) for component manufacturers, the manufacturer knows a safe basic configuration and can bring the devices to market with settings that comply with the principle of "Security by Default". Only a few of the products currently available have been developed according to the "Security by Design" principle. The same goes for products supplied according to "Security by Default". In the interest of future product security, it is urgently recommended that manufacturers apply both principles.
4. The integrator should be able to define all of the authentication criteria used by the component to verify other identities. These include all passwords and all certificates that the component trusts, except for a few certificates designed by the manufacturer for special purposes, such as authentication of firmware updates.	Using the above-described certificate management via the Global Discovery Server (GDS) according to OPC UA "Discovery" (15), components could be automatically provided with specific authentication criteria for each security domain by setting GDS relationships. User passwords would be automatically set specifically if components supported OAuth2 (for example with underlying LDAP servers). An initial password may have to be defined for an administrative user at the factory, so that a component can be put into operation in a secure manner. A procedure commonly used to securely individualise this password is, for instance, to set the password ex works to suit the individual device and to print it on the housing at a usually concealed location. Other methods are available, but are not discussed here. The specific choice depends to a large extent on the component's area of application. An appendix to OPC UA "Discovery" (15) roughly outlines how initial provisioning of an OPC UA application can be provided with an identity and a certificate. For security reasons, it is recommended that both parties make an explicit and once-off manual announcement. The GDS should be made known to the application. The application does not yet know the identity of the GDS. The device must be made known to the GDS. The GDS is not yet able to identify the device using an identity issued by the GDS. The procedure is only roughly explained in the OPC UA standard; support for this procedure through standardised parts in the information model (objects and methods) would be helpful. Although it is possible according to the standard to use a Local Discovery Server with Multicast Extension (LDS ME) in order to make new OPC UA applications with corresponding multicast capability known to a GDS, this approach does pose security problems that should be pointed out in the standard itself.

Source: Plattform Industrie 4.0

[3] Failure to do so would mean operating with certificates that cannot be revoked either by the integrator or the operator and accordingly an attacker could introduce into the communication network devices which are from the same manufacturer but were configured by the attacker.

