Journal of Universal Computer Science, vol. 30, no. 5 (2024), 563-589
submitted: 15/9/2023, accepted: 14/12/2023, appeared: 28/5/2024 CC BY-ND 4.0

Towards a single device for multiple security domains

Florian Skopik
(AIT Austrian Institute of Technology, Vienna, Austria
https://orcid.org/0000-0002-1922-7892, florian.skopik@ait.ac.at)

Arndt Bonitz
(AIT Austrian Institute of Technology, Vienna, Austria, arndt.bonitz@ait.ac.at)

Daniel Slamanig
(AIT Austrian Institute of Technology, Vienna, Austria
https://orcid.org/0000-0002-4181-2561, daniel.slamanig@ait.ac.at)

Markus Kirschner
(MUSE Electronics GmbH, Vienna, Austria, markus.kirschner@backbone.io)

Wolfgang Hacker
(Ministry of Defence, Vienna, Austria, wolfgang.hacker@bmlv.gv.at)

Abstract: Military field operations place high demands on information and communication technology (ICT) devices, both in terms of reliability and security. These requirements include robustness against environmental influences such as vibrations, water, and humidity as well as protection against physical attacks and cyber-attacks. Attempts to compromise a device must be detected immediately, and if necessary, trigger automated countermeasures such as alarms, partial deactivation or emergency wiping of all data. In this work, we specifically focus on cyber security issues and aim to deliver a concept for a device that can be used in multiple security domains, isolating mission-specific data from each other without the risk of data spillover. For that purpose, we outline a high-level concept for a resilient single device concept that is able to withstand common intrusion attempts. We identify threat agents, misuse cases and the risks of a single device concept for multiple security domains and evaluate the most pressing issues. Based on the identified risks, we determine additional mitigation measures and discuss their applicability. We foresee our work to provide valuable insights into the requirements on and design decisions of highly secure mobile device solutions.

Keywords: cyber security, hardened tablet, rugged device, risk analysis
Categories: H.4.0, K.6.5, B.m
DOI: 10.3897/jucs.112790

1 Introduction

The use of information and communication technology (ICT) to support soldiers is state-of-the-art in practically all operations today. Especially when deployed in the field – in vehicles or on foot – military operations place high demands on such devices in terms of reliability and security. Such a device must not only meet the required military standards with regard to compromising electromagnetic radiation (also known as Telecommunications Electronics Material Protected from Emanating Spurious Transmissions

(TEMPEST)) [U.S. Department of Defense, 2015, North Atlantic Treaty Organization, 2015] and robustness against environmental influences such as vibrations, water and moisture [MIL-STD-810H, 2019], but also protection against physical attacks and lately also cyber attacks.

While we focus in our work on the military domain, similar, albeit reduced, requirements can also be found in non-military application areas [Agre et al., 2013, Kurt et al., 2016]. Some examples include harsh industrial environments with dust, strong vibrations, splashing water, flying sparks and, last but not least, high electromagnetic exposure, such as near welding work, induction furnaces and other applications with high switching currents and current pulses. Other scenarios include mobile devices used for outdoor installation and maintenance work, often even on public places where equipment is easily accessible by foreigners, as well as equipment used by people from emergency services under high time pressure, such as rescue services, fire brigades and police. Not only harsh environmental conditions pose specific challenges to such devices, but also their use in uncontrolled places, with potential access of illegitimate users to these devices.

Attempts of compromise and manipulation, be it physical or software, must be detectable immediately and, if necessary, trigger automated countermeasures such as alarms, (partial) deactivation or emergency deletion. This work aims to outline an integrated overall concept for highly robust end devices that combine physical, hardware and software design requirements with individual configurability for specific purposes. For armed forces and for crisis intervention teams, the situation is aggravated by the fact that the replacement of equipment that has become unsuitable in areas that are difficult to access (e.g. after avalanches or mudslides or after earthquakes) can often only be carried out with great effort and/or valuable loss of time. The requirements and operating conditions are diverse and in part unpredictable.

All this means, that the demand for suitable terminals such as tablet computers and smartphones is growing significantly among authorities, emergency services and the armed forces. Furthermore, the demand in industry and industrialized agriculture is also growing rapidly [North Atlantic Treaty Organization, 2023a]. The manufacturing industry, for example, is currently undergoing a profound transformation process in which the digitization and networking of systems is advancing rapidly in the course of Industry 4.0. In addition to robustness, data security is also a top priority here.

Although robust products are available on the market, only a few of these devices – apart from a battery status display or the availability of hardware modules such as modems or network adapters – allow reliable monitoring of the device's components. This aspect has received little attention to date, although it is essential: once the device has been in use and may have suffered damage that is not immediately apparent, this can lead to failure or force an abort during a subsequent mission due to a device failure. The soldier in the field must be able to assess the condition of his data terminal at any time, for example, in order to be able to make reliable decisions. The use of the device in different operational contexts also makes it essential to maintain strict data separation. Data integrity must be maintained from device commissioning to decommissioning. Furthermore, there is the need for simplified administration: a device with exactly one hardware configuration, whose hardware modules and interfaces can be switched on or off according to mission requirements, can greatly reduce overhead, especially for smaller armed forces or other user groups. Furthermore, it is of immense importance to detect any unauthorized intrusion attempts into the device or forms of manipulation, and – depending on a predefined action plan – immediately initiate automated actions to avert greater damage. Special hardware components can be used to automatically trigger an action immediately in response to (linked) events. For example, a dedicated

security controller may transmit an alert signal to other devices in the network, deactivate certain services or modules, or initiate an emergency wipe of data on the device. Such a function cannot be implemented "on top" of an existing device concept, but requires a well-considered overall design from the ground up.

If the same devices are to be deployed at often short intervals one after the other in different mission-specific security domains, for example in command posts and mobile data centres, the security requirements further increase significantly. Ensuring that highly sensitive and mission-critical data is only visible in the currently relevant security domain is essential. International security and abuse incidents due to inadequate solutions based only on visualization and pure container solutions underline the importance of a robust and reliable "single device solution" that can be used in multiple application domains without security concerns. For security reasons, it is currently common to purchase multiple devices in order to be able to change the device for each use, depending on the security domain for which it was ultimately certified (cf. military classification levels [North Atlantic Treaty Organization, 2023b][1]). This is not only economically inefficient, since such special devices, including expensive accessories, often incur high additional costs, but also poses a logistical challenge and causes additional work in the administration of the devices.

Hence, in this work, we specifically focus on cyber security issues and aim to deliver a concept for a device that can be used in multiple security domains, isolating mission-specific data from each other without the risk of data spillover. The main contribution of this paper is not a single new approach, algorithm or methodology, but a showcase of the integrated application of a variety of known concepts in a consistent manner to discuss emerging risks, threats and appropriate countermeasures for a secure device for multiple environments in a holistic and conclusive manner. The building blocks to achieve this objective are as follows:

- **Single device concept**: We outline a high-level concept for a resilient single device concept that is able to withstand common intrusion attempts. Associated with the hardware concept is a life cycle model that dictates the efficient use of these hardware/software concepts.
- **Cyber security risks**: We identify threat agents, misuse cases and the risks of a single device concept for multiple security domains and evaluate the most pressing issues.
- **Advanced security controls**: Based on the identified risks, we determine additional security controls that serve as mitigation measures and discuss their applicability.

We foresee our work to provide valuable insights into the requirements on and design decisions of highly secure mobile device solutions.

The remainder of the paper is organized as follows. Section 2 contains background and related work. In a risk assessment, Sect. 3 analyzes and evaluates the potential threats affecting a secure device for multiple security domains. Based upon this assessment, Sect. 4 aims to outline appropriate security controls to deal with the threats and describe a secure architecture. Finally, Sect. 5 concludes the paper.

[1] In the context of this work, the level **SECRET** is mostly relevant

2 Background and Related Work

This section outlines available devices on the market and describes the starting point of designing a new single device solution.

2.1 Challenges and considerations

The most important challenges and considerations for IT solutions capable of supporting multiple security domains[2] are:

- **Data Isolation:** When dealing with multiple security domains, one of the primary concerns is maintaining strict data isolation between domains. Each security domain may have its own set of security requirements, access controls, and data classification levels. Achieving data isolation can be challenging because traditional IT solutions often lack the necessary mechanisms to enforce strict separation.
- **Cross-Domain Information Flow:** In environments with multiple security domains, there might be a need to share information selectively between domains while ensuring proper security controls. Implementing mechanisms for controlled and secure information flow across domains can be complex and require specialized solutions. The National Cross Domain Strategy Management Office (NCDSMO)[3] provides information and guidelines on cross-domain solutions for securely transferring information between security domains.
- **Trusted Computing Base (TCB):** The TCB refers to the set of hardware, firmware, and software components that are critical to the security of a system. Supporting multiple security domains typically requires separate TCBs for each domain to ensure isolation and minimize the risk of unauthorized access or data leakage. Replacing the hard disk is often a part of establishing a dedicated TCB for each security domain. The Common Criteria for Information Technology Security Evaluation (CC)[4] is an international standard for evaluating security features and capabilities of IT products.
- **Secure Configuration Management:** To support multiple security domains, IT solutions need to accommodate distinct configurations, policies, and access controls for each domain. This includes managing and maintaining separate security profiles, software versions, and configurations to meet the specific requirements of each domain.
- **Compliance and Certification:** Deploying IT solutions in multi-domain environments often requires compliance with stringent security standards and certifications. Solutions need to undergo rigorous testing and evaluation to ensure they meet the necessary security requirements for each domain. Achieving and maintaining compliance can be time-consuming and resource-intensive.

