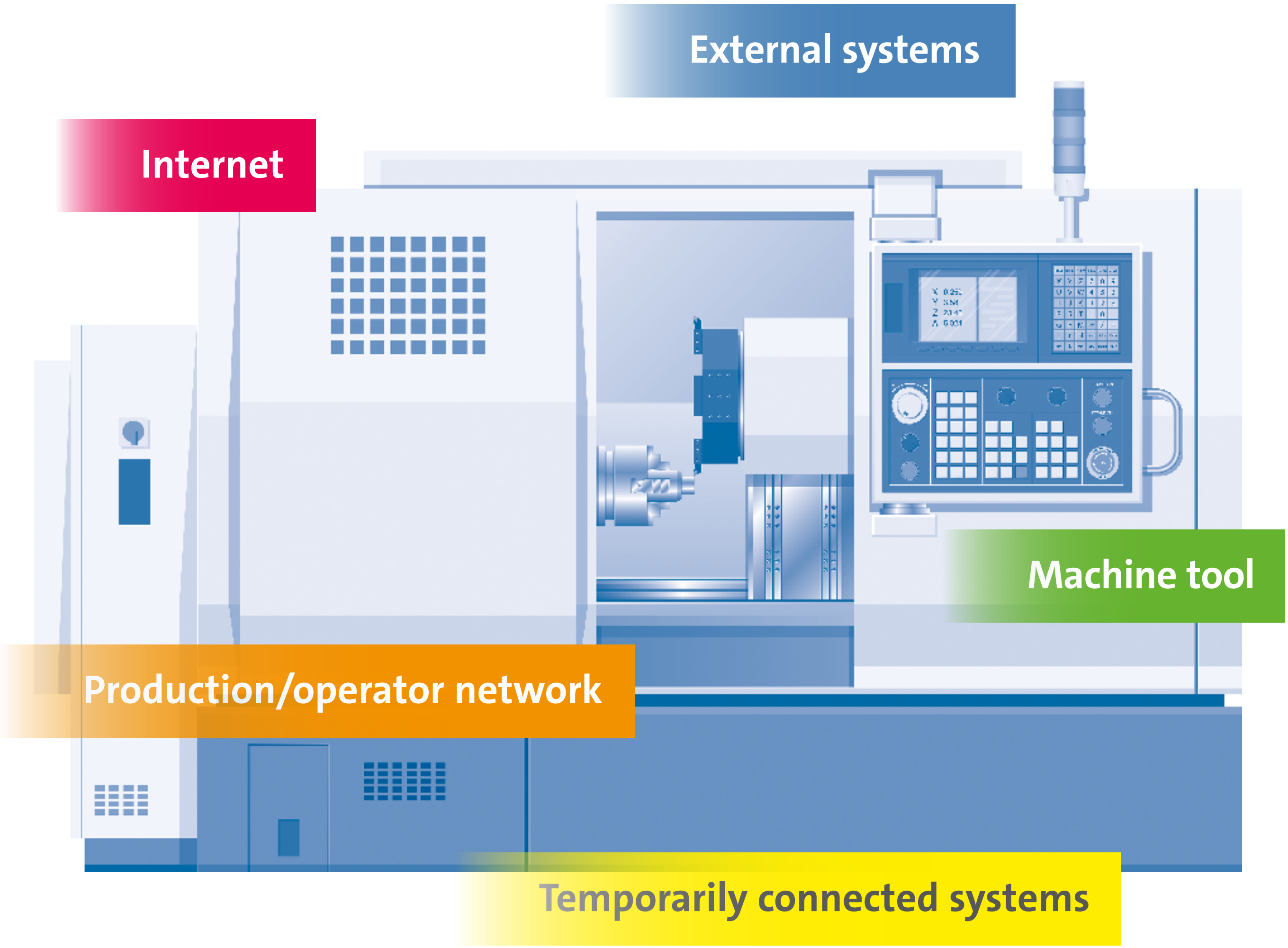
German Machine Tool Builders’ Association

Security for machine tools

A guide to methodical implementation, considerations over the entire service life and suggestions for the treatment of existing machines.

Guide for manufacturers



1. Introduction

1.1 Motivation

The digital transformation of manufacturing, especially of machine tools and plants, is steadily advancing. Control components that were previously operated as stand-alone solutions are being networked company-wide or directly connected to each other via the internet and interact with software services in the cloud. This creates so-called cyber-physical systems.

These systems are increasingly being targeted by hackers, as they are often very easy targets for cyber attacks. Malware (such as “Mirai”, “Hajime”, “WannaCry” or “Petya”) enables attackers to quickly and significantly impair the availability of plants and machines; production processes come to a standstill, resulting in economic losses costing millions. In addition to the extorted money payments, companies often also suffer great damage to their image.

The operator himself can make an initial and important contribution to secure machine operation. Information on and suggestions for this have already been presented elsewhere *(⟶ [1]*)*. In order to be able to adequately combat threats, plant and machine manufacturers will also have to attach much more importance to security in the future – both in the construction of the machines and in their operation. “Security by design” is a method that has been successfully applied in software development for many years. Transferred to machine and plant construction, it is applied in the international standard IEC 62443.

Previous publications offer a good basis for dealing with the complex issue and the standard, but they often refer to solutions from plant engineering and do not include the special requirements of machine tools. This document expands the scope of consideration to the entire development process in accordance with the IEC 62443 series of standards and is specifically designed for machine tools.

1.2 Target audience

This document is primarily aimed at manufacturers of machine tools and production equipment for manufacturing.

These are primarily companies in the mechanical engineering sector that develop, sell, bring to the market and maintain the corresponding production systems in the course of their life cycle.

1.3 “IT” vs. “OT”

In many considerations of IT security of the production systems in focus here, the term “OT” (“Operational Technology”) is used in distinction to “IT” (“Information Technology”). The term has become established to indicate the technological and functional differences between traditional IT systems (such as in the office environment) and the environment of industrial control systems, the so-called “IT in non-carpeted areas”.

Accordingly, OT primarily refers to hardware and software for the control, regulation, monitoring and/or surveillance of industrial plants, machines, assets, processes, or events. A comparison of IT/OT based on ⟶ *[2]* is shown in ⟶ *Fig. 1*.

*) *References in square brackets, see page 37*

Fig. 1:
Differences in the requirements for IT (office environment) and OT (production environment)
Source: ⟶ [2]

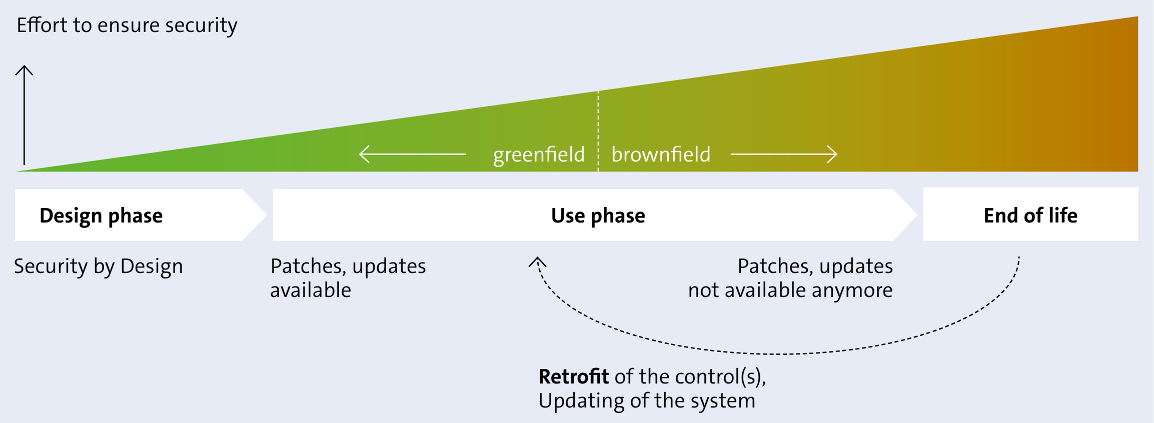
	IT-Systems	OT-Systems
Lifetime	3 to 5 years	5 to >20 years
Development cycle	Short	Adapted to lifetime
Patch management	Often, daily	Rarely, must be released by system integrator/component manufacturer
Availability	Short failures tolerated	24/7 IEC 62443 defines security objectives in part 1-1, where availability is defined as the highest security objective.
Time dependency	Delays accepted	Critical IEC 62443 defines security objectives in part 1-1, where real-time capability is specified as a millisecond range.
Investment	Approx. 1,000 €	Many € thousands up to € millions

It becomes clear that there are considerable differences, which in turn require a tailored approach.

1.4 Scope

Following an introduction to the topic with a clarification/definition of the most important terms in ⟶ *Chapter 2*, the focus of this publication follows in ⟶ *Chapter 3*: This is on providing a methodology for safeguarding the development and design process. For the most part, best practices are not mentioned in favour of a manageable scope and broad technical coverage. The main intention is to enable the machine manufacturer himself to carry out such a secure development process: The best starting conditions for efficiently achieving a high degree of protection exist during the greenfield design phase of new machines ⟶ *Fig. 2*. The effort increases steadily along the time axis in the later phases of use, which is why an initial effort represents the best possible basis for a secure system.

Fig. 2:
Scope of application of the present document


Effort to ensure security
greenfield
brownfield
Design phase
Use phase
End of life
Security by Design
Patches, updates available
Patches, updates not available anymore
Retrofit of the control(s), Updating of the system

Due to the expected long service life of machine tools – combined with the certainty that most of the customer's machine park consists of existing machines – a discussion of the development process alone is not sufficient, which is why the further service life *(⟶ Chapter 4)* and also the “brownfield” *(⟶ Chapter 5)*, i.e. the existing machines, are also taken into account. For these operating phases, information is provided that can be understood as the continuation of an overall concept for secure operation, but can also be implemented as individual measures, for example in existing buildings.

2. Technical background on security

2.1 Terms and definitions

2.1.1 Asset

Physical or logical object, generally worthy of protection, that is owned or under the care of an organisation and that has either perceived or actual value to the organisation.

Note: An asset can be tangible or intangible.

Examples: Component, control, know-how, data, NC programs

2.1.2 Availability

Ensuring that access to data is guaranteed within an agreed time frame.

2.1.3 Conduit (Security zone transition)

Logical grouping of communication channels connecting two or more zones to which common IT security requirements apply.

Note: A conduit may cross a zone as long as the IT security of the channels running in the conduit is not affected by the zone.

2.1.4 Confidentiality

Ensuring that information is not disclosed to unauthorised persons, processes or devices.

2.1.5 Device management (Device administration)

This is understood to mean the administration of an IT device (usually carried out from a central location, e. g. a service provider). This includes both the monitoring of hardware (utilisation, fan speeds, free memory, etc.) and the maintenance of software (updating the software status following security patches or function enhancements, installing/deleting software, etc.), as well as central management tasks, e. g. clear identification and cataloguing of devices, automatic re-setting when devices are replaced.

2.1.6 Embedded component

An embedded component is one that is used with the product. In the case of software, these are e. g. the libraries used. Embedded components can contain other embedded components (transitive integration).

2.1.7 Gross risk

Risk before application of risk treatment measures.

2.1.8 HMI (Human Machine Interface)

HMI refers to the user interface of a machine (includes hardware and software).

Note: In the context of this document, usually the software system for operating the machine.

2.1.9 IACS (Industrial Automation and Control System)

Industrial automation system(s).

2.1.10 Integrity

Quality of a system to ensure

- the logical correctness and reliability of the operating system,
- the logical completeness of the hardware and software to implement the protection mechanisms and
- the consistency of the data structures and the correctness of the stored data.

It must not be possible to change, manipulate or delete data without authorisation; any changes made must be traceable.

2.1.11 Interface

Any type of physical or software interface.

2.1.12 Level of protection

Security level.

2.1.13 Measure (Countermeasure)

Activity, device, procedure or method that reduces a threat, vulnerability or the consequences of an attack by reducing the potential damage or by detecting and reporting it so that corrective action can be taken.

2.1.14 Net risk / residual risk

Risk after application of measures.

2.1.15 Protection classes

Protection classes are the classification of the need for protection based on (potential) damage (e. g. financial or reputational) or hazards to enable uniform treatment.

Note 1: By grouping in protection classes, a simple and objective assessment of damage levels and consequences can be obtained.

Note 2: In this document, the protection classes "low", "medium" and "high" are applied.

Note 3: Other classifications are possible.

2.1.16 Protection goal

Protection goals are statements or definitions of the minimum security level to be achieved.