Integration

Requirement	Can be fulfilled by/Point open for discussion
1. It should be possible to assign an integrator-specific identity with a pertinent ZI_N certificate to the component.	The OPC UA standard defines ways to set certificates for servers and clients. Examples are already explained above together with certificate management according to OPC UA "Discovery" (15).
2. For the purpose of communication in the system, the component should: 2.1 be given and be able to use a system-specific identity including a ZAN certificate (which may be, but does not have to be the integrator-specific identity) and	Part 4 "Services" of the OPC UA standard (16) explains that the same server can have different endpoints. Each endpoint can be equipped with its own certificate. This means, for instance, that a component could offer several endpoints in its OPC UA server, for example, one endpoint for each identity. OPC UA clients integrated into a system can decide which identity to use with which certificate vis-à-vis an OPC UA server before establishing the connection. This is possible by individual assignment of the "clientCertificate" field in the "OpenSecureChannel" service request, which a client must access for a server in order to establish a secure connection.
2.2 be given and be able to use a system-specific trust list.	Part 12 "Discovery" of the OPC UA standard (15) explains how both OPC UA clients and OPC UA servers can be provided with trust lists via "Certificate Management" of a "Global Discovery Server" (GDS). As a rule, first provision takes place with the assignment of the identity and the corresponding certificate. This can also be implemented for components. For more complex systems, it makes sense for trust within the system to be autonomously created and managed using certificates that are valid only within the system. A "small" edition of a GDS with an integrated Certification Authority can be used for this purpose. For less complex systems, where using a GDS would be too complex, distribution can be implemented with file system operations. However, the provision paths and access authorisations must be proprietary. Instead of automatic certificate management via a GDS, manual distribution via freely available OPC UA clients with a user interface can also be implemented, for instance, via UaExpert.
3. For the purpose of communication with the integrator's processes and staff, the component should be given and able to use the integrator-specific identity, including the ZI_N certificate, as well as an integrator-specific trust list.	The previously outlined distribution of trust lists and identity certificates based on certificate management via a Global Discovery Server (GDS) allows specific distribution paths for each security domain. Since with the "push management" method the trust lists are implemented with their own objects with methods, the RBAC concept according to OPC UA "Address Space Model" (18) and "Information Model" (19) specifies that these objects can be set so that they can only be modified from the corresponding domain, for instance, by assigning the rights to a role for which assignment criteria (IdentityMappingRules) define that this role only applies to communication via the endpoint that is used for the respective security domain. OPC UA "Discovery" (15) defines three types of trust lists, one for application certificates, one for user certificates and one for HTTPS certificates. All types are referenced directly from the ServerConfiguration object via which the respective push management is implemented. However, according to the solution sketch already explained above, a set of trust lists is required for each endpoint of a server. This means that each endpoint would have to implement the set of trust lists below the ServerConfiguration object differently, which would be possible under the standard. The only thing missing in the standard is the option of initially establishing the provisioning of another endpoint via an existing endpoint.
4. The component should support an access control mechanism that can be used to define rights independent of specific identities.	As explained above in the anticipation of repetitive sketches, the OPC UA "Information Model" (19) explains that, according to the OPC CU standard, rules in the form of IdentityMappingRules for assigning identities to roles can be implemented and set in the OPC UA server. There are currently no tools available on the market that allow manufacturer-independent management of IdentityMappingRules.
5. The rights in the component should be set so that certain rights are required in order to change rights.	According to the OPC UA "Address Space Model" (18), one right that can be defined for a node in the OPC UA server is permission to change the rights of this node. If this right is not assigned to a role, the rights to this node cannot be changed.
6. The rights in the component should be set so that certain rights are required in order to set the rules for assigning rights to identities..	According to the OPC UA "Address Space Model" (18), the individual nodes of an OPC UA server can be determined and (given the corresponding rights), if necessary, it can also be determined which role has which right at this node. When clients are connected individually, roles are assigned to the session. Which role is to be assigned to which client and for which session can be implemented and set for each role using IdentityMappingRules according to OPC UA "Information Model" (19), because the IdentityMappingRules themselves are implemented as nodes and can be assigned rights. Since the standard also allows methods to be defined for each role via which the properties of the IdentityMappingRules can be changed, and because these methods are themselves their own nodes in the address space of the OPC UA server, rights can also be set for their use.

Source: Plattform Industrie 4.0

Integrator's preparation of handover

Requirement	Can be fulfilled by/Point open for discussion
1. It should be possible to adjust in the system the authentication criteria, rights and rights assignments (to roles or access rules) defined by the integrator for maintenance access.	As explained above in the anticipation of repetitive sketches, the OPC UA "Information Model" (19) explains that, according to the OPC CU standard, rules in the form of IdentityMappingRules for assigning identities to roles can be displayed and set in the OPC UA server. There are currently no tools available on the market that allow manufacturer-independent management of IdentityMappingRules.
2. The integrator should be able to activate in the system authentication criteria and rights temporarily needed for the commissioning engineer, so that the system can authenticate the commissioning engineer, if required, even without a network connection, and it should be possible to remove these rights assignments and authentication criteria for the commissioning engineer again.	If user authentication based on user name and password is possible in the system (for example, with password tables or an integrated LDAP server), a user can be temporarily set with a password for the commissioning engineer. This user can be removed in the same way as it was set up. Alternatively, a user certificate for the commissioning engineer can be temporarily included in the list of trusted certificates in the security domain of the system or in the integrator's domain. When a self-signed certificate is used, offline use is even possible without the need for up-to-date revocation information. Removing this certificate automatically blocks this explicit access path.
3. The integrator should be able to delete unnecessary access paths, authentication criteria and rights from the system.	The mechanisms already mentioned for managing authentication criteria, roles and rights can also be removed using mechanisms that conform to the OPC UA standard. The deactivation of access paths, such as endpoints in the OPC UA servers of systems, however, is so specific to the system that the integrator should define his own paths here.