[2] Since there is no need for simultaneous parallel operation of the device in different security domains, the implementation is more straightforward. Techniques that enable such operation, such as Kemmerer's resource matrix [Kemmerer, 1983], therefore do not necessarily have to be used.

[3] See https://www.ncdsmo.org/

[4] See https://www.commoncriteriaportal.org/

– **Resource Allocation and Efficiency:** In multi-domain environments, there is a need to optimize resource allocation to support the varying security requirements of different domains. This may involve segregating resources such as processing power, memory, and storage to prevent information leakage or unauthorized access.
– **Administration and User Management:** Managing users, access controls, and permissions across multiple security domains can be complex. IT solutions should provide robust user management capabilities, enabling administrators to define and enforce access policies specific to each domain while ensuring efficient user administration.
– **Scalability and Flexibility:** IT solutions should be scalable and flexible enough to accommodate future changes or additions of security domains. The architecture and design of the solutions should support easy expansion and integration with new security domains without significant disruptions or reconfiguration.

2.2 Market Overview

IT solutions are moving more and more in the direction of cloud computing. For this, three points are essential:

1. Connectivity
2. Data storage
3. Provision of services

If connectivity is not available in a sufficiently reliable way, an edge-computing capable device is necessary to mirror data locally.

The market for IT solutions catering to security level **SECRET** has seen significant growth and innovation in recent years. Government agencies, defense organizations, and other entities handling sensitive information require robust security measures to protect their data and systems. For instance, **SINA** (Secure Inter-Network Architecture[5]) is a popular solution that provides secure communication and data exchange capabilities. SINA Workstation is a high-security desktop solution that offers strong encryption, access control, and secure network connections. It enables users to access classified information and applications while maintaining confidentiality and integrity. Security elements are incorporated in shielded form and thus already require relatively high volumes of the equipment. Lightweight and portable devices are hardly possible with this technology. **Terminal servers** are widely used in secure environments to centralize computing resources and provide secure remote access. These servers enable multiple users to access applications and data from a central location while keeping the sensitive information within a secure environment. Terminal servers can be configured to meet the security requirements of **SECRET**-level environments, for instance Microsoft-based server solutions[6]. A good and reliable internet connection is necessary. This means that terminal servers are not well suited for edge-computing and mobile applications where constant data connections cannot be guaranteed or where operational reliability is a must

[5] Also see the SINA overview from the BSI: https://www.bsi.bund.de/dok/6603914
[6] See the Microsoft Remote Desktop Services https://docs.microsoft.com/en-us/windows-server/remote/remote-desktop-services/welcome-to-rds

even if the Internet connection fails. **Virtual Private Network (VPN) solutions** are essential for securing communications and ensuring the confidentiality of data transmitted over networks. VPNs establish encrypted connections between remote users and a secure network, making it suitable for accessing classified information securely. VPN solutions for **SECRET**-level security must adhere to stringent encryption standards and provide robust authentication mechanisms [Frankel et al., 2008]. The same as for terminal servers applies: a good and reliable Internet connections is mandatory.

Apart from the issues previously highlighted, all those solutions may not be suitable for use in multiple security domains without certain modifications or considerations.

In the area of mobile devices for military applications, the most important products, some of which are already being tested and used by the armed forces, are as follows:

- **Panasonic Toughbook**[7]**:** This series is well-known for its ruggedness and durability. These devices are designed to withstand harsh environments and are often used in industries such as military, public safety, and government. While they are popular for their rugged features, it is important to note that their specific suitability for specific security domains would depend on additional factors such as specific certifications, security configurations, and compliance with relevant standards.

- **Getac MX50 and Getac V110**[8]**:** Getac is another manufacturer known for producing rugged mobile devices. The MX50 is a fully rugged tablet, while the V110 is a convertible laptop with a detachable screen. Getac devices are commonly used in various industries that require durability and reliability. However, the suitability of these devices for specific security domains would depend on additional factors such as certifications, security configurations, and compliance with relevant standards specific to the domain.

- **RODA Rocky and RODA Panther**[9]**:** RODA is a manufacturer known for providing rugged mobile solutions. The Rocky and Panther series are rugged tablets designed for demanding environments. While these devices are built to withstand harsh conditions, their use in specific security domains would depend on certifications, security configurations, and compliance with relevant standards specific to the domain.

While all these devices are built to withstand harsh conditions, their use in specific security domains would depend on certifications, security configurations, and compliance with relevant standards specific to the domain, as listed above. In particular, using these devices consecutively in multiple security domains is not possible without additional and elaborate measures. In summary, when considering IT devices and solutions for use in multiple security domains up to level **SECRET**, it is possible to find suitable options in the form of terminal solutions. Terminal solutions, such as secure workstations or virtualized environments, can be designed and configured to meet the stringent security requirements and provide the necessary isolation between different security domains. However, when it comes to mobile devices like laptops and tablets in military environments, the availability of certified solutions for multiple security domains up to the **SECRET** level is limited. Mobile devices typically face additional challenges due to their portable nature, potential exposure to physical threats, and the complexity of ensuring secure data isolation and controlled information flow across domains.

[7] https://na.panasonic.com/us/computers-tablets-handhelds/computers/
[8] https://www.getac.com/en/products/
[9] https://www.roda-computer.com/products/

2.3 Architecture Outline

Based on preliminary work in cooperation with the Austrian Ministry of Defence (MoD), a rough architecture of a secure single device for multiple security domains has previously been sketched, which serves as the basis for the considerations carried out in this work. In the following, we briefly outline the components of this architecture. We then provide an overview of their interaction.

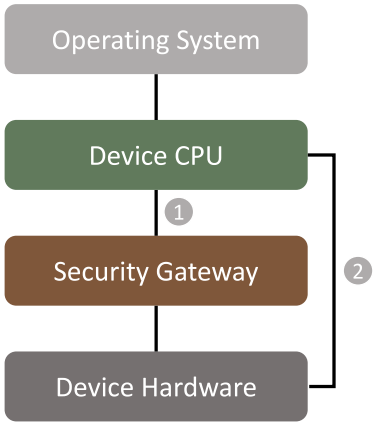

Figure 1: Architectural layers

Figure 1 shows an abstract layered architecture of the concept, Fig. 2 a schematic diagram of the communication between the components of the device. A central and, from a security perspective, most critical role is played by the security gateway, which 1) is responsible for the boot process on the tablet CPU and 2) controls (activates or deactivates) the interfaces and modules (e.g. camera, WiFi, cellular, GNSS, etc.) available to the tablet. After the boot process has been carried out on the tablet, the tablet can use the hardware visible to it and the security gateway takes over monitoring tasks in the background (e.g. monitoring the device integrity during operation). Reducing the number of connected interfaces already drastically limits the physical attack vectors.

The following device components are essential to establish a basic security concept:

Security gateway: A security gateway (SG) is built into the device, which takes over various security-critical tasks and plays a central role in the concept. This SG should also have access to additional secure elements built into the device in the form of Trusted Platform Modules (TPMs) [ISO/IEC 11889-1:2015, 2015]. In addition to device status monitoring and associated actions (e.g. deletion of all data on the device) and logging of derived events, the following additional security-critical tasks are also assumed:

- Control (activation and deactivation) of external and internal interfaces based on (external) configuration files;
- Verification of the boot media and if applicable the peripheral devices;
- Realisation of a secure boot process (from an external medium); and

– Management tasks for cryptographic keys and cryptographic operations (potentially in combination with a TPM).

Tablet CPU / device hardware: This is standard tablet hardware and there shall be the possibility to boot an operating system either from a boot medium. The chipset should be equipped with a secure element and in particular a TPM. The CPU is the unit on which the operating system runs. This also includes other hardware components, but these are not made explicit here.

Trusted Platform Module: A TPM is a secure element that can generate and store cryptographic key information and perform cryptographic operations. In addition, a TPM can be used to store and verify the status configuration of software (e.g. for a secure boot process of an operating system). To be able to use multiple external boot media together with Microsoft Windows BitLocker, a "Firmware TPM" (fTPM) must be used here. No more than two instances of Microsoft Windows can be associated with a regular hardware TPM. The use of BitLocker is required by the Austrian MoD. The term fTPM refers to a firmware-based implementation of a TPM, in this case the firmware of the SG.

Interfaces: External interfaces of the system, for instance USB.

Modules: Hardware modules of the device, for instance the camera module or GNSS module.

Sensors: Sensors that monitor the physical integrity of the device. These are intended to detect, for example, that the casing of the device has been damaged.

Boot stick: This is an external medium that is connected via USB and serves as data storage and also provides an operating system.

Operating System (OS): At the top level is the operating system and thus the interaction with the user. As mentioned before, this can be an external or internal OS, potentially supporting different OS distributions. Here it must be ensured that only authentic OSs may be booted by means of an authentic boot loader. In both cases, it should be ensured that all (important) user data is never unencrypted.