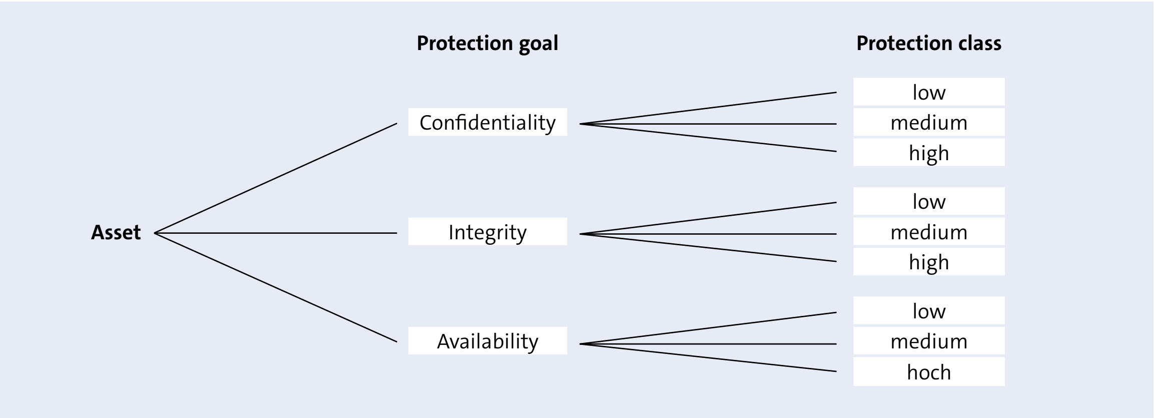
Note: In this document, the protection goals "confidentiality", "integrity" and "availability" are applied.

2.1.17 Protection needs

The protection requirement of an object is based on the extent of damage that can occur if its functioning is affected by the protection objectives not being met.

The protection requirement must be determined for each asset. For each of the protection goals, the protection class must be assessed, i.e. whether an asset is to be classified as low, medium or high with regard to confidentiality, integrity and availability, see also ⟶ *Fig. 3*.

Fig. 3: **Protection needs of an asset**



2.1.18 Risk

Damage expressed as the likelihood that a particular threat will exploit a particular vulnerability with a particular consequence.

2.1.19 Security Zone (Zone)

Grouping of assets to which common security requirements apply.

Note 1: A zone has a clear boundary to other zones.

Note 2: The security of a zone may be ensured by mechanisms within the zone or at the zone boundary, or by a combination of these.

2.1.20 Security zone transition (Conduit)

⟶ *see chapter 2.1.3*

2.1.21 Secure Development Lifecycle (SDL)

All phases of a development process, from the analysis and definition of security requirements to product design, implementation, verification and validation, handling of security issues throughout the entire product life cycle until the end of the use phase.

2.1.22 Security context

Description of the OT security environment in which the machine tool is used and which is assumed for the intended operation.

2.1.23 SL (Security Level)

Measure of confidence that the machine tool is free from security hazards and will function in the intended manner.

Note 1: Security vulnerabilities may arise during the development of the IACS, be introduced at any time during its lifecycle, or arise from changing threats.

Note 2: Security vulnerabilities that arose during development may not be detected until long after first use, for example, an encryption technique was insufficiently implemented or user account management guidelines were insufficient, such as not removing old user accounts.

Note 3: Introduced vulnerabilities may be the result of a patch or changed guidelines that open up a new vulnerability.

2.1.24 SL-A (Security Level Achieved)

Achieved security level.

2.1.25 SL-C (Security Level Capability)

Achievable security level.

2.1.26 SL-T (Security Level Target)

Target security level.

Note: The security level to be achieved is usually a result of the threat/risk analysis.

2.1.27 SuC (System under Consideration)

System under consideration.

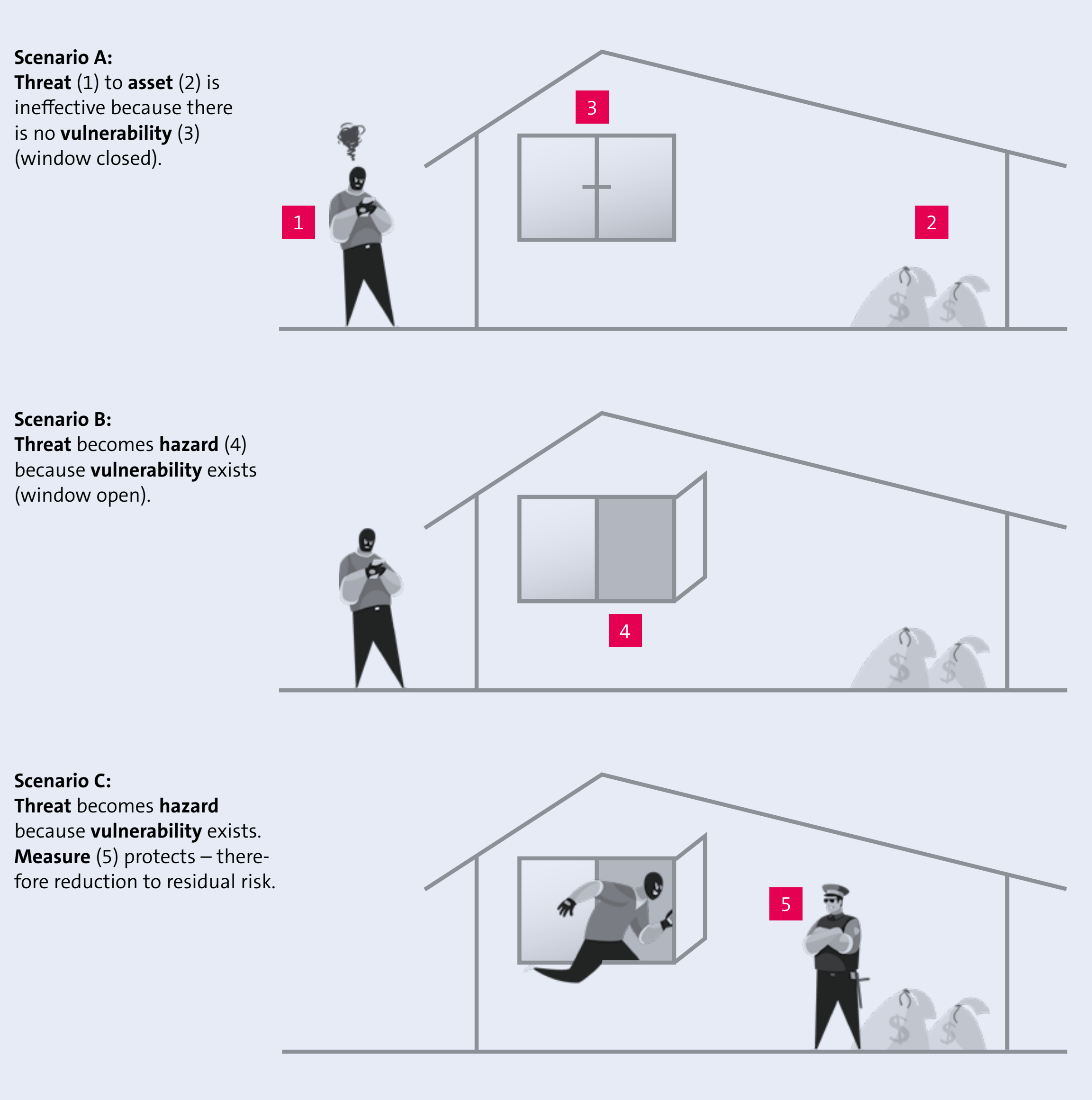
Note: In this document, the machine tool is the system under consideration.

2.1.28 Threat and hazard

Potential to compromise security or cause harm as the result of an action, capability, event or circumstance. The situation is illustrated in ⟶ *Fig. 4*.

A threat represents an abstract, theoretically possible danger. An example would be the existence of numerous malware programs on the Internet which, for example, perform port scans to find “victims”. The threat usually has no effect at first due to appropriate protective measures, such as a firewall. However, it becomes a real threat the moment a security vulnerability is exploited (e. g. a gap in the firewall) and the threat thus becomes a concrete danger for an asset. Even this does not necessarily mean that a damaging event will occur, but it significantly increases the probability or risk.

Fig. 4: **Relationship between threat and hazard**



Scenario A:
Threat (1) to **asset** (2) is ineffective because there is no **vulnerability** (3) (window closed).

Scenario B:
Threat becomes **hazard** (4) because **vulnerability** exists (window open).

Scenario C:
Threat becomes **hazard** because **vulnerability** exists. **Measure** (5) protects – therefore reduction to residual risk.

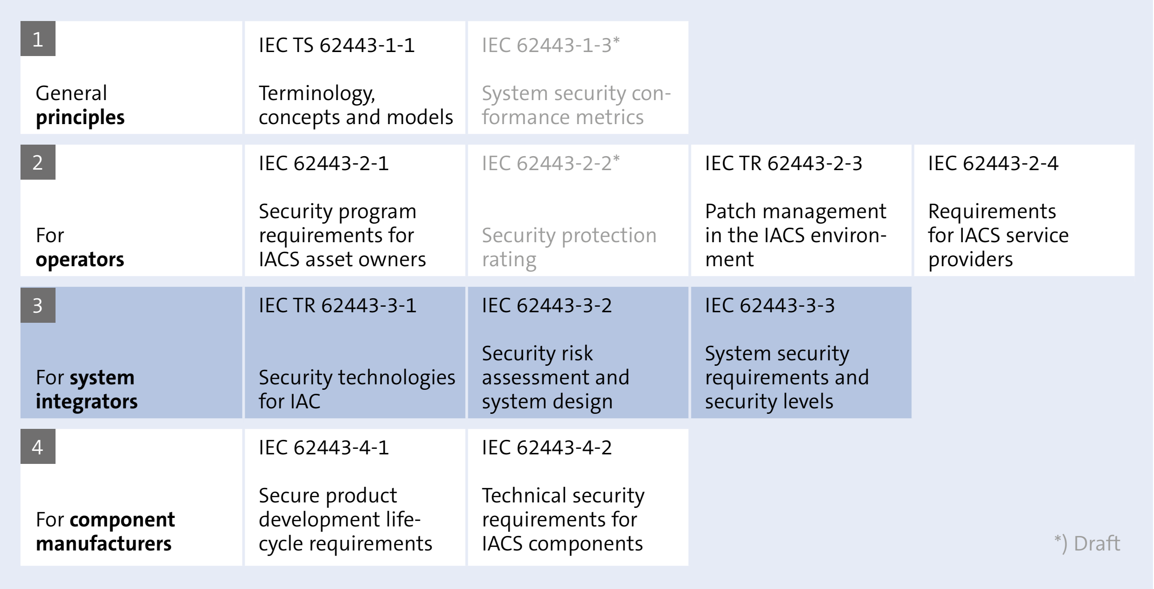
2.1.29 Vulnerability

Error or weakness in the design, implementation, operation or management of a system that can be used to violate the integrity or security policies of the system.

2.2 The IEC 62443 series of standards and the role of machine tool manufacturers

2.2.1 Classification of the standards series

IEC 62443 is a series of standards dealing with IT security for Industrial Automation and Control Systems (IACS). It was originally developed as a standard for automation technology in the process industry, but now covers all industrial sectors from discrete manufacturing to distributed supply systems for electricity, oil, water and gas. This series of standards has established itself as the benchmark for considering industrial security over the entire life cycle of automation solutions. It is regarded as the central standard for security, similar in importance to ISO 13849 for functional safety.



1 General **principles**	IEC TS 62443-1-1 Terminology, concepts and models	IEC 62443-1-3* System security conformance metrics		
2 For **operators**	IEC 62443-2-1 Security program requirements for IACS asset owners	IEC 62443-2-2* Security protection rating	IEC TR 62443-2-3 Patch management in the IACS environment	IEC 62443-2-4 Requirements for IACS service providers
3 For **system integrators**	IEC TR 62443-3-1 Security technologies for IAC	IEC 62443-3-2 Security risk assessment and system design	IEC 62443-3-3 System security requirements and security levels	
4 For **component manufacturers**	IEC 62443-4-1 Secure product development lifecycle requirements	IEC 62443-4-2 Technical security requirements for IACS components		

*) Draft

Fig. 5: **Overview of the IEC 62443 series of standards (representation based on IEC 62443-3-1)**

This is a comprehensive series of standards whose individual components address three different target groups. According to the overview shown in ⟶ *Fig. 5*, the individual parts can be assigned as follows:

1 IEC 62443-1 describes **general principles** and concepts and thus provides a reference basis for all other parts of the series of standards.

2 IEC 62443-2 is primarily aimed at the **operators of** industrial plants – from the point of view of a machine tool manufacturer, therefore, to the buyers and users of the machines.

