Common Approach to Functional Safety and System Security in Building Automation and Control Systems

Thomas Novak[1], Albert Treytl[1,2], Peter Palensky[1]

[1)] Vienna University of Technology,
Institute of Computer Technology
Gusshausstrasse 27-29
1040 Vienna, Austria
{novakt, treytl, palensky}@ict.tuwien.ac.at

[2)] Austrian Academy of Sciences
Research Unit for Integrated Sensor Systems
Viktor-Kaplan-Strasse 2
2700 Wiener Neustadt, Austria
Albert.Treytl@oeaw.ac.at

Abstract

Building automation and control systems (BACS) are an important part of modern automated buildings. More and more they are also responsible for functions affecting people's safety, security and health. Thus the respective technology is supposed to work reliably, securely, safely and efficiently. The two important features of such a BACS are functional safety and system security (short safety and security) of both the network nodes and the communication protocols. Up to now little effort has been made to specify a life cycle for a safe and secure BACS that defines requirements for the different stages of the product life of a BACS. Special focus is related to the commonalities between the development of safety and security systems to benefit from these commonalities in development.

1. Introduction and Problem Statement

Building automation and control systems (BACS) are often integrated into modern buildings. More and more modern BACS go beyond trivial control or measurement tasks. Their importance for the processes of a building (climate, logistics, etc.) is constantly growing. They also become responsible for functions that affect people's safety and security. Due to social developments and personal safety desires it is absolutely necessary that modern BACS feature functional safety (short safety) and network and system security (short security) of the network nodes and the communication protocols.

Today's BACS typically lack real security features [1]. In fact, most of them are not considered secure at all although effort is made to integrate such features. A solution for a BACS is presented in [19]. Speaking of functional safety, first promising extensions of standard BACS are currently making their way to the market. Functional safety of these systems, however, is compromised by their intrinsic security flaws. There is no real safety without security: proper measures to grant confidentiality, integrity, availability (CIA) of data as well as efficient access control. In fact, security must be seen as actually supporting safety, instead of hindering it and vice versa.

Harmonizing safety and security is not a new topic in literature. Eams [2] investigated safety and security requirements specification methods in the context of an air control system and problems relating to their independent development. Stavridou [3] and Simpson [4] discussed the relevance of the security concept of non-interference to safety related properties. Stoneburner [15] presented a unified security/safety risk framework.

The discipline called dependability pursues the idea of an unified approach. According to Laprie [5] dependability is an integrative concept. A dependable system is characterized by the following attributes: availability, reliability, safety, confidentiality, integrity and maintainability. Dewsbury [6] presented a dependability model for domestic systems.

Although a lot of research has been done on this topic, up to now there have not been any guidelines, technical specifications or standards for an open-standard BACS that give requirements for specifying a safe and secure BACS. It is not documented publicly how to benefit from the commonalities of safety and security during development of a BACS and how to deal with contradictions between both areas.

As a consequence an approach to specify a safe and secure system is being presented in the following that specifies different stages in the pre-design phase of a safe and secure BACS. It is the first phase of a safe and secure life cycle model. It harmonizes the safety and security disciplines by giving requirements for the various stages. The approach does not focus on particular applications, but concentrates on the properties of BACS to maintain their today's flexible utilization in a safe and secure way.

The remainder of the document is structured as follows: section 2 presents an approach to develop a safety related system according to the international

1-4244-0826-1/07/$20.00 © 2007 IEEE

standard IEC 61508. Section 3 deals with security issues mentioned in Common Criteria. Section 4 points out the approach to develop a safe and secure system. Section 5 discusses the approach while section 6 outlines the usage of the presented approach by means of a practical application.

2. Safety – IEC 61508

In the last years the need for establishing a technology for safety related data communication with BACS has been increased due to recent political and technical developments. A new international standard IEC 61508 [6] was developed and published that gives requirements for programmable electronic safety related systems.

The standard IEC 61508 defines safety as "the absence of unacceptable risk of physical injury or damage to the health of people […]" [6]. It standardizes a life cycle model for creating a safety related systems. It specifies requirements for every stage of the life of a system to avoid systematic failures and to handle stochastic failures. It guides the developer through the pre-design phase, the design and installation phase, and the operation phase of the system.