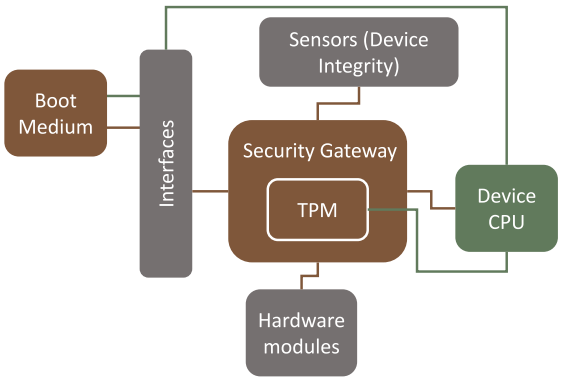

Figure 2: Schematic diagram of the communication between the components

2.4 Device Lifecycle

Figure 3 shows the entire life cycle, inspired by common IoT lifecycle models [Rahman et al., 2018], after manufacturing of the envisioned device. In the **provisioning** phase,

the device is delivered to device operator, registered and activated. After this initial step, the operator can configure the device in the *mission configuration* phase according to the needs of upcoming mission by creating templates with the intended operating system and device configuration, and apply both to the device. Next, the actual *operational use* in the field begins. The envisioned device should function as single device with a full hardware configuration. Depending on the mission, individual modules and interfaces can be made available or not. For some mission types, certain functionalities of the device can be disabled (e.g. data links, the camera, GNSS, etc.), according to the configuration in the previous phase. After the operational use, the device will be (temporarily) *decommissioned*. Here, all logs created during a mission are read out and analysed in order to be able to make conclusions about the use of the device, the occurrence of any anomalies or the device's condition, for example. In addition, all settings are reset so that no conclusions can be drawn about the past mission. If the unit was damaged during a mission or if other defects occurred, the optional *maintenance and repair* phase can now be entered. **Long-term decommissioning** of the unit is also optional - the unit can be prepared for long-term storage for this purpose. If the unit is to be re-used, the *recommissioning* phase begins. This is where the integrity and functionality of the gear is checked. If this is not the case, we can enter the *maintenance and repair* phase. If all tests are successful, the next step is the **mission configuration** phase. At the end of each devices' life the secure **equipment disposal** ensures that all data remnants are destroyed before a device is being destroyed and recycled.

3 Identifying the Risks

Figure 4 shows why identifying the risks that affect a device for multiple security domains is a crucial first step in defining a secure architecture for such a device. For this purpose, it is necessary to know and analyse potential threat agents that endanger the device and its boot media. In the next step, specific attack and misuse cases are assigned to each attacker type. These are then assessed in a qualitative risk assessment for criticality, probability of occurrence and risk. This subsequently serves as the basis for the selection of appropriate security controls of a secure device architecture in Sect. 4.

3.1 Threat Agent Profiles

In the context of creating abuse cases, a number of typical attackers were first defined. Each attacker type has a profile composed of different attributes as shown in Tab. 1 and Tab. 2. The following attacker types were defined:

- **Opportunistic thief**: The goal of the opportunistic thief is to steal a device with or without external boot media for the purpose of sale or personal use. It is not assumed that the end device or boot media will fall into the hands of a hostile analysis laboratory.
- **Inept user**: This user covers all cases of misuse that can be generated by misuse of a device without malicious intent.
- **Internal perpetrator user**: This user is an authorised user of the end device and at least one boot medium. The goal of this attacker is to manipulate or extract information stored on the end devices and boot media. The threat agent does not receive direct support from an external, hostile special laboratory; the end device is not stolen.

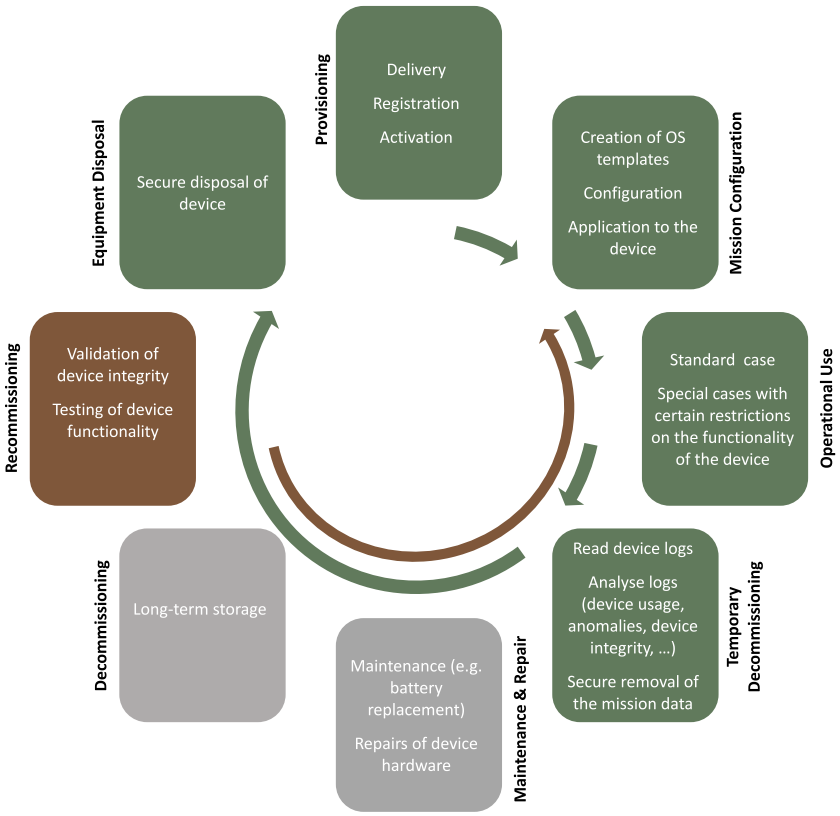

Figure 3: Device life cycle

- **Other internal perpetrator**: This internal perpetrator largely corresponds to the internal perpetrator user, but is not authorised to use end devices or boot media.
- **Internal perpetrator administrator**: This largely corresponds to the internal perpetrator user, but differs in skills, duration of temporary access and tools. Has access to the administration system, key material and has special IT knowledge.
- **Internal perpetrator maintenance**: This largely corresponds to the internal perpetrator user, but differs in skills, duration of temporary access and tools.
- **Hostile analysis laboratory**: It is assumed that this (state-sponsored) laboratory has a large number of expert personnel and special equipment. The end device and / or boot media remain in enemy hands, so there are no time restrictions for attacks.
- **Malicious service partner**: Here, one or more corrupted employees of the service workshop are assumed. They can receive support from an external, hostile analysis laboratory. The service workshop has special equipment, but must return the end device and has no access to boot media. The goal of this threat agent is to manipulate the end device.

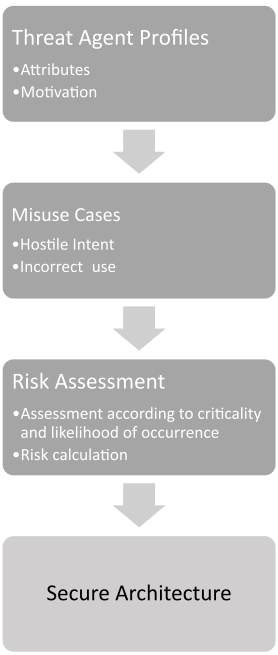

Figure 4: Approach of generating security requirements

3.2 Misuse Cases

In the cases of attack and misuse presented here, it is assumed that no special countermeasures and security mechanisms have been implemented. The device considered has the following attributes:

1. Physical construction: Water and dust resistant tablet device. Battery replacement or repair by an end user (i.e. also the armed forces) is not foreseen.
2. Use of external boot media on which the critical data and the operating system intended for the mission deployment are provided. In order to be able to boot the OS, the user must be authenticated.
3. Use of an internal boot medium for private use of the end device.
4. Standard security measures from the operating system ("lock screen" or "user authentication") are provided.

The design and selection of the misuse cases was modeled with the approach outlined by OWASP [OWASP, 2023]: After selecting five experts from different fields (a risk management expert, a cryptology expert, a penetration tester, a device software and hardware architect, and the product owner), a workshop was held to review the device use cases and identify potential misuse cases. In course of this 3 hours workshop, every expert suggested up to 10 misuse cases individually in a first round. In a second round overlaps and redundant cases were identified in a moderated group discussion, misuse cases then clustered depending on their nature and prioritized by the means of majority

	Threat Agents / Attributes	*N/A*	*Internal*					*External*	
		Oppor-tunistic thief	Inept user	Inside perpe-trator user	Other internal perpe-trator	Internal perpe-trator admin	Internal perpe-trator main-tenance	Hostile analysis lab	Malic. service partner
Attributes of the Threat Agent	**Expertise**								
	Non-specialist	✓	✓	✓					
	Skilled				✓		✓		
	Expert					✓			
	Several experts							✓	✓
	Equipment								
	Standard	✓	✓	✓			✓		
	Specialised				✓	✓			
	Custom-made							✓	✓
	Knowledge of architecture / system design of the end device.								
	Public	✓							
	Limited		✓	✓			✓		
	Confidential				✓	✓			
	Critical							✓	✓
Temporal Context	**Window of Opportunity**								
	Unlimited access								
	Easy		✓	✓	✓				
	Moderate					✓	✓		
	Difficult							✓	✓
	None								
	N/A (Irrelevant)	✓							
	Duration of time available for attackers								
	Seconds-minutes								
	Several hours				✓				
	Several days		✓	✓		✓	✓		✓
	Over a week							✓	
	N/A (Irrelevant)	✓							

Table 1: Threat agent profiles (Part 1)

voting (i.e., those that need more attention when designing counter measures based on their impact and likelihood of occurrence). Eventually, the group of experts identified 17 misuse cases (`[MUC:]`).