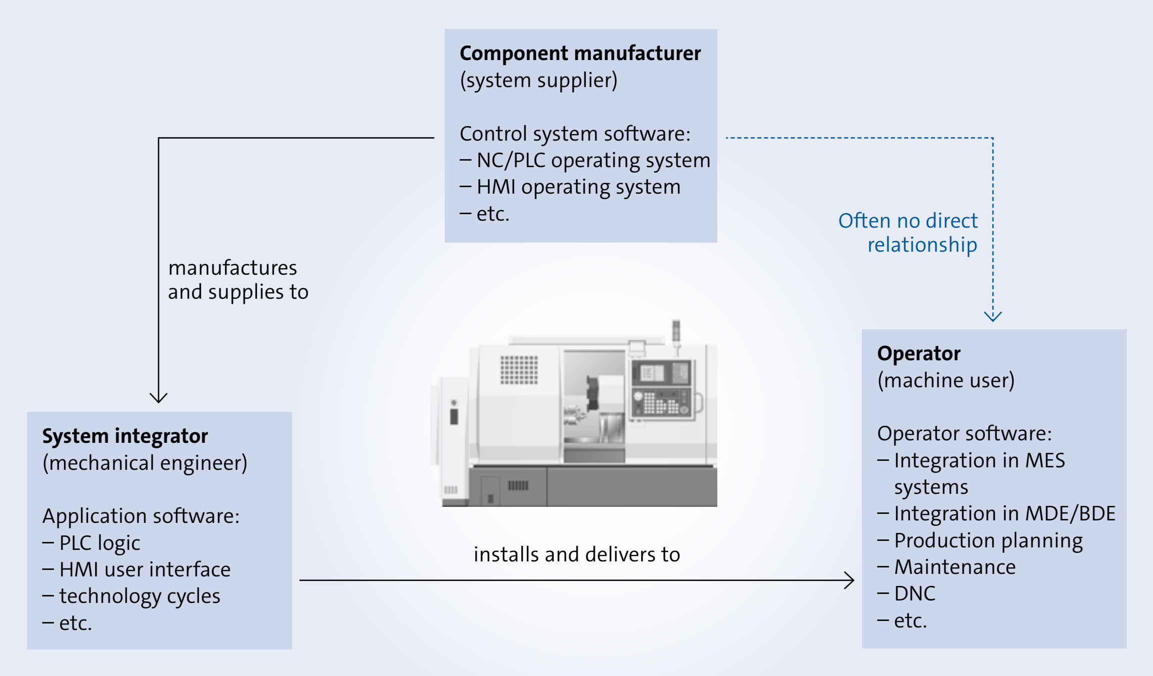
3 IEC 62443-3 addresses the **system integrators and equipment manufacturers** – this part is of **central importance for the manufacturers of machine tools**, as they primarily act as system integrators as defined in the standard.

4 IEC 62443-4 is focussed on the **manufacturers of machine components** (i.e. primarily on the suppliers, for example of control systems). There is a strong overlap in terms of content here, particularly with Part 3, as machine tool manufacturers often also manufacture components for their own systems and may also sell components individually.

2.2.2 Role of the machine tool manufacturer

Due to the history of IEC 62443 in the plant engineering environment, there is no completely clear separation for a machine tool manufacturer between the role of a system integrator and a component manufacturer. However, there is a clear focus: a machine building company as a manufacturer and developer of machine tools primarily assumes the role of a **system integrator** in this set-up, as a large number of components from different suppliers are combined to form an overall system, which is then regarded as a machine from the operator's point of view (especially from an IT perspective). According to this assignment, ⟶ *Fig. 6* shows the relationships between the three roles of component manufacturer, system integrator and operator.

Fig. 6:
Interaction of the roles of manufacturer, integrator, operator



Sometimes a machine tool builder finds himself in the role of the component manufacturer, if he takes appropriate action himself for his own (or other) machines. According to IEC 62443, **however, this role primarily concerns the IT perspective**; the in-house production of purely mechanical assemblies such as frame components, guides, etc. is therefore not the focus here. The case is different, especially with the widespread **in-house developments in the area of software** (e. g. HMI systems, technology cycles, interface software). Here, as defined in IEC 6443, the machine tool builder clearly has the role of the manufacturer and thus, as set out in Part 4 of the standard, also is responsible for corresponding secure implementation of his own development. This also applies to "combined" systems of hardware and software, such as intelligent spindles or component-integrated sensor technology, insofar as these form independent IT components.

In principle, the role model described also applies to component suppliers. For example, a control system manufacturer will in turn also use (software) components from third parties: from their point of view he is then the system integrator. If an operator develops his own software components which are used on the machine, he is also responsible under the terms of the standard as a manufacturer or system integrator.

2.3 Protection strategy

Machine tools are complex technical systems consisting of **several subsystems** (from a control point of view e. g. HMI, NC control, connectivity gateways) and are connected to a number of aggregates (such as loading magazines, coolant systems, hydraulic aggregates, etc.). In addition, the machines are regularly **adapted by and with the customer and often supplemented with third-party enhancements** (e. g. tool breakage monitoring) – and are often further modified in the course of their use over the years.

This complex use case means that **isolated, one-time protection of the machine (OT security) does not provide sufficient protection**.

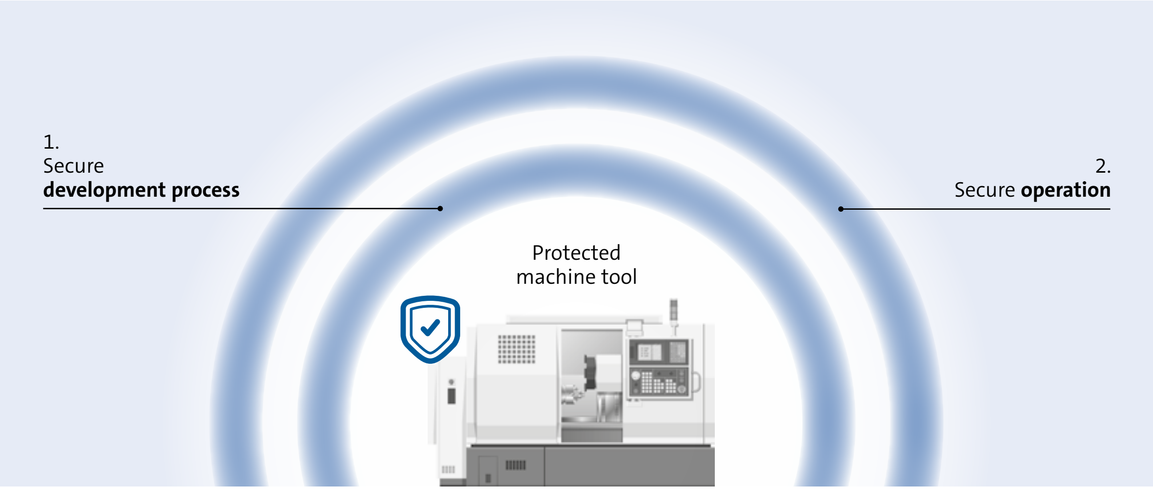


Fig. 7: **Protection strategy for machine tools**

Rather, a combined protection strategy ⟶ *Fig. 7* is required:

1. A **development process already geared towards a high security standard**, which is designed using an appropriate methodology.

- Such a methodology to ensure a secure development process is proposed in this document following IEC 62443-4-1 and is described below *(⟶ Chapter 3)*.
- This **inner protective shell** is formed by the properties secured in the development process – starting with the specification of the security requirements and guidelines, the methodology for “security by design”, through the secured implementation with subsequent verification.
- The aim is to **secure the machine** as completely as possible from the outset.

2. **Accompanying measures during the entire life cycle** of the machine, also after delivery to the customer, in order to be able to maintain a high security standard throughout the entire service life.

- The **outer protective shell** is formed by secure operation, i.e. measures taken at runtime or during the entire (!) lifetime of the machine.
- A publication *(⟶ [1])* has already been dedicated to this aspect; further explanations on protection after delivery follow in ⟶ *chapter 4* and ⟶ *chapter 5*.

IEC 62443 defines general requirements for a secure development process for products used in industrial automation systems. The standard describes a secure development lifecycle (SDL). The SDL covers all phases of a development process, from the analysis and definition of security requirements, through product design, implementation, verification and validation, and the handling of security problems throughout the entire product lifecycle to the end of life (EoL).

Furthermore, requirements are defined for security management, which should ensure that all activities are appropriately planned, executed and documented. It must also be ensured that these activities are supported by appropriate organisational processes and that corresponding expert responsibilities are defined.

All processes and principles should help to ensure that security is already implemented in the design (“security by design”) and with the help of a staggered security concept (“defence-in-depth concept”).

It is not enough to limit oneself to securing the machine once; the **entire development process and the further life cycle** must be oriented towards a high level of OT security.

Simple measures on the operator's side have already been presented in the document **IT Security on Machine Tools** *(⟶ [1])*.

3. Methodology for initial protection of a machine

From the very beginning, the highest possible security standard should be achieved in order to make the machine as robust as possible against the expected constant changes in the machine environment during its lifetime. The methodology presented below, whose sequence of steps is shown in ⟶ *Fig. 8*, serves this purpose.

1	**Security Context**	⟶ *Chapter 3.1*
2	**Determination of a Security Level Target (SL-T)**	⟶ *Chapter 3.2*
3	**Architectural design and division into security zones**	⟶ *Chapter 3.3*
4	**Assets and protection needs**	⟶ *Chapter 3.4*
5	**Threat analysis and product risk**	⟶ *Chapter 3.5*
6	**Security requirements and measures**	⟶ *Chapter 3.6*
7	**Residual risk**	⟶ *Chapter 3.7*
8	**Documentation**	⟶ *Chapter 3.8*

Fig. 8: **Overview of an IT security-oriented development process**

In this document, this methodology is carried out **on the basis of a generic machine tool** and explained with a few selected examples; the circumstances in the concrete individual case may therefore deviate accordingly.

The first step ⟶ *Chapter 3.1* is **consideration of the security context**; in this context, the interfaces of the machine to the outside are in the foreground, as these are the main entry points for threat situations. Subsequently, a **target protection level** must be determined in the form of a defined “Security Level (Target)” ⟶ *Chapter 3.2* The third step consists of a more detailed **consideration of the system architecture** ⟶ *Chapter 3.3*, whereby areas can be combined into security zones and considered together. Then the individual **assets and their protection needs are analysed** ⟶ *Chapter 3.4*. The core of the methodology is the subsequent **threat analysis** ⟶ *Chapter 3.5*, combined with an assessment of the real risks from the perspective of the machine operators. **Measures** are then derived from the analysis ⟶ *Chapter 3.6* and the **residual risk** after implementation of these measures is assessed ⟶ *Chapter 3.7*. The **entire process must be documented** ⟶ *Chapter 3.8* and, if necessary, iterated if the residual risk is considered too high.

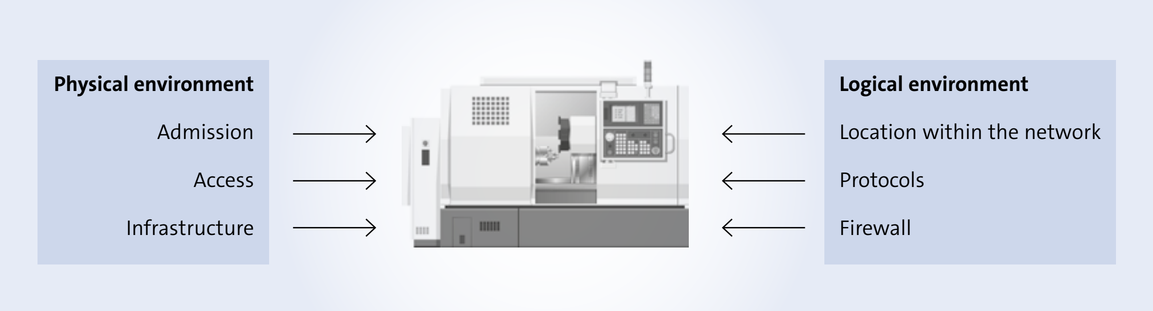
3.1 Security context

1

The first step of the method is to define the security context.

The first step is a systematic **documentation of the place of use and the security environment**. Some particularly noteworthy points are shown in ⟶ *Fig. 9*.

Fig. 9: **Physical and logical environment of the machine**



The environment is viewed from different angles for the analysis: On the one hand, the physical world with its infrastructure, the persons granted access, and access possibilities to the various machine functions. On the other hand, there is a logical classification of the machine in the "digital infrastructure".

This first step is not a detailed analysis of all subsystems of a machine, but **classification of the framework conditions to be expected in "intended operation"**. These are to be documented.

Such documentation could answer the following questions, for example:

Physical environment

- *Which group of people usually has access to the machine tool?*
  - *How are they qualified?*
  - *Are there different authorisations in this group of people?*
  - *Are different authorisations technically secured (e. g. by means of passwords)?*
- *Is there a possibility to control physical access (e. g. via access restrictions, time recording)?*
- *Is the machine tool technically integrated into a plant network or does it stand alone?*
- *Does a failure of the system directly affect downstream systems (e. g. interlinked handling systems) or are there decoupling points here?*

Logical environment

- *Is the machine located in a compartmentalised network segment, e. g. a production network?*
- *Is there separate partitioning of the machine tool by a firewall?*
- *With which target points/systems does the machine communicate (such as production systems like MES, office IT systems like ERP, cloud services)?*
- *Which protocols does the machine use to communicate with the outside world (e. g. SMB for file sharing, OPC UA for operational data)?*

For further consideration, it is crucial at this point to make a clear **distinction between the machine tool and the identified external systems** (physical and logical) in order to define a sensible framework for the analysis. As a point of reference, interfaces or protocol end points define the transfer points at which the machine/plant's observation area ends.

Example: Physical environment

- A machine tool is freely accessible within a factory building, but only a limited group of people has access to the building. It can therefore be assumed in the further consideration that access to the plant is regulated accordingly. Access control to the hall itself is clearly outside the "machine" system boundary.
- Access to certain machine functions is protected by a password. The proper function and scope of this password protection in the machine control system is included in the consideration; the organisational administration of the password at the customer's is not included and can only be provided in the form of recommendations if necessary.

Example: Logical environment

- A machine communicates with other systems in production, e. g. an MES, by means of an OPC UA server. The OPC UA server contained in the machine is included in the consideration, the system that accesses it is not.
- The machine communicates with a remote maintenance system via a cloud connection. The endpoint in the machine is included in the consideration, the cloud service itself is not.