Safety related systems are developed to reduce the inherit risk of the equipment under control (EUC) below the maximum tolerable risk by applying a variety of measures. The EUC, for example, corresponds with the building automation and control system.

The amount and kind of measures are always specified on account of hazards and its associated risks. As a result developing a safety related systems always requires a hazard and risk analysis of the EUC. It consists of a specification of hazards causing a dangerous situation, a description of the reason of the hazards and an identification of risks associated with the different hazards.

Safety requirements that describe how to handle hazards in a safe way are derived from the hazard and risk analysis. Safety requirements define the behavior of the safety functions performed by the safety related system.

Beside safety requirements there are also safety integrity requirements, i.e. performance requirements for the safety functions, necessary to be defined in order to achieve functional safety with a safety related system.

Table 1. Safety integrity level (IEC 61508)

Safety integrity level (SIL)	High demand or continuous mode (Error probability per hour)
4	$\geq 10^{-9}$ to $< 10^{-8}$
3	$\geq 10^{-8}$ to $< 10^{-7}$
2	$\geq 10^{-7}$ to $< 10^{-6}$
1	$\geq 10^{-6}$ to $< 10^{-5}$

Table 2. Safety integrity of deployed hardware (IEC 61508)

Safe failure fraction	Hardware fault tolerance[1]		
	0	1	2
< 60%	not possible	SIL 1	SIL 2
60% - < 90%	SIL 1	SIL 2	SIL 3
90% - < 99%	SIL 2	SIL 3	SIL 4
≥ 99%	SIL 3	SIL 4	SIL 4

1) A hardware fault tolerance of N denotes that N+1 faults cause a loss of the safety status of the system.

Safety integrity requirements specify the possibility of a safety function being performed according to expectation. Safety integrity requirements are derived from the risk assessment where the risk of every hazard is determined. Risk can be decided either by means of qualitative or quantitative measures. Due to general uncertainties in determining the probability and the damage of hazards it is state of the art to use qualitative measuring such as a risk matrix [7] or a risk graph [6].

The performance of the safety functions is categorized by four safety integrity levels (SIL) defined in IEC 61508. Safety integrity level 1 (SIL 1) is the lowest and safety integrity 4 (SIL 4) is the highest level. Each level corresponds with a specific error probability per hour (see Table 1). The value of the error probability specifies the probability of a dangerous error per hour.

On account of the safety integrity level the likelihood for successfully performing the safety functions is defined. The lower the likelihood of dangerous failure the higher the performance of the safety functions must be and the more thorough are the safety integrity requirements.

After specifying the safety functions and the safety integrity level, designing a safety related systems additionally requires a consideration of the deployed hardware where the safety functions are executed. A defined safety integrity level of safety functions can only be reached by increasing the hardware fault tolerance (see Table 2 for an explanation) or the safe failure fraction. The safe failure fraction (SFF) specifies the quantity of failures that do not result in a dangerous situation. The standard IEC 61508 presents a couple of ways to reach a safety integrity level.

Safe failure fraction can be augmented by detecting failures with a high probability. These detected failures are handled by the safety related system properly. Another way is to deploy highly reliable hardware where per se failures occur with a very low probability.

An alternative way to reach a defined safety integrity level is to increase the hardware fault tolerance. That is, additional measures are taken (such as the use of redundant hardware) to avoid a dangerous situation although a hazard has occurred.

3. Security – Common Criteria

In 1993 the CC (Common criteria) project was started to harmonize US, Canadian and European security criteria and create a single set of IT security criteria. After some draft versions were published and extensive reviews were made, CC version 2.0 was finally standardized as ISO/IEC 15408 [9] in 1999. The standard (for historical purpose called CC) is a basis for evaluation of security properties of IT products and systems. CC specifies a set of requirements for the security functions of IT products and systems. Additionally, it gives requirements for assurance measures applied to the security functions during security evaluation. As a consequence CC permits to compare results of independent security evaluations.

Within this paper security is defined the following: "Security is concerned with the protection of assets from threats, where threats are categorized as the potential for abuse of protected assets" [9]. Assets are described as information or resources to be protected by security countermeasures. Security especially pays attention to those threats related to malicious or other intentional activities.