The first case considered is the act of a threat agent simply stealing the device and boot medium (`[MUC:DeviceAndBootDeviceStolen]`). The intention here is either to use the device themselves or to sell it. The threat agent has no interest in the data on the device or on attached memory. A similar situation occurs when only the device itself is stolen, without the boot medium (`[MUC:DeviceStolen]`).

The following cases are similar without malicious intent: the device can simply be lost by its designated user (`[MUC:DeviceLost]`). Of course, additionally the external boot

Aspects of the Threat: Attributes \ Threat Agents	N/A	Internal					External	
	Opportunistic thief	Inept user	Inside perpetrator user	Other internal perpetrator	Internal perpetrator admin	Internal perpetrator maintenance	Hostile analysis lab	Malic. service partner
Device state on access								
Powerd Off				✓	✓	✓	✓	✓
Standby		✓		✓	✓		✓	✓
Hibernated		✓		✓	✓	✓	✓	
Powered On / Logged Off		✓		✓	✓			
Powered On / Logged On		✓	✓		✓			
N/A	✓							
Access to ...								
Device	✓	✓	✓	✓	✓	✓	✓	✓
Boot medium (mission data)	✓	✓	✓	✓		✓	✓	
Boot medium (not configured)					✓	✓	✓	✓
Unlocking credentials for boot medium	(For personal use)	✓	✓				✓	
Unlocking credentials for OS	(For personal use)	✓	✓		✓		✓	
Device life cycle state								
Delivery state	✓			✓	✓	✓		
Configured	✓	✓	✓	✓	✓	✓	✓	✓
After Reset	✓			✓	✓	✓		✓

Table 2: Threat agent profiles (Part 2)

medium can be attached to the device, while it is in operation. Here, an important differentiation exists between a device with a locked screen (`[MUC:DeviceAndBootDeviceLost:Locked]`) and an unlocked screen (`[MUC:DeviceAndBootDeviceLost:Unlocked]`).

A threat agent ((knowingly or unknowingly) might also plug in corrupted external hardware (`[MUC:CompromisedHW]`). This does not refer to specially prepared devices that have been tailored to the end device, but storage media with malware like ransomware or remote access trojans. This can be part to gain access to the OS, sabotage a mission or extorting ransom.

Data extraction is covered by three different cases: An attempt for an extraction of large amounts of data without the use of any specialist tools is represented by `[MUC:DataExtraction:Simple]`. This includes extracting larger amounts of data via data interfaces or mass storage (no screenshots or photos of the device screen). More sophisticated attacks are represented by `[MUC:DataExtraction:Elaborated]`. These attacks range from side-channel attacks [Agrawal et al., 2002, Nikova et al., 2006], to breaking encryption, to microprobing attacks [Skorobogatov, 2011]. Of course, it is possible that only the boot medium is targeted (`[MUC:DataExtraction:BootDevice]`).

Similarly, but a continuous threat, is the automated leaking of data. In a military scenario, this could be critical mission data (e.g. orders, logistic information, tactical plans, etc.) or even location information of the device. For instance, an attacker may achieve this by manipulating the end device (firmware or software) to automatise data leakage (`[MUC:AutoDataLeak:SW]`). This includes malware in the system. Likewise, malicious external hardware can be used (`[MUC:AutoDataLeak:HW:External]`). For example, this includes bad USB attacks [SRLabs, 2014]. More sophisticated threat agents might even choose to manipulate the end device via internal hardware attacks to automatically leak data (`[MUC:AutoDataLeak:HW:Internal]`). The boot medium might also be targeted (`[MUC:AutoDataLeak:BootDevice]`) with, for example, depositing malware on boot media used by authorised users.

Manipulating the installed sensors / components could be made in order to cause misdecisions (`[MUC:Malfunction]`). Here the assumption was made that the attacker manipulates the device in such a way that it suddenly becomes unusable, e.g. in the field. The attack therefore impacts the functionality itself (availability) and less data security.

Very sophisticated attacks include the creation of dedicated backdoors to take control of a device. These attacks can target either the software (`[MUC:Backdoor:SW]`) or the hardware (`[MUC:Backdoor:HW]`). An example for this kind of attacks are attempts to exploit vulnerabilities in software and firmware within the booted operating system in order to access or modify the security gateway (`[MUC:PrivilegeEscalation]`).

3.3 Risk assessment

The risk of an attack and misuse case is classified with the help of a risk matrix [Garvey and Lansdowne, 1998] (see Fig. 5). As described by Markowski et al. [Markowski and Mannan, 2008], the basis for the risk matrix is the standard definition of risk as a combination of the severity of the consequences (impacts) that occur in a given scenario and its likelihood:

Risk = impact · likelihood

Very likely	Acceptable Risk (medium, 3)	Unacceptable Risk (high, 6)	Unacceptable Risk (high, 9)
Likely	Acceptable Risk (low, 2)	Acceptable Risk (medium, 4)	Unacceptable Risk (high, 6)
Unlikely	Acceptable Risk (low, 1)	Acceptable Risk (low, 1)	Acceptable Risk (medium, 3)
Likelihood / Impact	Low	Medium	High

Figure 5: Risk matrix

A qualitative assessment was conducted by the same experts that designed the misuse cases. In a further two hours workshop, the group collaborative identified plausible combinations of misuse case and threat agents. The final evaluation result was reached by consensus in extended round table discussions. Afterwards, the values determined for impact and likelihood were reviewed for plausibility by four members of the Austrian Ministry of Defence. No corrections of the expert group's initial estimations were made.

The results of this assessment can be found in Tab. 3. Please note, that the assessment in the case of malicious internal perpetrators was carried out on the assumption that the user had already been corrupted, and in the case of the hostile analysis laboratory on the assumption that the device had already been stolen.

During the risk assessment, it became apparent that special attention must be paid to preventing or impeding information leakage and the creation of software backdoors. Moreover, not only the device itself should be considered here, but also the boot medium. Similarly, measures to mitigate (simple) bulk data extraction should be implemented[10]. Particularly in the military context, not only the success of a mission can be jeopardized, but also the lives of the forces involved. Depending on the mission type and existing redundancies, the manipulation of sensors and components can also have a serious impact on mission success. This explains the high risk rating of `[MUC:Malfunction]`. Regarding the threat actors, very high risk is posed by rogue administrators, internal maintenance personnel and service partners. However, unlike the hostile analysis lab, these threat agents can be addressed directly.

4 Determining Security Controls

Once the threats and risks have been identified, appropriate security controls as effective counter measures are implemented. In this section we discuss the means to deal with the threats identified in the previous section. Relevant standards and best practice approaches, such as the Critical Security Controls [Joint Task Force, 2020] and the security controls collected by NIST [Center for Internet Security, 2021], were taken into account in determining the final controls.

4.1 Protection Objectives

This section describes the key protection objectives resulting from the misuse cases and risk assessment.

4.1.1 Authenticity of external hardware

The aim here is to determine whether external hardware, such as memory sticks, is allowed to communicate with the tablet. This can be achieved in different ways and therefore with different security guarantees:

– Allowing the connection of all devices over an interface enabled by the security gateway (no authenticity).

[10] It is important to point out that the assessment assumed that a large amount of sensitive information is present. If this is not the case, the impact should be assumed to be lower.

	Misuse Case	Impact	Likelihood	Risk
Opportunistic Thief	[MUC:DeviceAndBootDeviceStolen]	Low	Medium	2
	[MUC:DeviceStolen]	Low	Medium	2
Inept user	[MUC:CompromisedHW]	Medium	High	6
	[MUC:DeviceLost]	Low	Medium	2
	[MUC:DeviceAndBootDeviceLost:Locked]	Low	Low	1
	[MUC:DeviceAndBootDeviceLost:Unlocked]	Medium	Low	2
Inside perpetrator user	[MUC:DataExtraction:Simple]	High	High	9
	[MUC:AutoDataLeak:SW]	High	Medium	6
	[MUC:PrivilegeEscalation]	Medium	Medium	4
	[MUC:AutoDataLeak:HW:External]	Medium	Medium	4
	[MUC:AutoDataLeak:BootDevice]	High	Low	3
Other internal perpetrator	[MUC:DataExtraction:Simple]	High	Low	3
	[MUC:AutoDataLeak:SW]	High	Low	3
	[MUC:PrivilegeEscalation]	Medium	Low	3
	[MUC:AutoDataLeak:HW:External]	Medium	Low	2
	[MUC:AutoDataLeak:BootDevice]	High	Medium	6

	Misuse Case	Impact	Likelihood	Risk
Internal perpetrator administrator	[MUC:AutoDataLeak:HW:Internal]	High	Medium	6
	[MUC:AutoDataLeak:SW]	High	High	9
	[MUC:Malfunction]	Medium	Medium	4
	[MUC:AutoDataLeak:BootDevice]	High	High	9
Internal perpetrator maintenance	[MUC:AutoDataLeak:HW:Internal]	High	Medium	6
	[MUC:AutoDataLeak:SW]	High	Low	3
	[MUC:Malfunction]	Medium	Medium	4
	[MUC:AutoDataLeak:BootDevice]	High	Medium	6
Hostile analysis lab	[MUC:DataExtraction:Elaborated]	Medium	High	6
	[MUC:DataExtraction:BootDevice]	High	High	9
Malicious service partner	[MUC:AutoDataLeak:HW:Internal]	High	High	9
	[MUC:Malfunction]	High	High	9
	[MUC:Backdoor:SW]	High	High	9
	[MUC:Backdoor:HW]	High	Low	3

Table 3: Risk Assessment

– Allowing certain manufacturers or device types via allow / deny lists, as well as device class-specific or manufacturer-specific authentication (e.g. using the USB Type-C Authentication Program [USB Implementers Forum, 2016]). This feature must be implemented by the security gateway. However, this only results in weak authenticity guarantees.