As a result of this first methodological step, a document is available in which the **framework conditions and delimitations** of the machine are listed as described above.

Example: Generic machine tool

The generic machine tool shown in ⟶ *Fig. 10* explains these different access paths in more detail. It will be considered in more detail below.

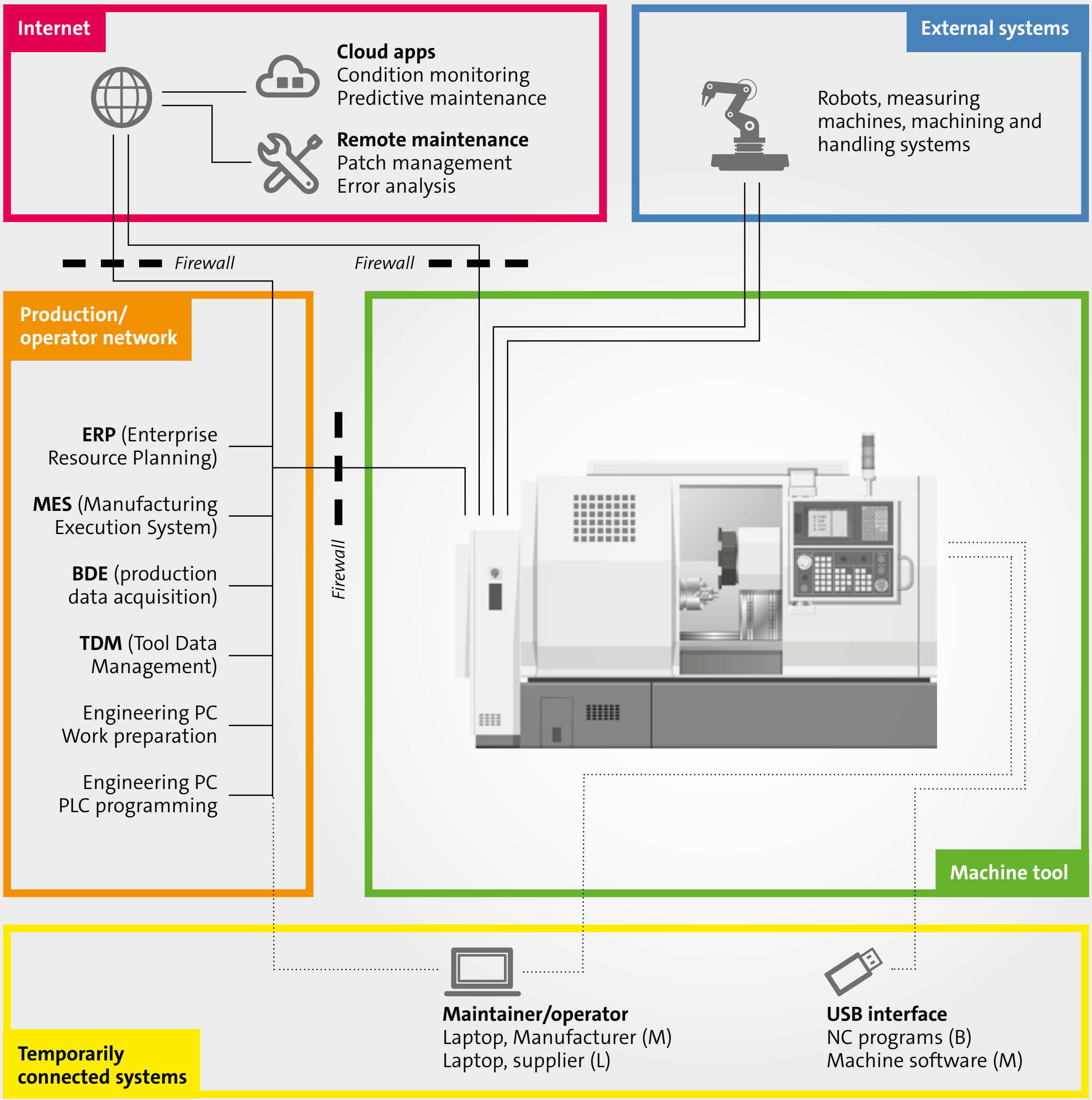

Internet
Cloud apps
Condition monitoring
Predictive maintenance
Remote maintenance
Patch management
Error analysis
External systems
Robots, measuring machines, machining and handling systems
Firewall
Firewall
Production/ operator network
ERP (Enterprise Resource Planning)
MES (Manufacturing Execution System)
BDE (production data acquisition)
TDM (Tool Data Management)
Engineering PC Work preparation
Engineering PC PLC programming
Firewall
Machine tool
Temporarily connected systems
Maintainer/operator
Laptop, Manufacturer (M)
Laptop, supplier (L)
USB interface
NC programs (B)
Machine software (M)

Fig. 10: **Generic machine tool**

The example machine is connected via LAN to a company network in which various production-relevant server services are available (e. g. MES) and via which the NC control/PLC can also be configured (engineering PCs). The machine tool also communicates with other machines (e. g. as part of a production cell or e. g. with measuring machines, industrial robots), either directly via a fieldbus connection or also via the company network. The connection to cloud services and services for remote maintenance is made via internet access. Furthermore, the local control point is indicated with various functionalities, e. g. data backup and access by service technicians. Physical, USB interfaces or additional Ethernet ports are usually available here.

The assumed OT security environment is also described, which is assumed for the intended operation. This "security context" describes both the physical and the logical place of use, e. g. within a network structure.

Relevant for this are:

- The operator's **network infrastructure** or the location in the network where the machine is to be operated, e. g. whether the machine is located in a separate, secured network section or whether there is a connection to the internet.
- The physical security provided by the environment, including access regulations.
- Possible **effects on the environment**, e. g. loss of production, environmental influences.

An essential part of defining the system under consideration ("SuC") is determination of the components that belong to the area of application or responsibility. **These components must be considered in detail in the security-oriented development process by means of a threat analysis**. The area outlined in green in ⟶ *Fig. 10* shows the delimited area of responsibility of the manufacturer (the machine tool). The subsequent threat analysis takes into account data flows to these components, but not the demarcated components themselves. Here in the example, these are server services and engineering PCs, as well as other machines with which communication takes place via M2M, and the cloud connections. Furthermore, the technical connection to the various servers, e. g. via plug-ins as in the case of the ERP system or via an offered application programming interface (API) as in the case of shop floor data collection, is delimited from the area of responsibility.

The **assumptions made regarding the security context**, which are assumed for the intended operation of the machine tool, must be documented and are the basis for a common understanding of the responsibilities of the manufacturer, integrator and operator.

3.2 Determination of a Security Level Target (SL-T)

2

The second step of the method is to determine the security level.

To evaluate the desired/required security level of a security zone (defined area), IEC 62443-1-1 or -3-3 describes different characteristics according to the following definition in ⟶ *Table 1*:

Table 1: **Security level according to IEC 62443 1-1/3-3**

Security Level	Description	Examples
SL-1	• Protection against unintentional, accidental misuse	• An operator accidentally changes machine parameters • A programmer accidentally deletes NC programs
SL-2	• Protection against intentional misuse • Simple means • Little effort • General competences • Low motivation	• A maintenance worker's computer is infected by malware, which is transferred to the machine • An attacker uses easily accessible malware that exploits known vulnerabilities in operating systems
SL-3	• Protection against intentional misuse • Technically sophisticated means • Moderate effort • Automation-specific competences • Moderate motivation	• Staggered access: A security vulnerability in one system (e. g. PLC) is used to penetrate other systems until the target system is reached (e. g. the company ERP). • Extortion of companies by means of malware via targeted manipulation or interference with production
SL-4	• Protection against intentional misuse • Technically sophisticated means • Considerable effort • Automation-specific competences • High motivation	• Intelligence activities (as used e. g. with "Stuxnet") • Organised crime activities with the aim of extorting certain companies • Instrumentalisation of large resource networks (cloud, botnets) • Customised developments for special systems of individual customers, for example in the defence sector

The security levels are assigned to individual functions in the course of the security assessment, whereby IEC 62443 differentiates here between foundational requirements (FR), security requirements (SR) and requirement enhancements (RE).

Security levels can be assigned to individual functionalities to varying degrees. In many cases it is appropriate (or a customer requirement) to **agree on a security level for the entire system**, e. g. SL-2.

At this point, reference should be made to **Guide IEC 62443 for mechanical and plant engineering** (⟶ *[3]*) published by the VDMA, which specifies SL-2 for the entire plant and describes in detail the procedure according to IEC 62443 3-3.

In addition to their "level" *(⟶ Table 1)*, security levels are also further specified in IEC 62443 according to their intended application. A distinction is made between:

- SL-T (Security Level Target): Target security levels can be determined during the PDCA (Plan-Do-Check-Act) cycle through a risk assessment. They represent the **security level to be achieved**.
- SL-C (Security Level Capability): Achievable security levels represent the **maximum level** that can be achieved using security measures.
- SL-A (Security Level Archieved): **Achieved security levels** are used to assess whether security measures fulfil their purpose and thus the specifications of the target security level.

As a result of the second step, it has now been determined which **protection requirement** the machine tool should fulfil.

In the example machine shown in ⟶ *Fig. 10*, it is assumed that all parts of the system should have a common security level. This security level is defined as the target level (security level target, SL-T).

3

3.3 Architectural design and division into security zones

The third step of the method is to analyse the architecture and divide it into security zones.

Whereas the machine tool was previously considered as a complete system, it is now subdivided into individual security zones for further consideration in accordance with the standard clause IEC 62443-3-2. These, in turn, are in contact with each other through security zone transitions.

When grouping into security zones and security zone transitions, the following aspects must be taken into account:

- Plant systems (i.e. systems belonging to the machine tool) and enterprise or business systems (i.e. higher-level systems of the production operation) must be divided into different security zones.
- Components or assets that are critical in terms of their functional safety (safety-critical) belong in their own security zones.
- For components that are only temporarily connected to the system under consideration, a separate security zone should be established.
- A separate security zone is established for wireless devices and wireless communication.
- Devices that obtain access via untrusted networks (e. g. remote access) also belong in their own security zone.

☑ As a result of the third step, a **high-level architecture** and **classification into different security zones** based on this architecture are available.

A high-level architecture image of the exemplary/generic machine tool (WZM) shown in this document from ⟶ *Fig. 10* forms the starting point of the further analysis and is shown in ⟶ *Fig. 11*.

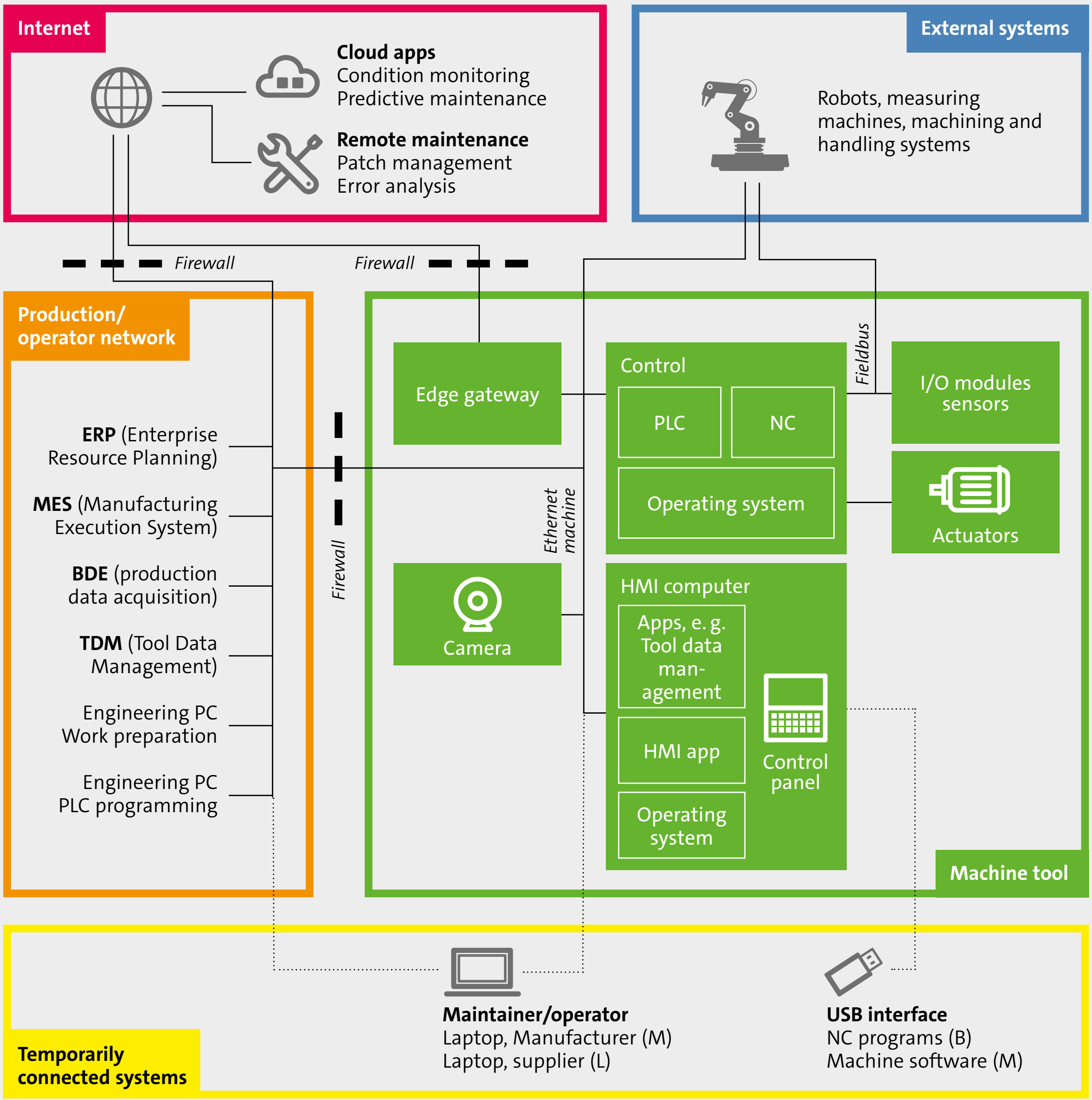


Fig. 11: **Detailed overview showing components and connections**

The following security zones or security zone transitions are distinguished in the machine tool under consideration:

- The **core machine tool with its components** (outlined in green) such as NC control, HMI PC and other assemblies, all of which are also related to functional safety.
- The **production or operator network** (outlined in orange) with all services available and possibly used therein, such as ERP, MES, BDE, etc.
- **Temporarily connected systems** (outlined in yellow), these include maintenance computers and devices used by the operator such as USB sticks for backups
- **Untrusted external networks** (outlined in red), e. g. Internet connections
- **External systems** (outlined in blue), e. g. loading magazines, handling systems, robots, measuring machines

The structure of several system components and the various interfaces through which they communicate with each other and with the periphery can be clearly seen. Included are all APIs and plug-ins, the access point of the service technicians of the manufacturer and integrator as well as their laptops, but not of the operator. In addition, the remote maintenance options offered via an internet connection, the cloud-based apps (e. g. for preventive maintenance), the local access options via USB interface and the backup functionality are included in the scope.

4

3.4 Assets and protection needs

In the fourth step of the method, the individual assets and their respective protection needs are considered.

3.4.1 Definition and examples of assets

An analysis oriented only to security zones is too coarse for achieving a secure system. A further refinement of the representation from ⟶ *Fig. 11* is therefore the identification of individual assets.

☑ After this step, all **components with a need for protection ("assets")** in the machine tool itself and in the immediate environment are **defined and marked**.

Assets can be derived from the design overview shown *(⟶ Fig. 12)* by labelling and tabulating each component, connection and data flow. Each asset is then given a unique number so that it can be identified later in the process. When assigning the numbers, **prefixes** are useful to distinguish **assets** A from **data flows** D.

The assigned numbers are given in the diagram, the result for the present example WZM is shown ⟶ *Fig. 12*.

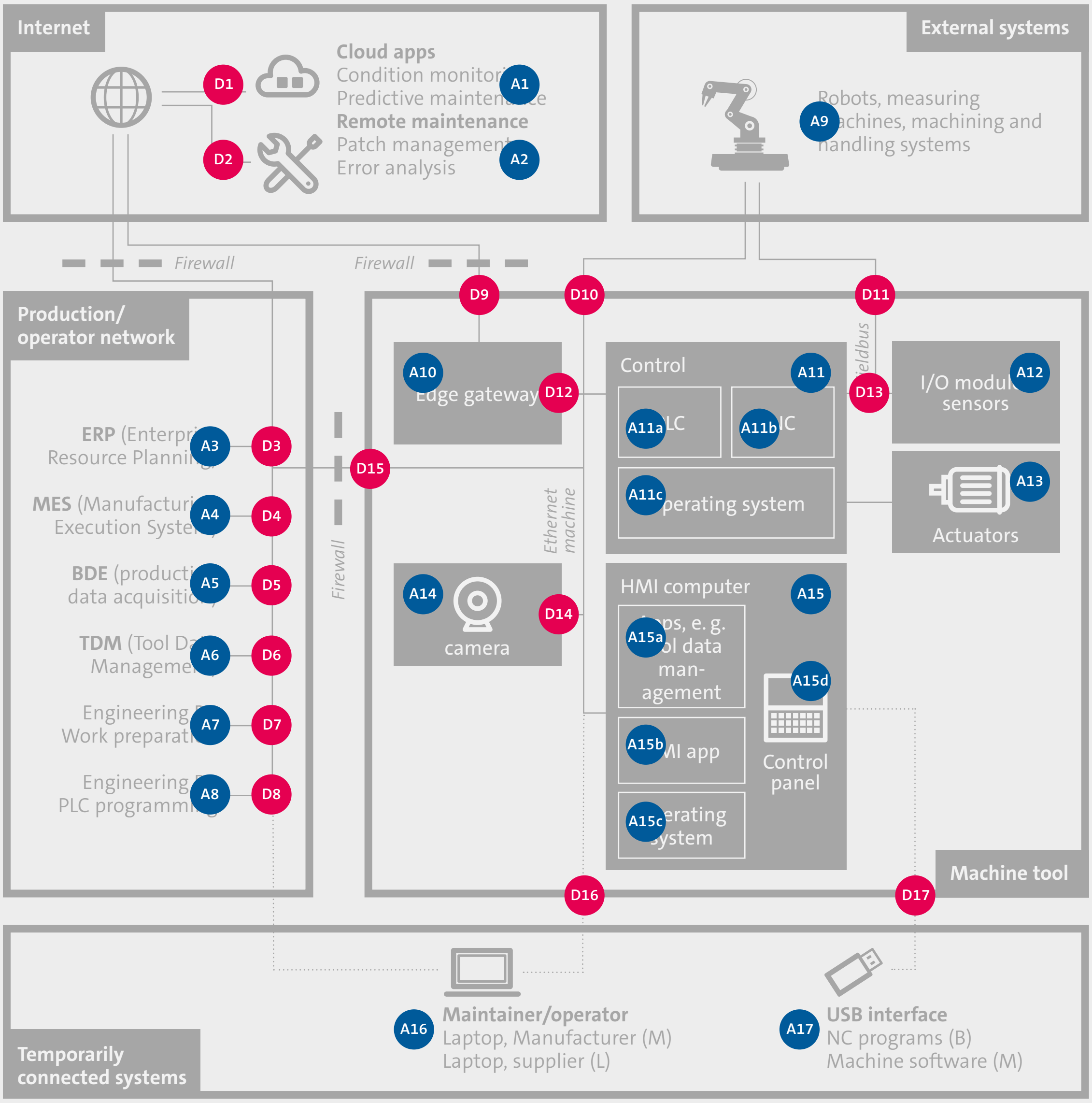


Fig. 12:
Assets and data flows with unique numbering

3.4.2 Determining the need for protection

The assets defined in this way may have different protection needs, which must be assessed individually. For this purpose, a risk analysis is carried out for all assets. The procedure is based on the ISO/IEC 27005 standard and defines the protection requirements of the assets in the first step. The assessment of the protection needs of each asset is based on the three protection goals defined in ⟶ *Table 2*:

Protection target	contextual significance
Confidentiality (C)	Confidential data may only be accessible to a defined group of recipients (persons or machines)
Integrity (I)	It must not be possible to change data without tracking or logging.
Availability (A)	Describes the time horizon within which the data must be accessed (short-term, medium-term, long-term, …)

Table 2: **Protection goals according to ISO/IEC 27005**

When evaluating (or assessing) the protection goal, for example, a graduation into three protection classes (low, medium, high) can be made which describes the respective requirement level for the protection need. A general classification criterion for the use of protection classes is the (potential) damage (e. g. financial or economic) or the expected hazard in the event of a possible violation of one or more protection goals.

The reasons for the classification should be noted in a commentary for future reference.

After this step, **protection requirements** are defined in three categories (Confidentiality, Integrity, Availability) for all assets. ✓

The following table shows examples of some of the defined assets and their assumed protection needs:

Asset-ID	Asset	C	I	A	Comment / justification
A3	**ERP plugin**	High	High	High	Necessary to control production.
A16	**Laptop manufacturer**	High	High	High	Confidential, as manufacturer passwords may be stored. Depending on the urgency of the service call, high availability is necessary.
A1	**Cloud Apps (Condition Monitoring)**	Medium	High	Low	Only low availability requirements, as production data is buffered in the edge gateway. Different assessment if Adaptive Control is used with cloud apps.
A10	**Edge gateway**	Medium	High	High	Assumption: Relevant to production, therefore high availability.
A14	**Camera**	Medium	Low	Medium	For documentation of the production process. Assumption: Failure does not lead to immediate loss of production.
A17	**USB interface**	Medium	High	Medium	Medium confidentiality as several people have access, Medium availability as access is irregular.
A17	**Interface backup machine software**	Medium	High	Medium	Backups can also be made one day later.

Table 3: **Sample definitions of protection needs for assets**

All three protection goals (Confidentiality, Integrity, Availability) are considered, and the comments contain information on the justification of individual properties.

5

3.5 Threat analysis and product risk

In the fifth step of the method, the threat and risk to the assets are assessed.

3.5.1 General description of the threat analysis

Threat analysis is a tool to obtain systematic overview of all threats (according to the definition from ⟶ *section 2.1.2*) that affect a system. There are various procedures for carrying this out. A suitable methodology should generally fulfil the following criteria:

- **Completeness**: Certainty that all threats will be found.
- **Repeatability**: A threat analysis carried out later must come to the same conclusion. The same applies if the second analysis is carried out by a different team.
- **Cost-effectiveness**: In practice, it is not possible to invest unknown levels of effort into a threat analysis. The methodological approach must therefore also define a framework to achieve the goals of completeness and repeatability within reasonable time and cost limits.

One method that fulfils the above criteria is the **"S.T.R.I.D.E." methodology** developed by Microsoft, which is described in the following ⟶ *chapter 3.5.2*.

✓ In the first part of the fifth step, the previously defined assets are tested for their vulnerability with respect to a wide range of criteria.

3.5.2 The S.T.R.I.D.E. methodology

The name "S.T.R.I.D.E." represents an abbreviation of the threat categories considered in it. The methodology is based on the fact that each threat can be assigned to one of the following categories:

- **Spoofing**
  Refers to the unauthorised use of the identity of users or processes. The simplest example of spoofing is the unauthorised use of other users' credentials (e. g. username and password).
- **Tampering**
  Refers to unauthorised alteration, manipulation or deletion of resources and data. This includes, for example, the manipulation of transmitted data via the internet or unauthorised alteration or deletion of stored data, e. g. in a database.
- **Repudiation**
  Threat due to lack of verifiability of actions or manipulations. Actions or manipulations cannot be clearly attributed to any user, or all users can deny corresponding access. An example would be if several users have access to a common account (e. g. "admin"), so that an individual action cannot be assigned to an individual user.
- **Information Disclosure**
  Disclosing confidential information: Stealing or retrieving information that is considered confidential.
- **Denial of Service (DoS)**
  The availability or reliability of an application is affected permanently or temporarily in such a way to prevent further regular use. Examples include the inaccessibility of web servers or databases, often caused by too many simultaneous access attempts and the resulting overloading of the computers.
- **Elevation of Privilege**
  A non-privileged user with restricted rights obtains higher access rights and can thus compromise further system components, possibly even gaining administrative access rights.

A S.T.R.I.D.E. analysis is now carried out in such a way that various **attack scenarios from all categories are systematically applied to the previously defined assets**, where this is possible. In other words, a full-factorial application of the attack methods to the assets is carried out, see also the diagram shown in ⟶ *Fig. 13* for explanation.

Attack variants	Assets				
	Asset 1	Asset 2	Asset 3	...	Asset n
Spoofing – Variant 1	✓	⚡	✓	✓	...
Spoofing – Variant 2	✓	✓	Not applicable	Not applicable	...
...	Not applicable	✓	✓	✓	...
Spoofing – Variant n	✓	✓	⚡	✓	...
Tampering – Variant 1	✓	Not applicable	✓	✓	...
Tampering – Variant 2	✓	✓	✓	✓	...
...	...	...	...	...	...

✓ Protected ⚡ vulnerable

Fig. 13: **Schematic representation of the S.T.R.I.D.E. methodology**

Almost every conceivable attack can be assigned to one of the categories mentioned. With a corresponding number of attack variants, a very complete security analysis can be carried out. The aim of the S.T.R.I.D.E. analysis is to obtain as complete an overview of the threat situation as possible..

As a result of a S.T.R.I.D.E. analysis, the following possible conclusions, can be drawn for the sample machine tool shown (only a few selected examples):

Asset	Category	Analysis result
A10, A11, A15 (Edge gateway, control, HMI computer)	Information disclosure	A non-customer-specific master password is used for a product series. These master passwords are published e. g. via websites like Shodan. Anyone who comes into possession of these master passwords and is on site can achieve access to the machine.
A16 (Laptop of the supplier)	Spoofing Tampering Repudiation	Service technicians often manage the access data in an unsecured file (e. g. Excel). If the technician's laptop is stolen or the access data is obtained, access to all machine data may even be possible via remote maintenance access.
A11b (NC control)	Tampering Repudiation	Due to weak or non-existent authorisation, there is a possibility of undetected (not immediately noticeable) manipulation of machine data and tool parameters (e. g. cutting speed, tolerance shifts). This can lead to workpiece scrap or even damage to the machine. Due to a lack of traceability, this can also lead to problems with the warranty.
A11c, A15c (operating system)	Denial of Service	Attacks on newly discovered, unpatched security holes in the operating system; e. g. encryption Trojans can lead to permanent impairment of the machine's availability.