CC includes two basic concepts: a security concept and an evaluation concept. The idea of the first one is that owners of assets analyze the possible threats to the assets. They determine which threats apply to their environment. These threats result in risk to the assets. To reduce the risk to assets, countermeasures are required that themselves may posses vulnerabilities and lead to a risk to the assets.

The evaluation concept is based on the idea that evaluation gives evidence of assurance and assurance techniques produce assurance. Owner of assets require assurance because it gives confidence that countermeasures minimize risk to their assets.

The standard presents a framework in which an effective evaluation is possible by defining a way to derive requirements and a specification of the TOE (Target of evaluation; IT product or system that is subject of evaluation). It, however, does not mandate any life cycle model. To receive a TOE specification, four major stages must be run through.

1. Establish security environment: Investigate the physical environment, assets requiring protection, purpose of the TOE.
2. Establish security objectives: Identify assumptions, threats and security policies.
3. Establish security requirements: Derive requirements from the security objectives by means of the CC requirements catalogue.
4. Establish TOE summary specification: Functional and assurance requirements lead to the TOE summary specification.

Table 3. Functional Security Classes [9]

Security audit	Privacy
Communications	Trusted path
Cryptographic support	Resource utilization
User data protection	TOE access
Identification and authentication	Protection of the trusted security functions
Security management	

Table 4. Security Assurance Classes [9]

Configuration management	Tests
Delivery and operation	Vulnerability assessment
Development	Evaluation criteria
Guidance documents	Assurance maintenance
Life cycle support	

Security function and security assurance requirements are specified in the CC catalogue.

- Security requirements describe the security behavior of a TOE.
- Assurance requirements define the scope, depth and rigor of evaluation of a TOE.

Both, security and assurance requirements, are categorized in classes. Security requirements of a class share a common focus. The name of an assurance class indicates the covered topics. Each class consists of different families of security requirements which share same security objectives. Families are finally divided into components that are the smallest set of requirements (Figure 1).

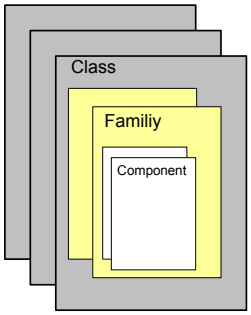

Figure 1. Class, family, component hierarchy

Security requirements from the different functional security classes (Table 3) are chosen depending on the security objectives. Security assurance requirements are selected from various assurance classes (Table 4) regarding the evaluation assurance level (EAL).

The philosophy of CC is to grant assurance based on an evaluation, i.e. active investigation of the IT product or system that is to be trusted. For that reason seven evaluation assurance levels (EAL) are specified that provide a uniformly increasing scale. A higher EAL reduces the likelihood of vulnerabilities and increase the

amount of confidence, but the effort is getting greater because a larger portion of the system is included in the evaluation process. In addition, more details of the design are covered and the evaluation process is carried out in a more structured and formal manner the higher the EAL is required.

4. Integrated Pre-design of Safe and Secure BACS

Section 2 and section 3 outlined the basics specified independently in safety and security standards of getting requirements for a system to develop. Whereas the safety standard specifies a life cycle model, the security standard only describes a way of deriving requirements and specifications. Both safety and security standards define levels to categorize the system. Safety integrity level (SIL) specify the level of performance of safety functions. Evaluation assurance level (EAL) give information about the level of security evaluation.

The integrated pre-design of safe and secure BACS is the first phase of a safe and secure life cycle model. The idea proposed by the authors and in standardization of BACS (CEN/TC 247) is to use the safety life cycle from IEC 61508 and integrated the way of deriving security requirements from CC. Moreover activities are added to consider safety and security dependences.

A life cycle model is a general model of the life of a system, including all the activities needed to develop, maintain and dispose a system. The advantage of such a life cycle model is the structured and formal way of development and maintenance.

This paper focuses on the pre-design phase of the safety and security life cycle (see Figure 2). The pre-design phase summarizes activities of the first phase of the life cycle. Step 1 to 4 is following IEC 61508 [7], steps 5 to 8 are following the Common Criteria [9]. Note that the arrows in Figure 2 do not intend to symbolize a sequential development process such as the waterfall model [10] or V-model [11] does. They imply that activities of step *n* requires input from the preceding step *n-1*. The pre-design phase can/must be repeated in an iterative process to finally receive a complete set of safety/security requirements for the BACS.