– Active authentication of the hardware components (active component on the externally connected hardware) gives the strongest authenticity guarantees, but is more complex to implement and handle. This feature must be implemented by the security gateway and the type of authentication mechanisms supported may vary per device and vendor. This type of assurance of the authenticity of external hardware is considered very important especially when the security gateway has read access to this medium. Otherwise, in the absence of authentication, it cannot be ensured that the data read is not corrupted and could lead to the system being taken over. A widespread example of this scheme are electronic machine-readable travel documents (eMRTD), such as passports [International Civil Aviation Organization, 2021]. Here, it is used to authenticate the contactless integrated circuit (IC) of an eMRTD by signing a challenge sent by the document inspection system with a private key known only to the IC.

4.1.2 Authenticity of firmware

Here we distinguish between the firmware of the security gateway and the firmware for the boot process (UEFI / BIOS). In both cases, it must be ensured that only authentic

code is executed on the device.

4.1.3 Authenticity of software

Vital software components comprise the operating system boot process (boot loader) as well as the operating system itself (device drivers, modules in the operating system kernel). This guarantee is usually provided by secure boot mechanisms. However, implementation can involve varying effort depending on the specific system architecture. The authenticity of application programs must be guaranteed at operating system level (e.g. only software signed by trustworthy manufacturers may be installed) and is not considered in more detail here.

4.1.4 Authentication of users

Authentication of users can be achieved at different levels.

- Active authentication of users at the security gateway level does not seem to be necessary based on the current concept. However, active authentication of external components (USB devices) seems to be necessary.
- Pre-Boot Authentication: Active authentication of or interaction with the user may be necessary to unlock corresponding cryptographic keys for a secure boot process or hard disk encryption (e.g. Microsoft BitLocker).
- Finally, various means may be required for user authentication at the OS level (e.g. password / PIN entry, biometrics, external security tokens such as smart cards, ...). Here, care must be taken to ensure that the necessary sensors / interfaces are available and enabled.

If special mechanisms are to be used for user authentication (either pre-boot or at the OS), the components must be provided by the tablet or connected via external interfaces. A detailed consideration of these aspects is outside the scope of this paper.

4.1.5 Confidentiality and authenticity/integrity of data

It must be ensured that any data (including logging data) that is to be stored on internal or external fixed storage devices is only stored in encrypted form. It must also be ensured that, in addition to confidentiality, the integrity of this data is also guaranteed, i.e. that manipulations of this data can be detected. It may also be necessary to ensure the authenticity of this data (to make it assignable to a source). In addition, the authenticity of all data made available to the security gateway externally and processed by the security gateway (configurations, firmware updates) must be ensured, i.e. they must originate from trustworthy sources.

4.1.6 Secure cryptographic key management

Many of the methods mentioned above, as well as security guarantees, require the use of cryptographic methods and thus the generation and secure management of cryptographic keys. Here, care must be taken to ensure that cryptographic keys are generated using good random numbers and are never stored in a readable domain. Unless absolutely necessary, cryptographic keys should also never be kept in the tablet's volatile memory, but should only be used in secure elements (such as the TPM).

4.1.7 Secure erasure of data

With volatile memory, there is the problem that content remains available for a certain time even after a reset. Thus, especially when using secret cryptographic keys (if they are used directly by the security gateway, for example), it must be ensured that they are adequately deleted in volatile memory and not just released. Windows offers SecureZeroMemory [Microsoft, 2018] for this purpose, for example. In principle, however, there are a wide variety of strategies to achieve secure erasure.

4.1.8 Secure implementation of cryptographic methods

When using cryptographic methods, an important aspect is secure implementation. A basic requirement is that an implementation is free of typical security vulnerabilities, such as programming language-dependent vulnerabilities (e.g. buffer overflows or overflows in arithmetics), as well as defensive and secure programming techniques, e.g. validation of all input data, are used.

In addition to security vulnerabilities resulting from programming errors, there are a variety of other attacks that pose problems in the practical application of cryptographic techniques. Such attacks exploit information (side channels) resulting from the way of implementation but also from the construction methods used. These include computing time, power consumption, or the emission of electromagnetic fields. If these signals are dependent on the private keys or data used, this information can be utilized to carry out side-channel attacks. It should be noted here that the available side channels and possible countermeasures are platform-dependent, as software and hardware implementations are exposed to different attack scenarios. In particular, such signals can be measured via network traffic, power consumption (especially if an external power source is used), or with specialized hardware[11].

4.2 Technical Security Controls and Countermeasures

Based on the protection objectives described in Sect. 4.1, a set of potential technical countermeasures, organised in different categories, has been devised in order to prevent or mitigate cases of attack and misuse. Table 4 shows, which measures can be used to mitigate an identified attack or misuse case.

The first category covers measures regarding **encryption**. Encryption of user data on the end device `[MTG:TECH:CRYPT:01]` and encryption of (mission) data on external boot medium `[MTG:TECH:CRYPT:02]` form the basis for making it harder for an attacker to access critical information. Side-channel resistant cryptographic implementation `[MTG:TECH:CRYPT:03]` and management of key material to access mission data of external boot medium below OS level `[MTG:TECH:CRYPT:04]` ensure a more secure design of the encryption.

Authenticity verification ensures that components of the system are genuine by means of the verification of cryptographic signatures. This concerns the firmware of the device `[MTG:TECH:AUTHZ:01]`, the boot loader `[MTG:TECH:AUTHZ:02]` and the user applications, especially those that communicate with the SG `[MTG:TECH:AUTHZ:03]`. More difficult to implement, but nevertheless conceivable, are also active cryptographic

[11] A good introduction to side-channel attacks is given by Standaert [Standaert, 2010]. More detailed insights into side-channel attacks on mobile devices are presented by Spreitzer et al. [Spreitzer et al., 2017].

	Measure	[MUC:DeviceAndBoot DeviceStolen]	[MUC:DeviceStolen]	[MUC:CompromisedHW]	[MUC:DeviceLost]	[MUC:DeviceAndBoot DeviceLost:Locked]	[MUC:DeviceAnd BootDeviceLost:Unlocked]	[MUC:DataExtraction: Simple]	[MUC:DataExtraction: BootDevice]	[MUC:DataExtraction: Elaborated]	[MUC:AutoDataLeak:SW]	[MUC:AutoDataLeak: HW:External]	[MUC:AutoDataLeak: HW:Internal]	[MUC:PrivilegeEscalation]	[MUC:AutoDataLeak: BootDevice]	[MUC:Malfunction]	[MUC:Backdoor:SW]	[MUC:Backdoor:HW]
Encryption	[MTG:TECH:CRYPT:01]	✓	✓		✓	✓				✓								
	[MTG:TECH:CRYPT:02]	✓				✓		✓	✓		✓			✓	✓			
	[MTG:TECH:CRYPT:03]								✓	✓			✓					
	[MTG:TECH:CRYPT:04]							✓	✓		✓			✓			✓	
Checking Authenticity	[MTG:TECH:AUTHZ:01]								✓		✓			✓			✓	
	[MTG:TECH:AUTHZ:02]								✓		✓			✓			✓	
	[MTG:TECH:AUTHZ:03]										✓			✓				
	[MTG:TECH:AUTHZ:04]			✓								✓		✓	✓			
	[MTG:TECH:AUTHZ:05]			✓								✓		✓				
	[MTG:TECH:AUTHZ:06]			✓								✓		✓				
Authentification	[MTG:TECH:AUTHN:01]	✓				✓												
	[MTG:TECH:AUTHN:02]	✓				✓												
	[MTG:TECH:AUTHN:03]	✓				✓												
	[MTG:TECH:AUTHN:04]	✓				✓												
	[MTG:TECH:AUTHN:05]	✓	✓		✓	✓		✓									✓	
Software Hardening	[MTG:TECH:HRDSW:01]			✓				✓			✓							
	[MTG:TECH:HRDSW:02]							✓			✓							
	[MTG:TECH:HRDSW:03]			✓				✓										
	[MTG:TECH:HRDSW:04]			✓								✓						
	[MTG:TECH:HRDSW:05]										✓							
	[MTG:TECH:HRDSW:06]													✓				
	[MTG:TECH:HRDSW:06]																	✓
Hardware Hardening	[MTG:TECH:HRDHW:01]					✓				✓	✓		✓			✓		
	[MTG:TECH:HRDHW:02]								✓						✓			
	[MTG:TECH:HRDHW:03]			✓		✓						✓						
	[MTG:TECH:HRDHW:04]					✓				✓	✓		✓			✓		
Monitoring	[MTG:TECH:MONITOR:01]							✓			✓			✓		✓	✓	
	[MTG:TECH:MONITOR:02]							✓			✓			✓		✓	✓	
	[MTG:TECH:MONITOR:03]			✓								✓					✓	
	[MTG:TECH:MONITOR:04]									✓	✓			✓			✓	
	[MTG:TECH:MONITOR:05]										✓			✓				
	[MTG:TECH:MONITOR:06]	✓	✓		✓	✓												
	[MTG:TECH:MONITOR:07]							✓			✓						✓	
	[MTG:TECH:MONITOR:08]			✓								✓						
	[MTG:TECH:MONITOR:09]			✓								✓		✓				
Other	[MTG:TECH:OTHER:01]	✓	✓		✓													
	[MTG:TECH:OTHER:02]	✓	✓		✓													

Table 4: Mapping of attack and misuse cases to the individual technical mitigation measures

authentication of external boot media `[MTG:TECH:AUTHZ:04]` and external devices `[MTG:TECH:AUTHZ:05]` (via active components, for example smart cards). Apart from this sophisticated approach, hardware properties can also be used to determine whether components may be used `[MTG:TECH:AUTHZ:06]`. This “white listing” can be done via identifier, e.g. serial number, vendor ID, product ID, etc., however, this only offers significantly less protection than active authentication, as attackers can manipulate these identifiers.