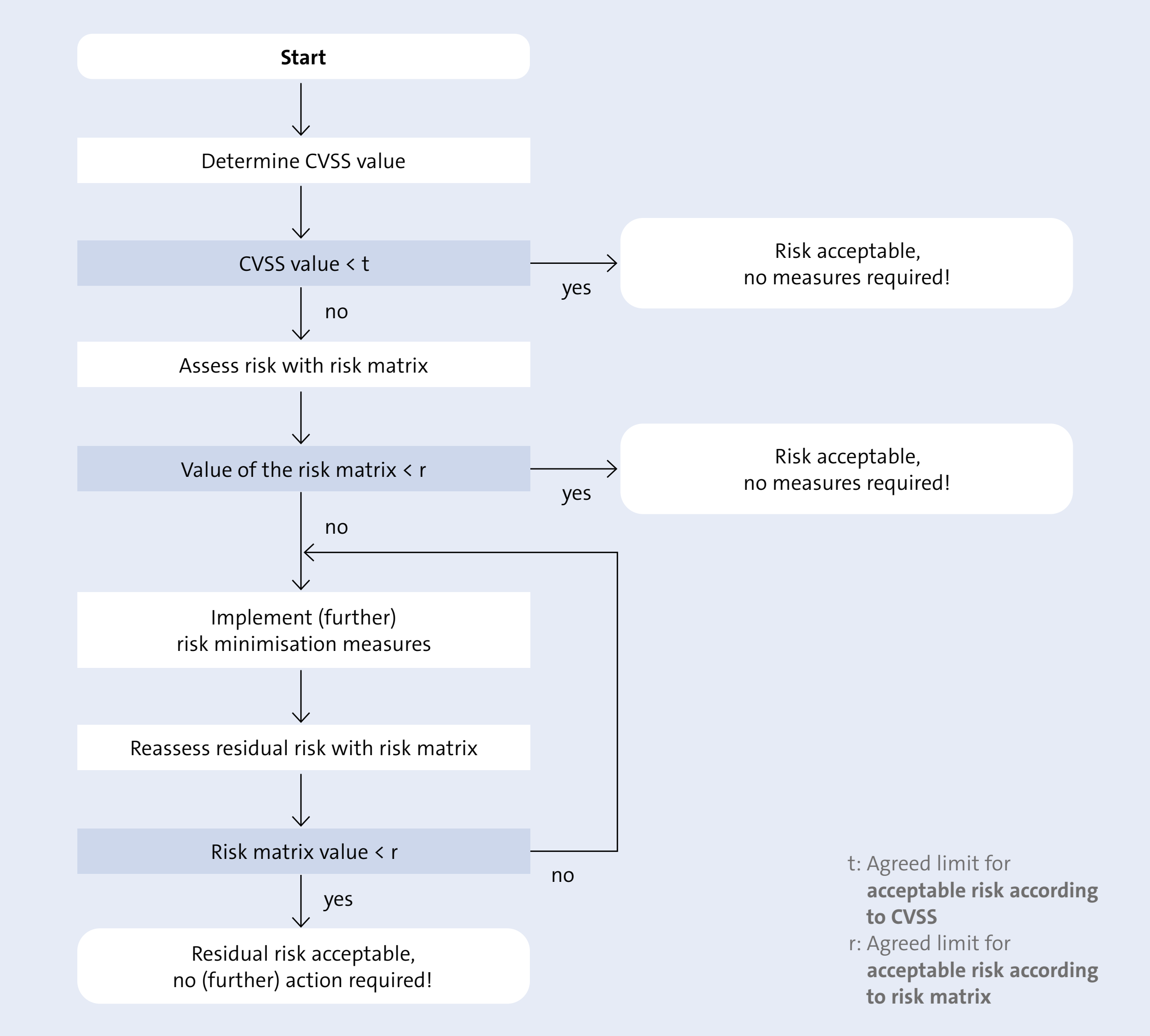
3.5.3 Threat assessment

A subsequent evaluation of these threats (e. g. according to a recognised scoring procedure and based on the risks) then allows truly relevant threats to be identified.

Two methods are presented here for the methodology:
- Assessment according to the Common Vulnerability Scoring System (CVSS) *(⟶ Section 3.5.4)* as well as
- Application of a risk matrix *(⟶ section 3.5.5)*.

Both methods are equally effective. Although they differ in their approach, they can be combined in a favourable way. A recommended way to combine both methods is shown in ⟶ *Fig. 14*.

Fig. 14:
Flow chart for the assessment of threats


Start
Determine CVSS value
CVSS value < t
yes
Risk acceptable, no measures required!
no
Assess risk with risk matrix
Value of the risk matrix < r
yes
Risk acceptable, no measures required!
no
Implement (further) risk minimisation measures
Reassess residual risk with risk matrix
Risk matrix value < r
no
yes
Residual risk acceptable, no (further) action required!
t: Agreed limit for acceptable risk according to CVSS
r: Agreed limit for acceptable risk according to risk matrix

The process described combines the CVSS method with the risk matrix in such a way that the risks are first pre-assessed using CVSS. One advantage of CVSS is that the criteria are largely objectified. In the next step, an assessment is made using a risk matrix for control purposes, meaning that an assessment has now been carried out and documented using two established assessment methods. After the defined measures to reduce the risks, a new assessment is carried out (if necessary iteratively) using the risk matrix.

The above-mentioned procedure does not represent an obligation, merely a recommendation – it is not necessary to use both procedures. Basically, a corresponding evaluation must be easily comprehensible.

From a timing perspective, it is strongly recommended that the threat analysis and risk assessment be carried out at the beginning of product development. Only then is there a chance to consider threats in the design. Often, small changes to the architecture, which are also still relatively easy to make at this early planning stage, are sufficient to create “security by design” and a tiered security architecture. The threat analysis must be updated when changes are made to the architecture. Dealing with threats that are only detected later, e. g. during implementation, is often much more time-consuming. This is because additional functionality may need to be integrated to compensate for security vulnerabilities in the architecture. In some cases, however, no compensatory measures are possible.

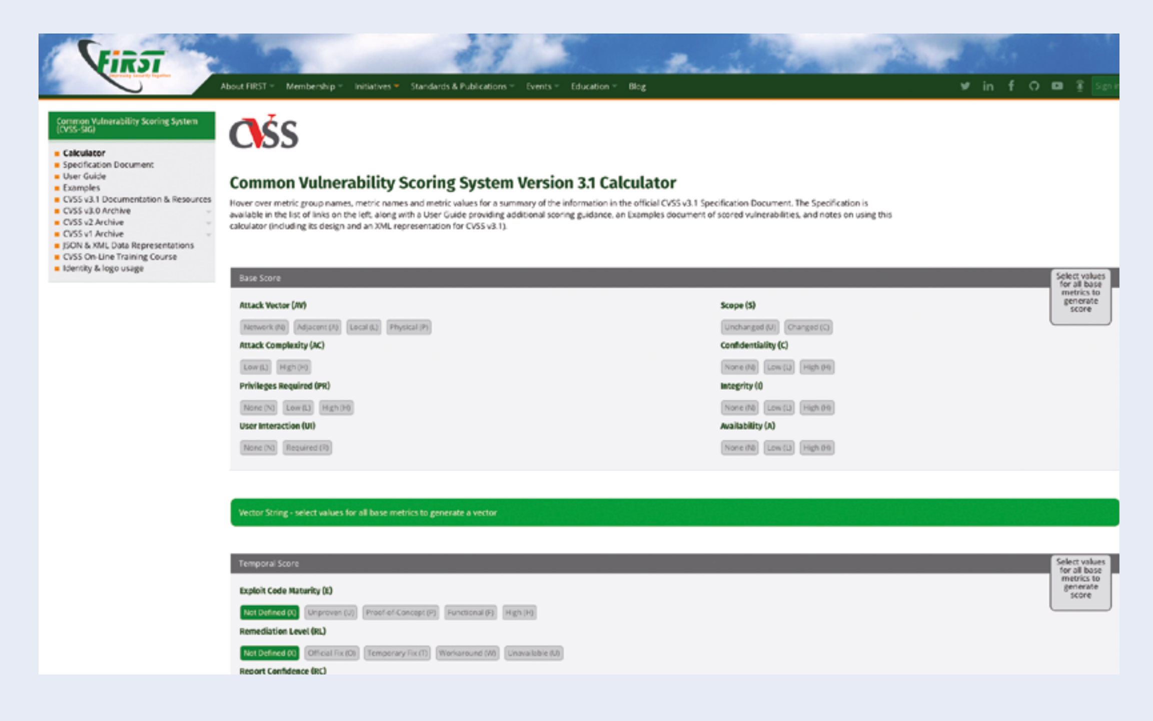
If changes to the architecture are still necessary during implementation, the threat analysis must be updated. Some threats can only be dealt with through compensatory measures by the operator. In some cases, there may be no suitable measures or they are so costly that the most efficient treatment is to bear the risk. Both the measures for the operator and the unavoidable risks must be documented and agreed with the integrators and operators.

In the second part of the fifth step, the previously defined assets were subjected to a risk assessment using of the CVSS and/or risk matrix. ✓

3.5.4 The CVSS rating system

CVSS stands for "Common Vulnerability Scoring System" and refers to an industry standard for assessing the severity of potential or actual security vulnerabilities in IT systems.

A detailed introduction to CVSS is provided in the ⟶ *article [4]*. Generally speaking, CVSS provides a methodology that can be used to quantify the danger posed by a security vulnerability. For practical implementation, there are corresponding online tools (calculators), see ⟶ *[5]*.



Common Vulnerability Scoring System Version 3.1 Calculator

Hover over metric group names, metric names and metric values for a summary of the information in the official CVSS v3.1 Specification Document. The Specification is available in the list of links on the left, along with a User Guide providing additional scoring guidance, an Examples document of scored vulnerabilities, and notes on using this calculator (including its design and an XML representation for CVSS v3.1).

Base Score

Select values for all base metrics to generate score

Attack Vector (AV): Network (N) Adjacent (A) Local (L) Physical (P)

Attack Complexity (AC): Low (L) High (H)

Privileges Required (PR): None (N) Low (L) High (H)

User Interaction (UI): None (N) Required (R)

Scope (S): Unchanged (U) Changed (C)

Confidentiality (C): None (N) Low (L) High (H)

Integrity (I): None (N) Low (L) High (H)

Availability (A): None (N) Low (L) High (H)

Vector String - select values for all base metrics to generate a vector

Temporal Score

Select values for all base metrics to generate score

Exploit Code Maturity (E): [✓ Not Defined (X)] Unproven (U) Proof-of-Concept (P) Functional (F) High (H)

Remediation Level (RL): [✓ Not Defined (X)] Official Fix (O) Temporary Fix (T) Workaround (W) Unavailable (U)

Report Confidence (RC)

Fig. 15: **Online calculator for a CVSS score,** *Source:* ⟶ *[5]*

The assessment is based on several criteria, all of which then flow into a CVSS score. The criteria include *(see also ⟶ Fig. 15)*:

- The attack vector (AV) – e. g. via the network or physically (=on-site)
- The complexity of the attack (AC)
- Whether special rights are required for the attack (PR)
- Whether user interaction (UI) is required, e. g. executing a mail attachment.
- The target object (Scope, S): A distinction is made here as to whether the attack target is also the gateway (“unchanged”) or whether the attack moves on from there (“changed”). Such a “changed” scenario would be, for example, an attack on computer A in order to reach computer B via the network.
- The requirements already defined in ⟶ *chapter 3.4.2*:
  - Confidentiality (C)
  - Integrity (I)
  - Availability (A)

The CVSS score then results from the information on corresponding assessments. In this way, potential security vulnerabilities can be evaluated according to various criteria, so-called metrics, and compared with each other so that a priority list for countermeasures can be created.

3.5.5 Risk matrix according to IEC 62443 3-2

In addition to an assessment according to CVSS, a risk matrix is used here. The risks are quantified via the relationship between probability of occurrence, possible damage and risk – depending on the probability of occurrence and damage, a point value is produced which represents the extent of the risk. ⟶ *Fig. 16* shows an example of a risk matrix, as illustrated in the IEC 62443-3-2 standard. In practice, the organisation can replace the example with its own risk matrix.