Source: Plattform Industrie 4.0

Takeover during commissioning

Requirement	Can be fulfilled by/Point open for discussion
1. It should be possible to verify that the system comes from the integrator and is in the state defined by the integrator.	The verifiability of the authenticity of the system can be supported via an endpoint in an OPC UA server of the system, if, for instance, this endpoint is defined precisely for the purpose and always uses the certificate issued by the integrator to OPC UA clients. It is left to the system manufacturer to define the exact procedure for verification. It is to be expected that within the operator's environment the system uses other DNS names and/or IP addresses than those used in the integrator's environment. For security reasons, the OPC UA standard also allows clients to search and compare the DNS names and/or IP addresses of the endpoint URL in the certificates. For these two reasons, an exception should be made here, for example, either the OPC UA client should ignore the failed comparison of DNS names and/or IP addresses during verification or the certificate issued by the integrator for the installation should not contain any DNS names and/or IP addresses at all.
2. It should be possible to introduce the operator's identities together with the ZB_N certificates issued by the operator into the system, while retaining the previous identities and pertinent certificates (e.g. ZI_N integrator certificates and ZH_N certificates of the component manufacturers).	OPC UA servers and clients can be designed in compliance with the OPC UA standard, so that they have and can use multiple identities and the pertinent certificates.
3. It should be possible to store authentication criteria for identities of different security domains separately and simultaneously in the system.	The previously outlined distribution of trust lists and identity certificates based on certificate management via a Global Discovery Server (GDS) allows specific distribution paths for each security domain. OPC UA "Discovery" (15) defines three types of trust lists, one for application certificates, one for user certificates and one for HTTPS certificates. All types are referenced directly from the ServerConfiguration object via which the respective push management is implemented. However, according to the solution sketch already explained above, a set of trust lists is required for each endpoint of a server. This means that each endpoint would have to implement the set of trust lists below the ServerConfiguration object differently, which would be possible under the standard. The only thing missing in the standard is the option of initially establishing the provisioning of another endpoint via an existing endpoint. This is a comfort function which could promote acceptance of the procedure because it avoids work and facilitates the process. What's more, this would create a manufacturer-independent option.

→

Takeover during commissioning (continued)

Requirement	Can be fulfilled by/Point open for discussion
4. It should be possible to both set and at the same time activate rights with reference to the authentication criteria of the identities of a security domain, so that the system can distinguish between the operator and the integrator (for maintenance) and enforce the corresponding rights.	The OPC UA "Address Space Model" (18) and "Information Model" (19) specifications define roles as full-scales nodes in the OPC UA address space – with an identifier (role name) and the corresponding namespace. The same name combined with different namespaces results in different nodes. A separate namespace can be defined for each security domain. Because rights according to the OPC UA standard are always defined in relation to roles, it is possible to define roles with the same names and pertinent rights in relation to different security domains without overlapping and colliding by making the names unique; this is achieved by combining them with the namespace of the security domain. From a technical, OPC UA perspective, these roles are different despite the fact that the names are identical. The IdentityMappingRules for domain-specific roles can be used to additionally define that these roles can only be used for certain endpoints. The endpoints are assigned to exactly one security domain after the above anticipation of repetitive sketches. The requirement can be met in this way. Because different security domains should use different certificate hierarchies and autonomously determine which application has which identity (application URI according to the OPC UA standard), the restrictions or permissions permitted by the OPC UA standard for certain applications by naming the application URI should not be defined in the IdentityMappingRules. This is because they have no fixed reference to a security domain and an attacker could exploit this to extend the rights from one domain to the other. One remedy here is to only ever permit application URIs in conjunction with precisely the endpoint for the role that is assigned to the security domain from which the application URIs originate.
5. It should be possible to map the descriptions actually used by the operator (external roles, external attributes and their characteristics) for the identities of his personnel and his processes to designations defined by the integrator (internal designations).	The OPC UA standard defines ways to set certificates for servers and clients. Examples are already explained above together with certificate management according to OPC UA "Discovery" (15).
6. It must be possible to delete the access rights and authentication criteria temporarily set in the system regarding the identities for access by the commissioning engineer.	The solutions described above for setting temporarily active access rights and authentication criteria can be reversed.