A more broader category covers **authentication** in general. First and foremost, in order to boot from a boot device, a password or PIN must first be entered [MTG:TECH:AUTHN:01]. For highly critical missions, the use of a second factor (i.e., an NFC smart card) may be considered [MTG:TECH:AUTHN:02][12]. However, due to the nature of military missions and with difficult environmental conditions this actually very sensible security measure is not always feasible. To make brute-force attacks more challenging, two simple options are available: Forced waiting times between incorrect password entries [MTG:TECH:AUTHN:03] and locking (or even disabling) the boot medium [MTG:TECH:AUTHN:04]. Locking or disabling the external boot medium can also be a measure if certain predefined conditions occur; for example, certain user behavior (or the lack thereof), an unknown USB devices has been plugged in, or if the network was not connected for a predefined time [MTG:TECH:AUTHN:05].

Hardening is the process of reducing the system's vulnerability by reducing its attack surface. For **software components**, device specific measures are preventing the use of external mass storage devices [MTG:TECH:HRDSW:01] or blocking all data interfaces, for instance during critical missions [MTG:TECH:HRDSW:02]. Whitelisting of allowed external device classes (e.g. no human interface devices (HIDs), network interface cards (NICs), etc.) can prevent the exfiltration of critical data or remote access [MTG:TECH:HRDSW:03]. This measure is more effective when also blocking changing the device class of an (external) hardware component during operation [MTG:TECH:HRDSW:04] (i.e. a keyboard changes its class to a network interface). Similarly to hardware, whitelisting of allowed applications is also conceivable [MTG:TECH:HRDSW:05]. To prevent unauthorized re-use, the microcontroller of the security gateway (cf. Fig. 2) should not be directly controllable by the OS [MTG:TECH:HRDSW:06]. Furthermore, trustworthy computing on non-trusted components [Götzfried, 2017] techniques can further increase the security [MTG:TECH:HRDSW:07].

Hardening of **hardware components** includes preventive measures like creating a tamper-resistant enclosure of the device (physical hardening) [MTG:TECH:HRDHW:01], the physical hardening of external boot media (e.g. by sealing the flash memory) [MTG:TECH:HRDHW:02] and avoiding external interfaces with direct memory access (DMA) (e.g. FireWire, eSATA, PCIe, etc.) [MTG:TECH:HRDHW:03]. A detective measure is the active monitoring of the device casing integrity (with subsequent locking of the device after tampering attempts have been detected) [MTG:TECH:HRDHW:04].

Monitoring and logging of the software components are the basis for a large set of counter measures, as well as the the post-detection of security events. This includes logging of device level activities [MTG:TECH:MONITOR:01], logging of operating system level activities [MTG:TECH:MONITOR:02] and logging of presence and type of external devices [MTG:TECH:MONITOR:03]. To detect unauthorised modifications, integrity monitoring at firmware level [MTG:TECH:MONITOR:04] and at OS level [MTG:TECH:MONITOR:05] can be employed. Tracking of the device via position detection (e.g. proximity, GPS, WLAN, etc.) and communication channel to the backend [MTG:TECH:MONITOR:06] can be used as a basis for anomaly detection or geofencing. Anomaly detection can further be used to detect improper access to mission data [MTG:TECH:MONITOR:07]. If the device is connected to a backend, all anomalies and possible security events can be reported in real-time [MTG:TECH:MONITOR:08]. Monitoring for anomalies when using attached components is also part of this category [MTG:TECH:MONITOR:09].

[12] The guidelines on multi-factor authentication (MFA) by NIST should be considered here. [Grassi et al., 2020]

Other measures include the support of remote management and remote wiping [MTG:TECH:OTHER:01], and the employment of an administration application that enables reconfiguration (which prevents the unauthorized re-use of the device) [MTG:TECH:OTHER:02].

4.3 Organisational Countermeasures

In addition to purely technical measures, it is important not to lose sight of the fact that many attack vectors arise due to human weaknesses or inadequate processes. It is therefore essential to take organisational measures to ensure secure operations. It is assumed that an organisation operating the end device in a critical security context has at least implemented the usual security-related standards such as [Wikipedia, 2023] or BSI IT Grundschutz [Bundesamt für Sicherheit in der Informationstechnik, 2021]. Other non-technical measures include setting up procedures or policies for handling the end device, external boot media and management system, as detailed in the following.

Studies suggest that awareness training can contribute well in reducing successful social engineering attacks [Bullée et al., 2015, Conteh and Schmick, 2016], therefore, conducting awareness trainings is one initial but essential measure to increase the overall security in operating the device [MTG:ORG:01]. For critical configuration work, as well as for handling security-relevant key material, following the four-eyes principle, which requires two individuals to approve an action before it can be taken, can help to counter misuse or even human error [MTG:ORG:02]. For limiting the access to critical management system servers or PKI infrastructure, those should be only operated in a secure environment [MTG:ORG:03]. This is also true for the configuration of external boot media [MTG:ORG:04]. External boot media, which has been distributed, must always remain with the authorized user[13] [MTG:ORG:05]. To prevent, for example Evil Maid attacks[14] [F-Secure, 2020], the devices should be placed in a secure environment for safekeeping [MTG:ORG:06]. If a second factor is used for unlocking a boot medium, those should also be placed in a secure environment [MTG:ORG:07]. The same holds true for configured external boot media [MTG:ORG:08] and data carriers and devices (e.g. radios) enabled for use with the device [MTG:ORG:09]. To detect malicious modifications, external devices, boot media and their USB cables[15], should be checked regularly [MTG:ORG:10]. Powered-on devices should never be left unattended [MTG:ORG:11]. If vulnerabilities become known and can be patched, it is necessary to apply these updates immediately. This applies to the device [MTG:ORG:12] as well as to critical applications on the operating system (this includes also the device administration applications) [MTG:ORG:13]. To prevent any mix-up and to detect a possible loss more quickly, it is a good practice to clearly label external boot media [MTG:ORG:14]. Performing regular audits of assigned rights for device and management systems [MTG:ORG:15] helps to detect vulnerabilities in the software components or configuration. Regularly changing administrator passwords in conjunction with a password policy[16] [MTG:ORG:16] can help to counter password leaking or guessing. To prevent possible unauthorized actions

[13] This is analogous to guidelines of the Austrian Armed Forces, for example, guards must never let their weapons out of their hands and must not let superiors take them from them.

[14] An evil maid attack is an attack on an unattended device, in which an attacker with physical access alters it in some undetectable way so that they can later access the device, or the data on it.

[15] This is necessary, as modified USB cables can also be used as an attack vector. [mg.lol, 2019]

[16] A good introduction to the subject of password policies is given by Summers and Bosworth [Summers and Bosworth, 2004]. A maximum validity period of 45 to 60 days for passwords is

by system users, separation of administrative roles[17] is highly recommended [MTG:ORG:17]. The professional and controlled destruction of external and internal disks from after their end-of-life is important to prevent data leakage[18] [MTG:ORG:18]. This approach should also be followed for mass storage used in management systems, after these are at their end-of-life [MTG:ORG:19].

4.4 Supply Chain

When selecting partners and vendors in the supply chain, care needs to be taken to ensure that only certified and verified organizations are contracted with. Attacks via the supply chain cannot be covered by the architecture, so these are excluded from the consideration of misuse cases. For example, if the end device is to be used at the "confidential" security level, recognized / certified cryptographic component manufacturers must be used.

Software supply chain security has already been recognized as a major issue. For instance, the U.S. White House has issued an executive order covering this topic [Biden, 2021]. One of the measures proposed there is also highly relevant to the implementation of the device in our work: The introduction of a "Software Bill of Materials" (SBOM), which contains the details and supply chain relationships of various open source and third-party components used in building software in a formal record. The SBOM, which helps to maintain accurate and up-to-date data, can then be used as a starting point for regular audits and (automatically) checking for known vulnerabilities. This also concerns the dependencies of the used components to further third party components – dependencies between those components should also be tracked. Of course, with any third-party software component, it is important to consider the source of the software, the vendor's track record for dealing with vulnerabilities, and whether it is continuously maintained. For more detail, see the Cloud Native Computing Foundation's best practices for software supply chain [Cloud Native, 2021]. Furthermore, the secure handling of third-party components is also part of the Secure Software Development Framework (SSDF) [Souppaya et al., 2022], which should be referred to for the development of a device such as the one in the focus of this paper.