Fig. 16:
Risk matrix according to IEC 62443-3-2

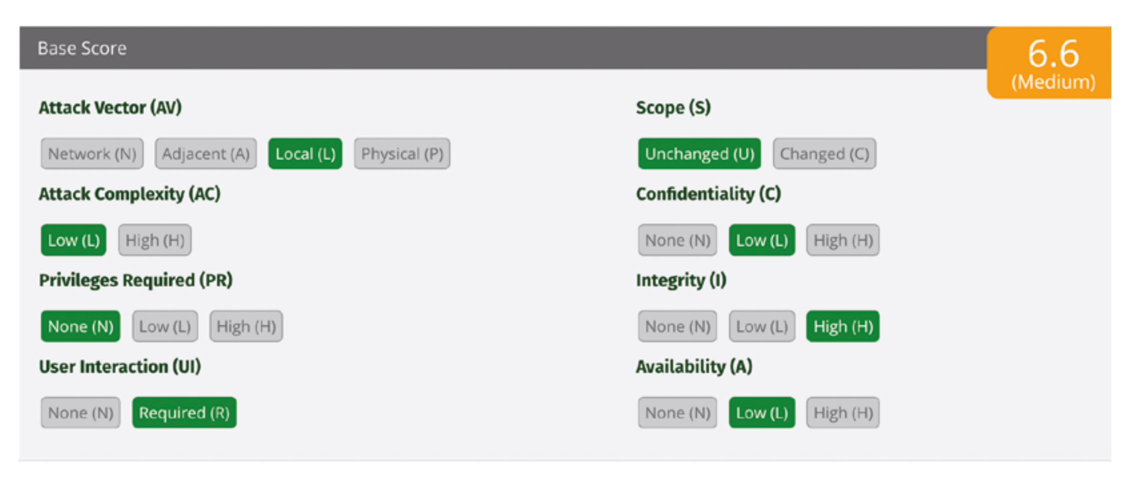
Damage		**Probability of occurrence**				
		Negligible 1	Unlikely 2	Possible 3	Probable 4	Definite 5
Very low	1	**1**	**2**	**3**	**4**	**5**
Low	2	**2**	**4**	**6**	**8**	**10**
Medium	3	**3**	**6**	**9**	**12**	**15**
High	4	**4**	**8**	**12**	**16**	**20**
Critical	5	**5**	**10**	**15**	**20**	**25**

3.5.6 Exemplary threat analysis

The following is a summarised example of a complete threat analysis.

The threats are assessed according to severity and risk. For this purpose, the CVSS vulnerability rating system *(⟶ section 3.5.4)* and the risk matrix *(⟶ section 3.5.5)* are applied.

An exemplary analysis of the USB interface using CVSS is shown in ⟶ *Fig. 17*, and an application of the risk matrix applied to it in ⟶ *Fig. 18*.



Base Score 6.6 (Medium)

Attack Vector (AV)
Network (N) Adjacent (A) [✓ Local (L)] Physical (P)

Attack Complexity (AC)
[✓ Low (L)] High (H)

Privileges Required (PR)
[✓ None (N)] Low (L) High (H)

User Interaction (UI)
None (N) [✓ Required (R)]

Scope (S)
[✓ Unchanged (U)] Changed (C)

Confidentiality (C)
None (N) [✓ Low (L)] High (H)

Integrity (I)
None (N) Low (L) [✓ High (H)]

Availability (A)
None (N) [✓ Low (L)] High (H)

Fig. 17: **Online tool for evaluation by means of CVSS using the USB interface as an example**

Damage		**Probability of occurrence**				
		Negligible 1	Unlikely 2	Possible 3	Probable 4	Definite 5
Very low	1	**1**	**2**	**3**	**4**	**5**
Low	2	**2**	**4**	**6**	**8**	**10**
Medium	3	**3**	**6**	**9**	**12**	**15**
High	4	**4**	**8**	**12**	**16**	**20**
Critical	5	**5**	**10**	**15**	**20**	**25**

Fig. 18: **Risk assessment of an attack via the USB interface**

Asset	**Category**	**Description**
A17 (USB interface)	Information Disclosure	Via the USB interface, the entire production data and NC programs can be extracted and made accessible to unauthorised third parties. This can lead to loss of know-how and infringement of the operator's intellectual property.

Threat assessment			
CVSS Score	**Risk probability of occurrence**	**Damage**	**Risk value according to risk matrix**
6.6	Probable	High	16 → Risk not acceptable, measures as described in the following chapter must be taken

Table 4: **Summary of the evaluation of the USB interface**

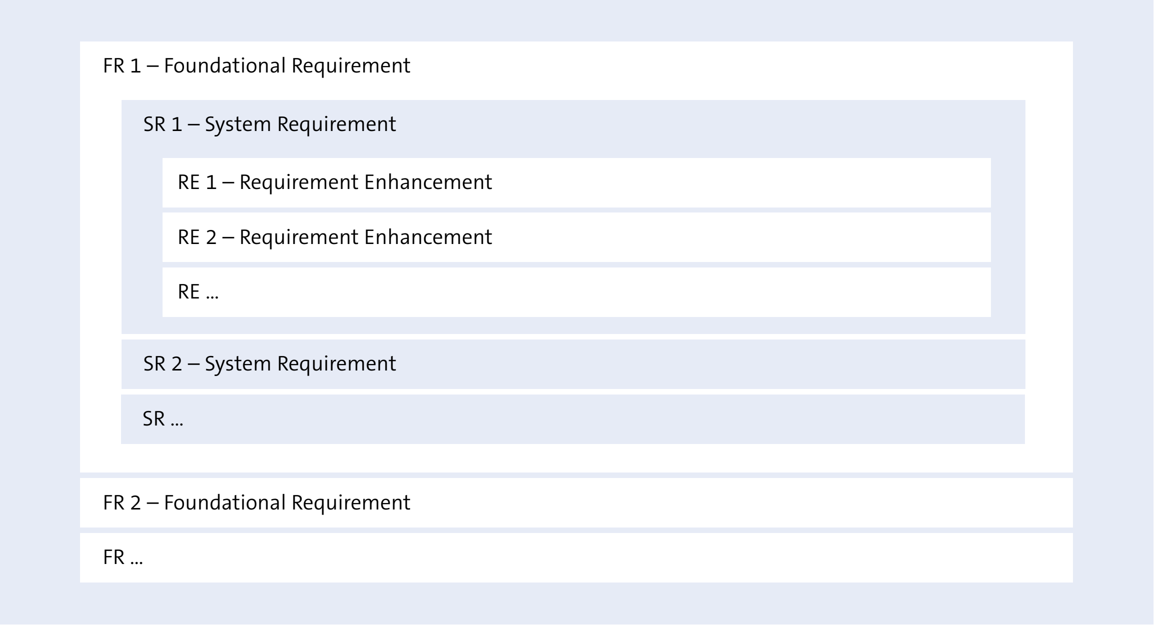
6

3.6 Security requirements and measures

In the sixth step of the method, the security requirements are to be backed up with corresponding measures.

For the different security levels (SL-C 1 to 4), **capability requirements** for systems (3-3) and components (4-2) are defined in parts 3-3 and 4-2 of the standard. In the standards, these requirements are hierarchically structured; a distinction is made between foundational requirements (FR), system requirements (SR) and further requirements contained therein (requirement enhancement, RE); see also ⟶ *Fig. 19*.

Fig. 19:
Hierarchy of requirements in IEC 62443-3-3


FR 1 – Foundational Requirement
SR 1 – System Requirement
RE 1 – Requirement Enhancement
RE 2 – Requirement Enhancement
RE ...
SR 2 – System Requirement
SR ...
FR 2 – Foundational Requirement
FR ...

The respective requirements differ according to the targeted security level. This was defined in the second step *(⟶ Chapter 3.2)*, either for the entire system or on an asset basis.

After the sixth step, suitable measures for reducing the risk are listed for the assets under consideration at the respective required security level.

Although the standard does not specify any concrete measures in detail, some threat mitigation measures can be derived from the given requirements.

As an example, the requirement defined in IEC 62443-3-3, chapter 5.3.1. is cited here. Classified in the basic requirement FR1 (“Identification and authentication”), subcategory SR 1 (“Identification and authentication of human users”) is required here:
“The automation system shall provide the capability to identify and authenticate all human users. This capability shall enforce identification and authentication at all interfaces that grant human users access to the automation system, so as to support the separation of duties and the principle of minimum necessary rights in accordance with the applicable IT security policies and procedures.”

A justification for this is given in the following section 5.3.2:
“All human users must be identified and authenticated for all types of access to the automation system. The identity of these users should be authenticated through the use of such methods as passwords, tokens, biometrics or, in the case of multifactor authentication, a combination of these methods [...]”.

It can be seen that some measures can be derived directly from these explanations (passwords, tokens, biometric features, ...). **This step must always be taken with regard to one's own particular system** – industrial automation systems and machines are so different that no generally valid advice can be given here. However, the standard offers a variety of suitable suggestions.

With regard to the USB interface mentioned in the previous example, the following measures could be suggested:

- Use of the USB interface may only be possible after user login with password. (Unique authentication)
- Event logging when exporting data to track the use of the USB interface.
- Depending on criticality: enforce encryption on external media.

3.7 Residual risk

7

The seventh step of the method is to assess the residual risk after implementation of the defined measures.

The implementation of measures changes the probability of occurrence and/or the damage potential of threats. The aim of a new risk assessment is to determine the residual risk (net risk) **after the implementation** of the proposed measures.

After the seventh step, the residual risks of the assets assessed after implementation of remediation measures are quantified.

As can be seen from the chosen example “USB interface”, residual risks always remain that cannot be completely eliminated by the manufacturer. In order to further reduce these risks, further compensatory measures must be implemented by the operator, for example, or the risk must be accepted. Transparent communication of the results to the operator is therefore imperative in order to achieve this acceptance (or also corresponding further measures).

Table 5: **Summary of the evaluation of the USB interface after taking measures**

Asset	Category	Description
A17 (USB interface)	Information disclosure	Via the USB interface, the entire production data and NC programs can be extracted and made accessible to unauthorised third parties. This can lead to loss of know-how and infringement of the operator's intellectual property.

Threat assessment			
CVSS Score	**Risk probability of occurrence**	**Damage**	**Risk value according to risk matrix**
6.6	Probable	High	16

Measures
• Use of the USB interface may only be possible after user login with password (unique authentication). • Event logging when exporting data to track the use of the USB interface. • Depending on criticality: enforce encryption on external media.

Renewed assessment with risk matrix after implementation of the measures			
	Risk probability of occurrence	**Damage**	**Risk value according to risk matrix**
	Unlikely	High	8

8

3.8 Documentation

All steps carried out should be documented in a comprehensible way. Appropriate documentation backs up the arguments presented to the customer and serves as proof of a secure development methodology.

The tables of the different assets with their protection needs/measures, etc are an essential part of the documentation. They can be found in the following pages.

In principle, the documentation should fulfil the requirement of rendering efforts to safeguard the development process comprehensible and transparent, e. g. by also listing the reasons for actions.

☑ After the eighth step, steps 1 to 7 are summarised and documented, ideally also cross-checked by a person not involved in the analysis.

4. Considerations over the entire useful life

4.1 Basic information

In the explanations on the protection strategy from ⟶ *Chapter 2.3*, it became clear that a development process geared towards high OT security alone cannot achieve a sufficient level of protection, especially not in the long term. Additional accompanying measures are therefore required for the operational phase, which often lasts for decades. These include actions on the part of the customer, but can also be prepared and supported by the manufacturer of the machine tool. Proposals for the user can already be found in ⟶ *[1]*, the present document is therefore primarily dedicated to the possibilities and obligations of the machine tool manufacturer.

The reasons for the necessity of protective measures during operation are complex; in particular, four causes are highlighted here:

1. Different product life cycles
Compared to machine tools, IT systems and the associated environments and security technologies have much shorter product life cycles. It is therefore highly probable that significant findings on security vulnerabilities (which may have existed for a longer period of time) and also technological developments will arise during the operating time of the machine, which need to be responded to.

2. Wide range of variants
A machine tool is a complex system, often tailor-made for the customer, which is made up of a large number of supplier components and different units. Every significant change to one of these components, e. g. through software patches, requires renewed safeguarding of the function of the entire system, which is not easily achievable by the manufacturer.

3. Architecture of IT components
In general, digital products nowadays no longer consist only of self-developed components. Instead, a large number of third-party components in the form of libraries, frameworks and operating systems are delivered with the products. These components in turn integrate further components. Each of these dependencies, even transitively, can contain security vulnerabilities that can have an impact on the security of the products (e. g. “Log4j”).

4. Operational and functional reasons
Certain measures that appear helpful or even necessary from an isolated IT security perspective can only be implemented in an inconvenient way or simply not at all for machine tools in everyday operations. One example would be the data protection requirements (which are also relevant under labour law), which stand in the way of storing personal access data – but which would be desirable for purely security reasons. Other ways must therefore be found to secure the corresponding systems.

4.2 Measures and approach

It is also the responsibility of the machine tool manufacturer to ensure that a high level of security is maintained even after delivery. Therefore, appropriate processes must be established at least until the end of the guaranteed support period (end of service), which enable the handling of security vulnerabilities and the provision of security updates.

4.2.1 Vulnerability management

Security vulnerability management means operating a regulated process for dealing with possible security vulnerabilities in products. From the perspective of a mechanical engineering company, it therefore essentially consists of identifying security vulnerabilities in the embedded components, documenting them and dealing with their effects on the product.

To do this, it is necessary to create a directory of all embedded libraries, components, operating systems, etc. and their version statuses and keep it up to date (software inventory, component register). For each of these components, a strategy must then be defined on how to become aware of security vulnerabilities.