Source: Plattform Industrie 4.0

Operation and security maintenance

Requirement	Can be fulfilled by/Point open for discussion
1. In the case of identities and the pertinent certificates as well as key material, 1.1 it should be possible to renew them without interruption and	Certificate management via a GDS allows certificates and key material to be renewed. It implies use of the new material the next time a connection is established. Support for certificate management via GDS is not yet widely used in OPC UA applications. For many OPC UA applications, the use of renewed key material and/or the pertinent certificate requires restarting the application.
1.2 depending on the issuer, it should be possible to renew them on different paths (integrator certificate versus operator certificate).	It is possible for an OPC UA application (client or server) to obtain its certificates from different Global Discovery Servers (GDSs) at the same time, each for a different number of certificates for each GDS. One approach is already explained above in anticipating repetitive sketches.
2. Authentication and authorisation criteria for identities should be 2.1 renewed regularly and	Certificate management via a GDS allows trust lists to be renewed. It implies use of the new material the next time a connection is established. OPC UA "Information Model" (19) describes an information model with methods for managing rights and assigning roles in an OPC UA server to identities. Role and group assignments can also be managed via LDAP servers if user authentication takes place via OAuth2. Because the latest version of the OPC UA "Information Model" (19) is so new, there are currently no known tools for managing the authorisation criteria in OPC UA servers across different manufacturers.

→

Operation and security maintenance (continued)

Requirement	Can be fulfilled by/Point open for discussion
2.2 separately for each person responsible (integrator versus operator).	Different Global Discovery Servers (GDS) can be used for each security domain. This also supports different certificate validity periods which can be particularly helpful in the case of comparatively seldom maintenance access. For each security domain, separate servers can be used for password authentication of users, as already explained in the anticipation of repetitive sketches. The current OPC UA standard does not define a path via which password-based authentication of users can be configured if the password verification criteria are stored directly on the device. It is therefore recommended that password-based authentication services that enable password management be used, such as LDAP and OAuth2.
3. When establishing a connection to the system or its components or in the opposite direction, it should be possible to select, 3.1 which identity and which certificate are relevant (integrator versus operator) and	Within the OPC UA standard, it is possible for OPC UA servers to define different endpoints and to use different certificates for this. This means that different endpoints can be offered for the OPC UA clients of the different security domains. Within the OPC UA standard, an OPC UA client can decide each time a connection is established which identity (application URI) to use and which certificate to use to identify itself.
3.2 which verification criteria are relevant for verifying the remote station and its users.	When the connection is established, OPC UA clients can decide which trust list to use to check the certificate of the OPC UA server. For OPC UA servers, the identity of the endpoint and the respective certificate can be used to define which trust lists are to be used to verify clients and their users.
4. Different keys and certificates should be used for encryption and authentication/signing.	Current software development kits for OPC UA applications do not support the use of different keys for signature, authentication and encryption; nor is this included in the OPC UA standard. Since the procedures in the OPC UA protocol are inherently secure and there is no interference between the methods, the key pairs for the OPC UA applications should only be used for the OPC UA protocol. If a component requires certificates beyond the OPC UA protocol, for instance, to establish secure communication with other protocols, to receive encrypted files or to generate signatures for files, separate key pairs and pertinent certificates should be used in each case. As long as OPC UA applications use the same key pair for the authentication and the negotiation of the symmetric keys for encrypting communication, a communication inspection at trust boundaries, for example, using a so-called middlebox, will inevitably always also compromise authenticity. The OPC UA standard could include supporting properties in the protocol part of OPC UA, which are also not yet included in other protocols, for example, in the form of a recommendation of different key pairs and an explanation of the mechanisms for this.

Source: Plattform Industrie 4.0

Decommissioning

Requirement	Can be fulfilled by/Point open for discussion
1. Components should be able to securely delete sensitive data.	This requirement is usually addressed by reset mechanisms that reset the devices to their factory settings. This deletes all data that has been added to the factory settings, including sensitive data that was introduced into the component by the integrator or operator. A reset button, for example, is provided for this purpose.
2. In the infrastructure of the operator and of the integrator, it must be possible to disable access partially or completely for components or entire systems.	If, as suggested, trust lists are distributed using "Certificate Management" via a Global Discovery Server (GDS), it is possible to disable components and entire systems. Typically, the applications of a system or component are de-registered in the GDS and the pertinent certificates are revoked, so that this revocation information is distributed at the next opportunity. For more precise disabling, mechanisms can be used which, according to OPC UA "Address Space Model" (18) and "Information Model" (19), allow the rights and roles of an OPC UA server to be modified.
3. In the case of temporary disabling, it should be possible to activate and cancel disabling in the operator and/or integrator's infrastructure.	The disabling mechanisms mentioned above can also be used with temporary effect, because it is also possible to reverse disabling in the same way. Note: Temporary disabling requires effective time synchronisation. The correct time is generally important for security, for instance, also for the time stamps of audit data. At this point, reference is also made to threat number 37 "Manipulating the time" from the "OPC UA security analysis" by the Federal Office for Information Security (12).