Attacks on the supply chain also include attacks by means of dedicated hardware trojans. Various methods can be used to defend against these [Xiao et al., 2016]:

– *Trojan detection* aims at verifying existing designs and manufactured ICs (integrated circuit) without additional circuitry. They are performed either in the design phase (i.e., presilicon) to validate IC designs or after the manufacturing phase (i.e., post silicon) to verify manufactured ICs. Presilicon methods include functional validation (functional testing), behavioural [Zhang and Tehranipoor, 2011] or structural [Hicks et al., 2010] Hardware Description Language (HDL) analysis, as well as formal verification [Zhang and Tehranipoor, 2011, Rathmair et al., 2014, Love et al., 2011]. Post-silicon methods also include functional testing [Bhunia et al., 2014, Chakraborty and

---

also recommended in this document. Newer studies (e.g. from BSI [Bundesamt für Sicherheit in der Informationstechnik, 2021]) indicate that longer passwords allow longer duration.

[17] Separation of duties or separation of privileges are concepts from information security that have been known and used since the 1970s [Saltzer and Schroeder, 1975, Simon and Zurko, 1997]. In the context of this work, this could mean, for example, separating the roles "assignment of user rights" and "configuration of end devices".

[18] Even when overwriting flash memories, it is not always possible to ensure that they are completely deleted [Wei et al., 2011]. Further procedures for dealing with data carriers that are no longer required are described in the guideline SP 800-88 [Kissel et al., 2014].

Bhunia, 2009], and side-channel analysis approaches (e.g., delays [Jin and Makris, 2008], power consumption [Agrawal et al., 2007] and leakage power [Aarestad et al., 2010], temperature [Forte et al., 2013], or radiation behaviour [Stellari et al., 2014]).

– *Design-for-Trust (DfT)* targets the design phase and can be divided into three classes: The first DfT class aims to support trojan detection. The second class of techniques includes prevention measures to make it harder for an attacker to understand a design. Techniques here are logic obfuscation [Roy et al., 2010, Baumgarten et al., 2010, Wendt and Potkonjak, 2014], camouflaging [Bi et al., 2014, Rajendran et al., 2013, Cocchi et al., 2014], and functional filler cell [Xiao and Tehranipoor, 2013] (unused areas in an IC design are filled cells that have no functionality). The third class is trustworthy computing on untrusted components. The difference between runtime monitoring and trustworthy computing is that trustworthy computing is a priori tolerant to trojan attacks [McIntyre et al., 2010, Keren et al., 2010, Liu et al., 2014].

– *Split manufacturing for trust* [Xiao et al., 2016] is an approach in which a semiconductor design is split into front end of line (FEOL) and back end of line (BEOL) parts that can be manufactured in different semiconductors' fabs. An untrusted semiconductor fab performs the (more expensive) FEOL fabrication and then delivers wafers to a trusted foundry for (less expensive) BEOL fabrication. The untrusted semiconductor factory does not have access to the BEOL layer and therefore cannot insert hardware Trojans at these "secure" locations.

4.5 Maintenance and Service Partners

As with the selection of the supply chain, care must also be taken with maintenance and service partners to ensure that only certified and audited organizations are used. Strict contractual obligations and enforceable SLAs are advisable. As demanded by ISO 27001, the requirements for confidentiality or non-disclosure agreements reflecting the organization's needs for the protection of information shall be identified, regularly reviewed and documented. Additionally, background verification checks on all personnel of service partners and contractors shall be carried out in accordance with relevant laws, regulations and ethics and shall be proportional to the business requirements, the classification of the information to be accessed and the perceived risks.

In Austria, for example, it may make sense for service personnel to be subjected to a reliability check by the MoD in accordance with sections 23 and 24 of the Militärbefugnisgesetz (German for *"military authorization act"*) [Rechtsinformation, 2021, Bundeskanzleramt Büro der Informationssicherheitskommission, 2018]. Regular and unannounced audits of maintenance and service partners, in addition to the implementation of the industry standards ISO/IEC 27000 series [Wikipedia, 2023] or BSI IT Grundschutz [Bundesamt für Sicherheit in der Informationstechnik, 2021], are also advisable.

Additional measures for external workshops could be the enforcement of a dual control principle for critical maintenance work on the end device (for example, when opening the housing) and / or video surveillance during maintenance work. External workshops must also ensure that no tampered external media are plugged in. For better traceability, there should also be a record of the maintenance employee(s), including time, duration and the devices they are working on.

5 Conclusion and Future Work

This paper reported foundational work of a single device solution that is applicable to multiple security domains. We specifically focused on cyber security issues and delivered a concept for a device that is able to isolate mission-specific data from each other without the risk of data spillover. Besides a high level concept for a resilient single device concept that is able to withstand common intrusion attempts, we had a detailed look into associated cyber security risks. Here, we identified threat agents, misuse cases and the risks, and evaluated the most pressing issues. Based on the identified risks, we determined additional security controls that serve as mitigation measures and discuss their applicability. With these contributions, we foresee our work of a structured approach to designing appropriate solutions valuable for the community, and provide valuable insights into the requirements on and design decisions of highly secure mobile device solutions.

Future work includes the continuous application of this concept together with the industry partner MUSE Electronics GmbH to deliver a prototype of a mobile device that has been designed and implemented following the approach in this paper to the Austrian Armed Forces, and the successful evaluation and validation of this device by security experts in the military domain.

Acknowledgements

This work has been funded by the Austrian defense research programme FORTE of the Federal Ministry of Finance (BMF) in course of the project SD4MSD (879678).

References

[Aarestad et al., 2010] Aarestad, J., Acharyya, D., Rad, R., and Plusquellic, J. (2010). Detecting Trojans Through Leakage Current Analysis Using Multiple Supply Pad. IEEE Transactions on information forensics and security, 5(4):893–904. Publisher: IEEE.

[Agrawal et al., 2002] Agrawal, D., Archambeault, B., Rao, J. R., and Rohatgi, P. (2002). The em side-channel (s). In CHES, volume 2, pages 29–45. Springer.

[Agrawal et al., 2007] Agrawal, D., Baktir, S., Karakoyunlu, D., Rohatgi, P., and Sunar, B. (2007). Trojan detection using IC fingerprinting. In 2007 IEEE Symposium on Security and Privacy (SP'07), pages 296–310. IEEE.

[Agre et al., 2013] Agre, J. R., Gordon, K. D., and Vassiliou, M. S. (2013). Commercial technology at the tactical edge. Technical report, INSTITUTE FOR DEFENSE ANALYSES ALEXANDRIA VA.

[Baumgarten et al., 2010] Baumgarten, A., Tyagi, A., and Zambreno, J. (2010). Preventing IC piracy using reconfigurable logic barriers. IEEE design & Test of computers, 27(1):66–75. Publisher: IEEE.

[Bhunia et al., 2014] Bhunia, S., Hsiao, M. S., Banga, M., and Narasimhan, S. (2014). Hardware Trojan attacks: Threat analysis and countermeasures. Proceedings of the IEEE, 102(8):1229–1247. Publisher: IEEE.

[Bi et al., 2014] Bi, Y., Gaillardon, P.-E., Hu, X. S., Niemier, M., Yuan, J.-S., and Jin, Y. (2014). Leveraging emerging technology for hardware security-case study on silicon nanowire fets and graphene symfets. In 2014 IEEE 23rd asian test symposium, pages 342–347. IEEE.

[Biden, 2021] Biden, J. R. J. (2021). Executive Order (EO) 14028, ”Improving the Nation’s Cybersecurity”.

[Bullée et al., 2015] Bullée, J.-W. H., Montoya, L., Pieters, W., Junger, M., and Hartel, P. H. (2015). The persuasion and security awareness experiment: reducing the success of social engineering attacks. Journal of experimental criminology, 11(1):97–115. Publisher: Springer.

[Bundesamt für Sicherheit in der Informationstechnik, 2021] Bundesamt für Sicherheit in der Informationstechnik (2021). IT-Grundschutz-Kompendium. Bonn.

[Bundeskanzleramt Büro der Informationssicherheitskommission, 2018] Bundeskanzleramt Büro der Informationssicherheitskommission (2018). Merkblatt Industrielle Sicherheit.

[Center for Internet Security, 2021] Center for Internet Security (2021). CIS Critical Security Controls.

[Chakraborty and Bhunia, 2009] Chakraborty, R. S. and Bhunia, S. (2009). Security against hardware Trojan through a novel application of design obfuscation. In 2009 IEEE/ACM International Conference on Computer-Aided Design-Digest of Technical Papers, pages 113–116. IEEE.

[Cloud Native, 2021] Cloud Native (2021). Software Supply Chain Best Practices.

[Cocchi et al., 2014] Cocchi, R. P., Baukus, J. P., Chow, L. W., and Wang, B. J. (2014). Circuit camouflage integration for hardware IP protection. In 2014 51st ACM/EDAC/IEEE Design Automation Conference (DAC), pages 1–5. IEEE.

[Conteh and Schmick, 2016] Conteh, N. Y. and Schmick, P. J. (2016). Cybersecurity: risks, vulnerabilities and countermeasures to prevent social engineering attacks. International Journal of Advanced Computer Research, 6(23):31. Publisher: Accent Social and Welfare Society.

[F-Secure, 2020] F-Secure (2020). F-Secure’s Guide to Evil Maid Attacks.

[Forte et al., 2013] Forte, D., Bao, C., and Srivastava, A. (2013). Temperature tracking: An innovative run-time approach for hardware Trojan detection. In 2013 IEEE/ACM International Conference on Computer-Aided Design (ICCAD), pages 532–539. IEEE.

[Frankel et al., 2008] Frankel, S., Hoffman, P., Orebaugh, A., and Park, R. (2008). SP 800-113 Guide to SSL VPNs.