- In the case of **software developed in-house**, this task can be covered by the company's own developers – at least on the upper level, the software components used are known to them or have been selected by them. However, it is also important to take a look at other subordinate dependencies – these should also be examined and documented and can be critical in deciding for or against the use of a certain component.

- Appropriate regulations must be found with the providers of **supplied systems**. They should be required (if necessary through corresponding contracts) to name subcomponents and to inform the machine builder about security vulnerabilities in their own or connected software.

It is recommended to establish a PDCA (Plan-Do-Check-Act) process with progressively iterated risk assessment at fixed intervals (e. g. updated assessment once a year). In addition, as a result of events such as the disclosure of major security vulnerabilities or major changes in the software components used, e. g. operating system updates (such as Windows 10 instead of Windows 7, Linux instead of Windows, ...), a corresponding reassessment should take place.

4.2.2 Dealing with security updates

If security vulnerabilities affecting the software used are identified, the machine manufacturer must be in a position to announce and provide suitable security updates in good time. In the interest of the machine operator, however, it must be ensured that patches do not impair the proper operation of the product. This can often only be ensured through prior tests, such as regression tests. Furthermore, due to the complex structure of the coordinated components of a machine tool, it is possible and also probable that the operator cannot take care of a corresponding upgrade independently, but only with the help of the machine tool manufacturer or possibly even only with the support of the control manufacturer.

Operators must also be provided with appropriate documentation as part of security updates. This must contain the following information:

- The affected software and the version to which the update applies.
- Installation instructions and a description of the effects that the installation has, e. g. changed behaviour or the need to restart.
- Testing capabilities that allow the operator to ensure the integrity of the update and verify successful installation.
- A description of possible risks that arise if the update is not installed.

It is often necessary to provide patches for dependent components or operating systems or to recommend their installation. In this case, the following information must be made available to the operator:

- Indication of whether the software is compatible with the security update for the dependent component or operating system.
- Recommended mitigation measures in case the update is not approved for use by the manufacturer.

If an upgrade based on the above criteria is not possible, extremely costly or not desired for reasons of maximum availability, the methods recommended in the following ⟶ *Chapter 5* for the “brownfield” can help to achieve a higher security standard.

4.3 Support from system suppliers

4.3.1 Role of the system suppliers

From the perspective of OT security, the manufacturer of the control system is the most important partner for the machine tool builder, even after delivery of the machine. It should be noted that the manufacturers and suppliers of control systems for machine tools also find themselves in the role of a system integrator in addition to their role as software and hardware manufacturer: In modern control systems, “supplied” software components are often used, for which corresponding monitoring as described in ⟶ *Chapter 4.2* must then be implemented on the part of the control manufacturers.

From the point of view of a machine tool manufacturer, it is particularly important that the system supplier is integrated in the manufacturer's own security vulnerability management and provides corresponding information. In addition, operating system updates, for example, should be approved, so that these are already tested by the control system supplier for compatibility with their own system, as this is not generally feasible for a machine tool manufacturer.

4.3.2 Support services

Many system suppliers provide information and various publications that offer further assistance with the specific control system. At this point, we refer to the respective websites of the manufacturers.

5. Treatment of the brownfield

5.1 Initial situation

Machine tools are durable capital goods that are often operated for decades – a large proportion of the systems in a typical production facility are therefore inevitably older systems. From the point of view of OT/IT security, this also includes machines that are only a few years (or even months!) old which may no longer be up to date, e. g. in terms of the operating system, at the time of initial commissioning at the customer's premises, as the delivery and transport times alone can be longer than the time interval between security patches.

Even for series that are part of the current product range, but whose design and development process dates back a long time and which therefore can or should no longer be safeguarded in the way described, safeguarding methods beyond the design process must be found.

It follows that:

1. A solution must be found for existing plants to operate them in such a way that availability can be ensured and yet the benefits of digitalisation in production can be implemented while maintaining high security standards.
2. Such a solution can also further safeguard new machines as a supplementary component.

A promising approach is to encapsulate the systems as described below.

5.2 Delimitation and encapsulation of systems

The concept of zoning presented as part of the secure development process can also be seen as an integral part of the accompanying uptime measures. If an edge gateway, as shown in ⟶ *Fig. 11*, is used to decouple the machine tool from essential parts of your OT environment, this narrows the potential for attack considerably. Everything "behind" the gateway can then be considered a black box.

In practice, the interface to the outside world, i.e. the gateway, is of high security relevance in such an architecture: critical security vulnerabilities must be detected and remedied immediately (i.e. within hours or days, not weeks or months). This task can be undertaken by service providers who are specialised in the management of the devices they sell. The underlying – possibly very complex – control system is then no longer exposed to threats via the network interface in the same way.

This structure thus offers a solution to the problem that the high complexity of the hardware and software of a typical control environment for machine tools makes the application of security patches difficult or impossible in practice. The gateway, on the other hand, is independent of the machine and can therefore be kept up to date relatively easily. Corresponding gateways (with automatic update mechanisms) are available from the control manufacturers and other suppliers.

Such a structure only protects the network interface and not other entry paths (removable media, otherwise connected equipment, ...) and can therefore only represent *one*, albeit important, component of a security shield.

Example: A 5-year-old NC-controlled machine tool, integrated into the IT environment of a smaller manufacturing company, is to be secured.

Current machine tool situation:

- HMI based on Microsoft Windows 7.
  - Windows patch level with delivery status, i.e. 5 years old.
- HMI runs on an industrial PC with two network ports.
  - The first port of the industrial PC is connected to a network within the machine tool which is used for communication between the control components.
  - The second port of the industrial PC is connected to the operator's company network for the exchange of NC programs (Windows file sharing/SMB protocol).
  - In the HMI, the two ports are logically connected via a corresponding configuration of the controller manufacturer separately.
- Other machine components (e. g. sensor gateways, process monitoring) are connected to the machine's internal network and communicate with the control components.
- NC control and PLC run on a proprietary operating system (Linux derivative), the machine tool manufacturer has no access to these operating systems.
- A virus scanner has not been approved by the control manufacturer; this could lead to a slowing down of the user interface or other indefinite malfunctions.
- Remote service runs via client software installed locally on the HMI.

Current company network situation:

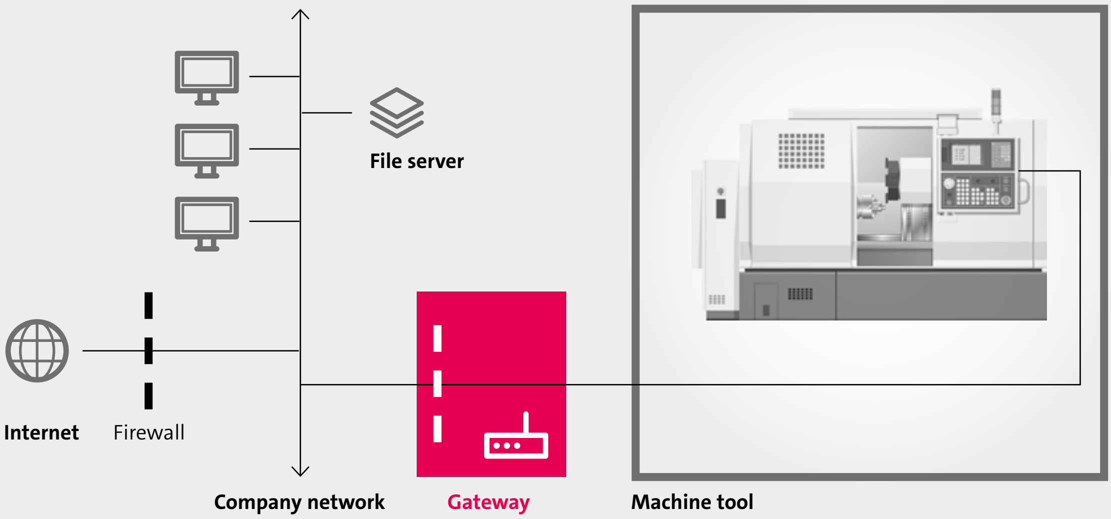
- The company network is decoupled from the Internet by a router with an integrated firewall from the consumer sector.
- Connection of the machine to the Internet indirectly via the company firewall in the router.
- No LAN segmentation within the company network, the production machines are all in the same network segment as other IT devices, such as the company's file server.

Threats:

- From today's IT perspective, Microsoft Windows 7 is outdated and can no longer be brought up to a contemporary level of IT security:
  - The system has been discontinued by Microsoft.
  - In the example, it has a very old patch level and is therefore particularly vulnerable to many security gaps that have arisen in recent years.
  - Not even all currently available security patches for the operating system can be applied, neither by the operator nor by the machine tool manufacturer, as the patches are not fully tested for compatibility with the control manufacturer's HMI software as well as any HMI software developed in-house.
  - Updating the operating system to the current Windows version is also not possible due to hardware limitations and compatibility problems of the software used.
- If the industrial PC is compromised, the NC/PLC itself could become the target of an attack via this path (or directly via a connected service computer). The software on the NC/PLC cannot be readily updated, as considerable effort and possibly incompatibilities are to be expected here.
- Other components in the machine network, such as process monitoring or similar, can be compromised in the same way.

- The industrial PC is directly connected to the company network and is only separated from the machine network zone by internal Windows mechanisms. These mechanisms are to be considered insecure as described.
  - An attack starting from the company network would most likely successfully compromise the machine. Such an attack could, for example, be carried out by other devices that may also be present in the company network for a short period of time, such as service laptops of suppliers or even devices infected by Trojans within the company network.
  - Conversely, if the machine were compromised, by whatever means, an attack could be launched against all accessible targets in the company network.
  - The machine can serve as a "retreat" for malware, which then repeatedly attacks the company network. Such a scenario could probably only be solved by completely restoring or reinstalling the industrial PC.

Fig. 20: **Structural diagram of the example scenario (with gateway)**



Measure: Gateway

- By encapsulating the industrial PC from the company network via a gateway with firewall function, the machine tool "behind" the gateway can be regarded as a zone (black box).
  - A corresponding gateway can also be retrofitted in the field at any time.
  - The system security with regard to the LAN interface (!) is thus guaranteed by the gateway and is thus effectively independent of the patch level and installed software on the machine tool.
  - The gateway is crucial for system security; it should therefore be kept permanently up-to-date (if necessary by a specialised provider).
  - Communication within the local machine network (e. g. between process monitoring and control) is not restricted or impaired in any way, but is also not monitored.
  - All desired access to the machine from the company network and vice versa must be explicitly approved in the gateway.

    In the case presented, this concerns:
    - Windows file shares for transferring NC programs to the machine
    - Screen sharing for remote service

    The exact releases depend on the protocols used and the access direction (e. g. company network → machine or vice versa).

⚠ However, for the application described, it should be noted: The system can still be compromised by external service computers or the USB interface! Here, further protective measures may have to be implemented, possibly also organisational ones (e. g. prohibition of USB sticks on the machine).

6. Bibliography

Index	Bibliography	Description
[1]	VDW: IT-Sicherheit an Werkzeugmaschinen – Maßnahmen zur einfachen Umsetzung. Whitepaper, 2020. ⟶ *https://vdw.de/wp-content/uploads/2021/03/pub_IT-Sicherheit-an-Werkzeugmaschinen_VDW.pdf*	Collection of measures for users and operators to increase IT security in the operation of machine tools, available on request from VDW in printed version
[2]	ZVEI: Orientation guide for manufacturers for IEC 62443. Whitepaper, 2017.	Orientation guide
[3]	VDMA: Guide IEC 62443 for mechanical and plant engineering. revised edition, 2021.	Introduction to IEC 62443 with many notes on the assessment of plants, available on request from VDMA
[4]	Andreas Kurtz: From Low to Critical: Vulnerability assessment with CVSS. ⟶ *https://heise.de/-5031983* (As of 25.01.2021)	Article describing the CVSS system
[5]	Forum of Incident Response and Security Teams (FIRST): Common Vulnerability Scoring System Version 3.1 Calculator. ⟶ *https://www.first.org/cvss/calculator/3.1* (As of 18.01.2022)	Online calculator for the CVSS Scoring System

Publisher
German Machine Tool Builders' Association
Machine Tool and Manufacturing Systems Association
within the VDMA
Lyoner Strasse 18
60528 Frankfurt am Main
Phone +49 69 756081-0
Fax +49 69 756081-11
E-Mail vdw@vdw.de
Internet www.vdw.de
Twitter www.twitter.com/VDWonline
YouTube www.youtube.com/metaltradefair

Chairman
Franz-Xaver Bernhard,
Maschinenfabrik Berthold Hermle AG, Gosheim

Executive Director
Dr. Wilfried Schäfer

Authors
Working Group "Control and Systems Engineering"
of the VDW Research Institute, with expert advice from
Prof. Dr. Felix Hackelöer, Institute for Automation & Industrial IT
(AIT), TH Cologne, as well as Bernd Gehring, Peter Schmidt and
Tilo Renz, ditis Systeme, branch office of JMV SE & Co. KG, Ulm

Design and layout
Klaus Bietz \ visuelle Kommunikation, Frankfurt am Main

Printing
Druck- und Verlagshaus Zarbock GmbH & Co. KG

Date
10/2022

Acknowledgements for photographs
Adobe Stock

German Machine Tool Builders' Association (VDW)
www.vdw.de