Source: Plattform Industrie 4.0

Contingency measures/Restoration

The discussion of contingency measures and restoration in the lifecycle examination already shows that very system-orientated solutions that use the capabilities of the individual components must be provided in both cases. The latter must be expected for large and small devices with very specific characteristics, so that it is very difficult and too far-reaching to outline a general solution in the solution discussion. This is therefore not included in this paper. Only individual, selected security requirements from contingency measures/restoration are dealt with here:

Requirement	Can be fulfilled by/Point open for discussion
1. It should be possible to take snapshots of system and component data (log data, temporary data, etc.) for forensic purposes, so that they do not contain any sensitive information but still allow an analysis of security incidents.	This is not discussed here because it depends too much on the system, especially due to the system's general data and the data protection context. Part 3 of the OPC UA standard "Address Space Model" (18) defines the concept of audit events. They are suitable for reporting security-relevant processes and can be forwarded to central systems. However, how these events are recorded and their inclusion in a snapshot depend on the system.
2. It should be possible to make and restore backup copies of systems and components in such a way that sensitive data is still protected in the backup copies and restoration only transfers the sensitive data to components in a trusted state.	This is not discussed here because it is far too system-specific.
3. It should be possible to quickly exchange key pairs and the pertinent certificates in the case of components that have probably been compromised.	This can be ensured by distribution via OPC UA GDS. If the key of a component is compromised due to an emergency, automatic renewal of the certificate via certificate management is not sufficient. Automatic renewal may only be used for components with keys that have not been compromised. For compromised keys, the key pair and the pertinent certificate must be renewed using manually supported paths, as is the case with the first provisioning of this material. Systems and components should support the above-mentioned push management (15) for unscheduled initiation of certificate and key renewal or they should offer a special method or mechanism for initiating a renewal cycle using pull management.
4. For users, components should support certificate authentication and/or check passwords using an authentication service.	Certificate management according to OPC UA GDS provides separate trust lists for users and thus also certificates and key pairs separate from applications (devices and software processes). These can therefore be distributed separately when using OPC UA GDS. For distributing other authentication criteria, such as passwords, OPC UA supports token authentication, for instance, via a JSON Web Token issued by an OAuth2 server. Password verification can take place behind this via the LDAP server. The password verification criteria can be quickly renewed there accordingly if necessary. This would have no direct effect on components and systems.

Source: Plattform Industrie 4.0

Summary and outlook

This document discusses the secure integration of a machine into an operator network with OPC UA. By examining the requirements over the lifetime of the machine, it can be seen that the latest version of the OPC UA standard supports the necessary functions. Based on the results, suppliers of OPC UA toolkits as well as component manufacturers and machine designer can develop their offerings further in order to achieve interoperable and efficient use of the security functions of OPC UA.

In a more detailed document, cross-company communication with OPC UA will be examined. An operator model will serve as an example here where two stakeholders, i. e. a service provider and a factory operator, have to interact in a secure manner with the same machine. Interaction between the two security domains and the corresponding requirements for integrity, confidentiality and monitoring will have a key role to play in this analysis.

Glossary

Authentication data – Data with which a communication or interaction partner (in short: partner) proves its identity to other partners, i.e. authenticates itself. Partners can be individuals, devices and software processes. The other partners use authentication criteria to verify the identity accordingly. Authentication data can be a user's user name and password, for instance, which the user uses when logging on. Authentication data for devices, for instance, can be a certificate and an asymmetric key pair.