[Garvey and Lansdowne, 1998] Garvey, P. R. and Lansdowne, Z. F. (1998). Risk matrix: an approach for identifying, assessing, and ranking program risks. Air Force Journal of Logistics, 22(1):18–21.

[Götzfried, 2017] Götzfried, J. (2017). Trusted Systems in Untrusted Environments: Protecting against Strong Attackers. PhD thesis, Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU).

[Grassi et al., 2020] Grassi, P. A., Newton, E. M., Perlner, R. A., Regenscheid, A., Fenton, J. L., Burr, W. E., and Richer, J. P. (2020). SP 800-63B Digital Identity Guidelines.

[Hicks et al., 2010] Hicks, M., Finnicum, M., King, S. T., Martin, M. M., and Smith, J. M. (2010). Overcoming an untrusted computing base: Detecting and removing malicious hardware automatically. In 2010 IEEE symposium on security and privacy, pages 159–172. IEEE.

[International Civil Aviation Organization, 2021] International Civil Aviation Organization (2021). Doc 9303, Machine Readable Travel Documents - Part 11: Security Mechanisms for MRTDs.

[ISO/IEC 11889-1:2015, 2015] ISO/IEC 11889-1:2015 (2015). ISO/IEC 11889-1:2015 Information technology – Trusted platform module library – Part 1: Architecture.

[Jin and Makris, 2008] Jin, Y. and Makris, Y. (2008). Hardware Trojan detection using path delay fingerprint. In 2008 IEEE International workshop on hardware-oriented security and trust, pages 51–57. IEEE.

[Joint Task Force, 2020] Joint Task Force (2020). NIST Special Publication 800-53 Revision 5: Security and Privacy Controls for Information Systems and Organizations - Control Catalog Spreadsheet.

[Kemmerer, 1983] Kemmerer, R. A. (1983). Shared resource matrix methodology: An approach to identifying storage and timing channels. ACM Transactions on Computer Systems (TOCS), 1(3):256–277.

[Keren et al., 2010] Keren, O., Levin, I., and Karpovsky, M. (2010). Duplication based one-to-many coding for Trojan HW detection. In 2010 IEEE 25th International Symposium on Defect and Fault Tolerance in VLSI Systems, pages 160–166. IEEE.

[Kissel et al., 2014] Kissel, R., Scholl, M., Stine, K., and Regenscheid, A. (2014). SP 800-88 Rev. 1. Guidelines for Media Sanitization.

[Kurt et al., 2016] Kurt, O., Kalem, G., Vayvay, O., and Kalender, Z. T. (2016). The role of mobile devices and applications in supply chains. International Journal of Economics and Management Systems, 1.

[Liu et al., 2014] Liu, Y., Huang, K., and Makris, Y. (2014). Hardware Trojan detection through golden chip-free statistical side-channel fingerprinting. In Proceedings of the 51st Annual Design Automation Conference, pages 1–6.

[Love et al., 2011] Love, E., Jin, Y., and Makris, Y. (2011). Enhancing security via provably trustworthy hardware intellectual property. In 2011 IEEE international symposium on hardware-oriented security and trust, pages 12–17. IEEE.

[Markowski and Mannan, 2008] Markowski, A. S. and Mannan, M. S. (2008). Fuzzy risk matrix. Journal of hazardous materials, 159(1):152–157.

[McIntyre et al., 2010] McIntyre, D., Wolff, F., Papachristou, C., and Bhunia, S. (2010). Trustworthy computing in a multi-core system using distributed scheduling. In 2010 IEEE 16th International On-Line Testing Symposium, pages 211–213. IEEE.

[mg.lol, 2019] mg.lol (2019). O.MG Cable: https://mg.lol/blog/omg-cable/.

[Microsoft, 2018] Microsoft (2018). SecureZeroMemory function.

[MIL-STD-810H, 2019] MIL-STD-810H (2019). MIL-STD-810H: Environmental Engineering Considerations and Laboratory Tests . Standard, U.S. Department of Defense.

[Nikova et al., 2006] Nikova, S., Rechberger, C., and Rijmen, V. (2006). Threshold implementations against side-channel attacks and glitches. In ICICS, volume 4307, pages 529–545. Springer.

[North Atlantic Treaty Organization, 2015] North Atlantic Treaty Organization (2015). NATO SDIP-27. Standard.

[North Atlantic Treaty Organization, 2023a] North Atlantic Treaty Organization (2023a). Global Rugged Devices Market 2023-2027.

[North Atlantic Treaty Organization, 2023b] North Atlantic Treaty Organization (2023b). NATO: For Your Eyes Only.

[OWASP, 2023] OWASP (2023). Abuse Case Cheat Sheet.

[Rahman et al., 2018] Rahman, L. F., Ozcelebi, T., and Lukkien, J. (2018). Understanding iot systems: a life cycle approach. Procedia computer science, 130:1057–1062.

[Rajendran et al., 2013] Rajendran, J., Sam, M., Sinanoglu, O., and Karri, R. (2013). Security analysis of integrated circuit camouflaging. In Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security, pages 709–720.

[Rathmair et al., 2014] Rathmair, M., Schupfer, F., and Krieg, C. (2014). Applied formal methods for hardware Trojan detection. In 2014 IEEE International Symposium on Circuits and Systems (ISCAS), pages 169–172. IEEE.

[Rechtsinformation, 2021] Rechtsinformation (2021). Bundesgesetz über Aufgaben und Befugnisse im Rahmen der militärischen Landesverteidigung (Militärbefugnisgesetz –MBG)StF: BGBl. I Nr. 86/2000 (NR: GP XXI RV 76 AB 218 S. 33. BR: AB 6203 S. 667.).

[Roy et al., 2010] Roy, J. A., Koushanfar, F., and Markov, I. L. (2010). Ending piracy of integrated circuits. Computer, 43(10):30–38. Publisher: IEEE.

[Saltzer and Schroeder, 1975] Saltzer, J. H. and Schroeder, M. D. (1975). The protection of information in computer systems. Proceedings of the IEEE, 63(9):1278–1308.

[Simon and Zurko, 1997] Simon, R. T. and Zurko, M. E. (1997). Separation of duty in role-based environments. In Proceedings 10th Computer Security Foundations Workshop, pages 183–194. IEEE.

[Skorobogatov, 2011] Skorobogatov, S. (2011). Physical attacks on tamper resistance: progress and lessons. In Proc. of 2nd ARO Special Workshop on Hardware Assurance, Washington, DC.

[Souppaya et al., 2022] Souppaya, M., Scarfone, K., and Dodson, D. (2022). NIST Special Publication 800-218: Secure Software Development Framework.

[Spreitzer et al., 2017] Spreitzer, R., Moonsamy, V., Korak, T., and Mangard, S. (2017). Systematic classification of side-channel attacks: A case study for mobile devices. IEEE Communications Surveys & Tutorials, 20(1):465–488.

[SRLabs, 2014] SRLabs (2014). What is BadUSB? https://opensource.srlabs.de/projects/badusb.

[Standaert, 2010] Standaert, F.-X. (2010). Introduction to side-channel attacks. Secure integrated circuits and systems, pages 27–42.

[Stellari et al., 2014] Stellari, F., Song, P., Weger, A. J., Culp, J., Herbert, A., and Pfeiffer, D. (2014). Verification of untrusted chips using trusted layout and emission measurements. In 2014 IEEE international symposium on hardware-oriented security and trust (HOST), pages 19–24. IEEE.

[Summers and Bosworth, 2004] Summers, W. C. and Bosworth, E. (2004). Password policy: the good, the bad, and the ugly. In Proceedings of the winter international synposium on Information and communication technologies, pages 1–6.

[U.S. Department of Defense, 2015] U.S. Department of Defense (2015). MIL-STD-461G: Requirements for the Control of Electromagnetic Interference Characteristics of Subsystems and Equipment. Standard.

[USB Implementers Forum, 2016] USB Implementers Forum (2016). USB 3.0 Promoter Group Defines Authentication Protocol for USB Type-C.

[Wei et al., 2011] Wei, M. Y. C., Grupp, L. M., Spada, F. E., and Swanson, S. (2011). Reliably Erasing Data from Flash-Based Solid State Drives. In FAST, volume 11, pages 8–8.

[Wendt and Potkonjak, 2014] Wendt, J. B. and Potkonjak, M. (2014). Hardware obfuscation using PUF-based logic. In 2014 IEEE/ACM International Conference on Computer-Aided Design (ICCAD), pages 270–271. IEEE.

[Wikipedia, 2023] Wikipedia (2023). ISO/IEC-27000-series.

[Xiao et al., 2016] Xiao, K., Forte, D., Jin, Y., Karri, R., Bhunia, S., and Tehranipoor, M. (2016). Hardware trojans: Lessons learned after one decade of research. ACM Transactions on Design Automation of Electronic Systems (TODAES), 22(1):1–23. Publisher: ACM New York, NY, USA.

[Xiao and Tehranipoor, 2013] Xiao, K. and Tehranipoor, M. (2013). BISA: Built-in self-authentication for preventing hardware Trojan insertion. In 2013 IEEE international symposium on hardware-oriented security and trust (HOST), pages 45–50. IEEE.

[Zhang and Tehranipoor, 2011] Zhang, X. and Tehranipoor, M. (2011). RON: An on-chip ring oscillator network for hardware Trojan detection. In 2011 Design, Automation & Test in Europe, pages 1–6. IEEE.