Authentication criteria – Criteria used to verify the identity of communication and interaction partners; these partners can be individuals, devices or software processes. A table of user names and password hashes, for instance, includes authentication criteria to verify the passwords of individual users. The user names are the identity in this case. So-called trust lists can implement authentication criteria. These are lists that include trusted certificates, issuer certificates, and revocation information to verify certificates. Individual certificates are used as proof of identity for communication partners.

Integrator – A system manufacturer who builds systems using components and lends them to operators or has them leased by the operator. In this scenario, the integrator takes over the maintenance of the system.

Trust list – Sets of certificates that are used by a component to collect the certificates of communication and interaction partners (together: partners). See also authentication criteria: Trust lists are therefore a specific type of authentication criteria. In OPC UA "Discovery" (15), the term "Trust List" (also in the TrustList notation) is used for the trust list. There, a trust list contains a set of certificates that are trusted by definition. Optionally, a trust list can contain a further set of certificates that can be used to complete certificate paths from the partners' certificates to the corresponding root certificates (root CA certificates). In addition, a trust list can contain a set of revoked certificates based on revocation information.

$\mathbf{ZI_N}$ – Integrator certificates issued by the integrator to identities created by the integrator for the system and its components. The purpose of these certificates is to establish beyond a doubt during communication with the systems or components that communication is with a system created by the integrator or with a component installed by the integrator.

$\mathbf{ZB_N}$ – Certificates issued by the operator to identities created by the operator for the system and its components, so that when communicating with the system or components it can be established beyond a doubt that communication is with a system operated by the operator or with components installed in the system.

$\mathbf{ZA_N}$ – Certificates within a system issued by the system for identities created by the system, for instance, to enable a security domain within a system for secure communication by its components.

$\mathbf{ZH_N}$ – Certificates issued by the manufacturer to identities created by the manufacturer for devices also created by the manufacturer, for instance, in order to be able to determine the origin of the device without a doubt when communicating with the device.

List of illustrations

References

1. *Discussion paper "Sichere Kommunikation für Industrie 4.0".* Berlin: Plattform Industrie 4.0, 2017.

2. *Umsetzungsstrategie Industrie 4.0.* Berlin/Frankfurt: Plattform Industrie 4.0, 2015.

3. *Welche Kriterien müssen Industrie 4.0 Produkte erfüllen?* Frankfurt/Main: ZVEI, 2016.

4. *Industrial Communication Networks – Network and System Security.* IEC 62443.

5. *Informationssicherheit in der industriellen Automatisierung.* VDI/VDE 2182.

6. *IT-Security in der Industrie 4.0: Handlungsfelder für Betreiber.* Berlin: Plattform Industrie 4.0, 2016.

7. *Information technology – Security Techniques – Information Security Management System.* ISO/IEC 27000:2014.

8. *Integrität von Daten, Systemen und Prozessen als Kernelement der Digitalisierung.* Frankfurt: ZVEI, 2018.

9. *NE 153: Automation Security 2020 – Anforderungen an Design, Implementierung und Betrieb künftiger industrieller Automatisierungssysteme.* Leverkusen: NAMUR, 2015.

10. *Technical overview of "Sichere unternehmensübergreifende Kommunikation".* Berlin: Plattform Industrie 4.0, 2016.

11. *Security der Verwaltungsschale.* Berlin/Frankfurt: Plattform Industrie 4.0/ZVEI, 2017.

12. *Sicherheitsanalyse OPC UA. s.l.*: German Federal Office for Information Security (BSI), 25 April 2016.

13. *OPC Unified Architectur.* IEC 62541.

14. *Practical Security Recommendations for building OPC UA Applications.* Scottsdale, AZ: OPC Foundation, 2017.

15. OPC Unified Architecture Part 12: Discovery. *OPC Unified Architecture Specification Part 12:* Discovery Release 1.03. s.l.: OPC Foundation, 19 July 2015.

16. OPC Unified Architecture Part 4: Services. *OPC Unified Architecture Specification Part 4:* Services Release 1.04. s.l.: OPC Foundation, 22 November 2017.

17. OPC Unified Architecture Mappings. *OPC Unified Architecture Specifications Part 6:* Mappings Release 1.04. s.l.: OPC Foundation, 22 November 2017.

18. OPC Unified Architecture Part 3: Address Space. *OPC Unified Architecture Specification Part 3:* Address Space Model Release 1.04. s.l.: OPC Foundation, 22 November 2017.

19. OPC Unified Architecture Part 5: Information Model. *OPC Unified Architecture Specification Part 5:* Information Model Release 1.04. s.l.: OPC Foundation, 22 November 2017.

20. IEC/TS 62351-8. *Technical Specification: Power systems management and associated information exxchange – Data and communications security – Part 8: Role-based access control. s.l.*: International Electrotechnical Commission (IEC).

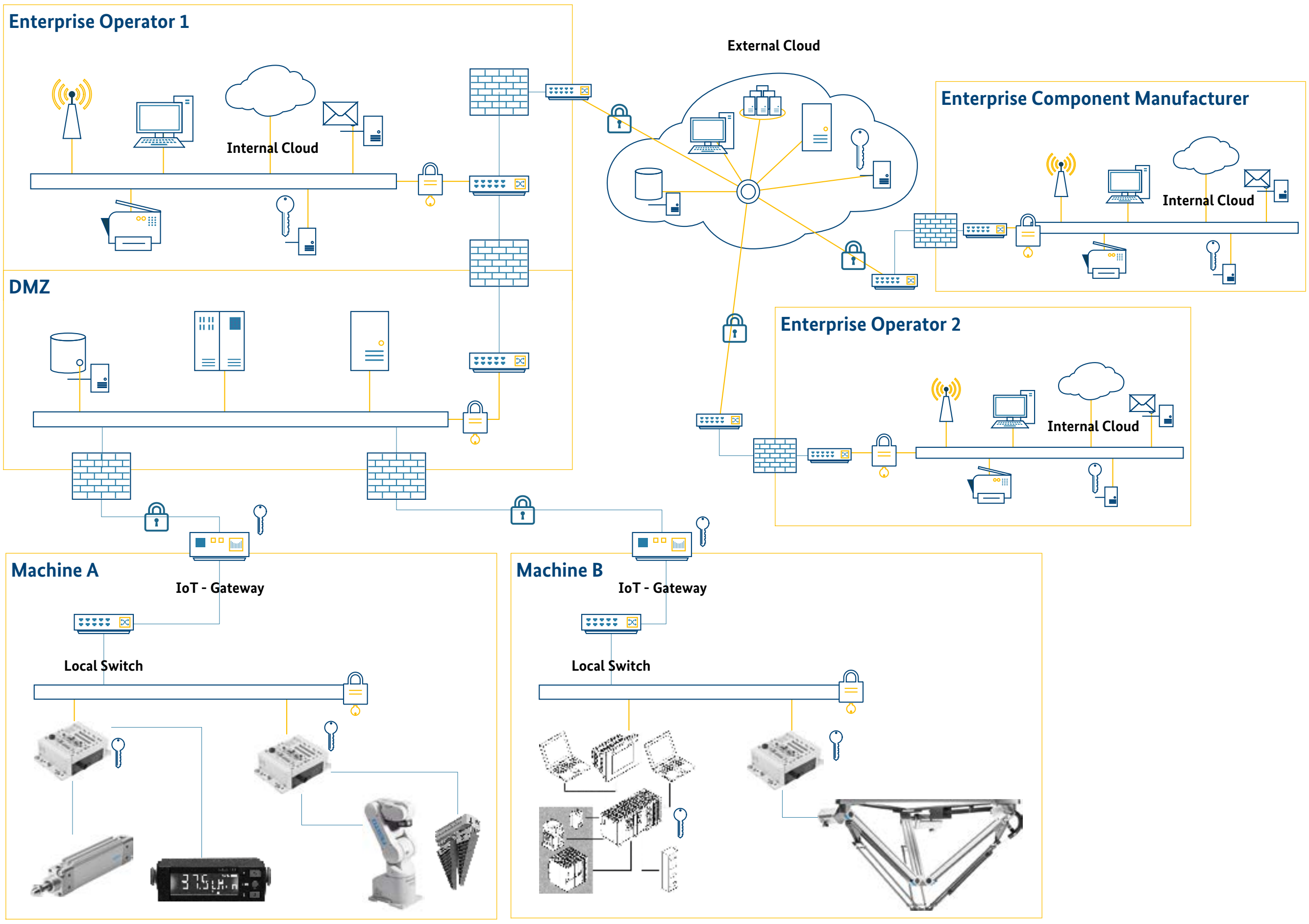
21. NIST Special Publication 800-162. *Guide to Attribute Based Access Control (ABAC) Definition and Considerations. s.l.*: National Institute of Standards and Technology (NIST), January 2014.

22. OPC Unified Architecture for Devices. *OPC UA Unified Architecture for Devices Companion Specification Release 1.01. s.l.*: OPC Foundation, 25 July 2013.

Appendix: Collaborative Factory

Figure 8 shows the overall “Collaborative Factory“ scenario. This scenario describes interaction between the various stakeholders in connected production.

Figure 8: Overall “Collaborative Factory” scenario



Source: Plattform Industrie 4.0

Operator

A factory operator, “Operator 1” in this case, operates a production facility in which machines from various suppliers are used. This example uses a tried-and-tested structure. The company’s processes are linked by a central infrastructure, i.e. the “enterprise” network. A security zone (demilitarized zone “DMZ”) is set up between the enterprise network and the production facilities, which securely separates the two parts. The DMZ and the connections passing through it (indirect access via DMZ systems, direct access by the DMZ, etc.) must be designed on the basis of the requirements for communication and security.

Machines in the "Operator Model"

In this example, the machines installed in production are to operate in the "operator model". They do not belong to the factory operator, but to the specialised service providers or the machine manufacturers themselves. In the related business model, "pay per use" could be the option. The suppliers take over maintenance and optimisation for the factory operator. This model is interesting for the present analysis since the given ownership and responsibility relationships mean that the factory operator does not have full responsibility and control over the machines, so that security domains have to be examined on a company-spanning basis.

Collaboration

The most important requirement, of course, is that the machines from the various suppliers operated in the factory must work together in order to achieve the economic goals pursued by the factory operator. This means that full interoperability of all systems and assets involved is essential.

Cloud services

The offerings by external companies to optimise business processes are represented by the "External Cloud". These offerings include the services of machine suppliers as well as other possible offerings by other providers. In this respect, the "External Cloud" symbolises external services outside the area of the factory operator and can comprise several independent offerings.

Other companies involved

In this example, the providers of the machines and the corresponding services are the relevant partners. Other offerings could, for example, come from the manufacturers of the components installed in the machines.

In principle, it should be noted that the service providers will not only look after one factory operator, i.e. "Operator 1". Just as the factory operator uses the services of multiple providers, the service providers, for their part, will support other factory operators. In terms of the model, this means that service providers process data and information from competing factory operators.

AUTHORS

Carsten Angeli, KUKA Roboter GmbH | André Braunmandl, Bundesamt für Sicherheit in der Informationstechnik | Kai Fischer, Siemens AG | Torsten Förder, PHOENIX CONTACT Software GmbH | Prof. Dr. Tobias Heer, Hirschmann Automation & Control GmbH | Dr. Detlef Houdeau, Infineon Technologies AG | Dr. Lutz Jänicke (Leitung), PHOENIX CONTACT GmbH & Co KG | Dr. Christian Krug, Geschäftsstelle der Plattform Industrie 4.0 | Fabian Mackenthun, NXP Semiconductors Germany GmbH | Jens Mehrfeld, Bundesamt für Sicherheit in der Informationstechnik | Andreas Pfaff, Mitsubishi Electric Europe B.V. | Tobias Pfeiffer, Festo AG & Co. KG | Uwe Pohlmann, Fraunhofer-Institut für Entwurfstechnik Mechatronik | Martin Regen, Microsoft Deutschland GmbH | Andreas Teuscher, SICK AG | Klaus Theuerkauf, Institut für Automation und Kommunikation e.V. | Dmitry Tikhonov, Assystem Germany GmbH | Thomas Walloschke, Fujitsu Technology Solutions GmbH

This publication is a joint result of the working groups "Security of Networked Systems"
and "Reference Architectures, Standards and Norms" (Plattform Industrie 4.0).

www.plattform-i40.de