The pre-design phase begins with “definition of the concept” and “safety scope definition”. During that steps the physical environment of the BACS, field of application and relevant laws are examined as well as typical hazards in BACS are identified. Next the scope of the BACS and the scope of the hazard and risk analysis is specified.

“Hazard and risk analysis” aims at identifying typical hazards in BACS. Additionally, it describes the reasons for the hazards and determines the risk associated with the hazards. In the following the safety requirement specification, used to define the safety related system, is

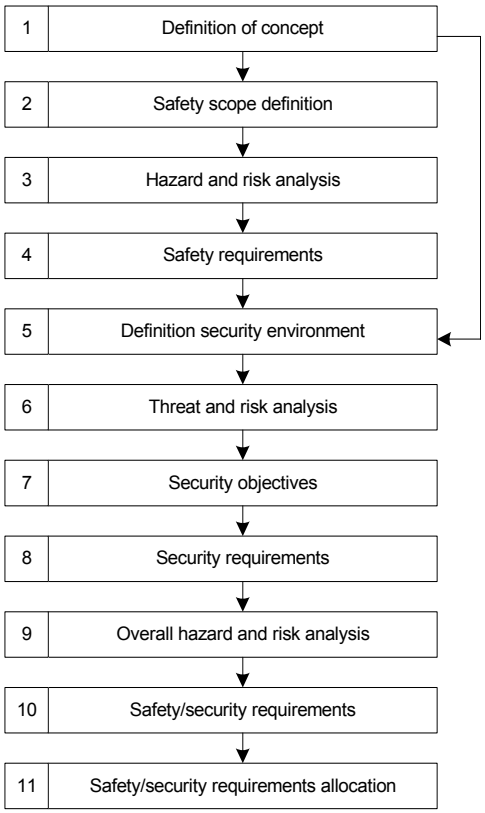

Figure 2. Pre-design of a safe and secure BACS

derived from the hazard and risk analysis.

The next step comprises the list of assets requiring protection, already including the safety related system functionality, the purpose of the target of evaluation (TOE) and the field of application. Next security objectives are derived from a list of threats and the associated risk to the BACS itself and the safety related system. Moreover security policies are investigated. The desired evaluation assurance level (EAL) is defined. Security objectives and the CC requirements catalogue are used to specify the security requirements, functional and assurance ones according to the EAL.

Step 9 “overall hazard and risk analysis” investigates safety and security requirements. It is checked whether security requirements lead to new hazards and risks to human health. I.e., are there new safety requirements necessary due to security needs and how they influence security.

In the end the common, overall safety/security requirements are specified. At this point the commonalities and contradictions between safety and security requirements are evaluated. In case of contradicting requirements safety is favored over security or vice versa depending on the field of application. Finally, requirements are allocated to the safety/security functions as well as the level of performance regarding safety (SIL) is chosen.

5. Discussion

The developed approach to integrated safety and security in building automation and control systems (BACS) is a life cycle model trying to harmonize the safety and security discipline. Based on well approved standards for safety and security the common concepts and methods (see section 2 and 3) are integrated.

Both, safety and security disciplines, deal with the problem of risk reduction. However, their objectives differ. Safety measures try to protect people, security measures aim at protecting resources or information. Risk reduction is achieved by safety and security functions respectively. They are derived from the respective requirements. IEC 61508 on the safety side and Common Criteria (CC) on the security side specify a big number of requirements to receive safety or security requirements by using the same concept(s) of requirements derivation.

Moreover, the standards define levels (safety integrity and evaluation assurance) to allow for a comparison of different system designs. Safety integrity levels are defined by different numbers of error probability per hour (Table 1). Evaluation assurance levels specify a set of requirements from the different assurance classes (Table 4). Although the measurands are different, both levels, however, have in common that the higher the level the more and stricter requirements must be met. A higher level results in a higher risk reduction. I.e., safety and security have the strikingly similar goals.

For that reason approaches to unify safety and security have already been published. In [15] an unified security/safety risk framework is presented. It combines the existing security and safety risk model. It specifies that the output of the existing security model, the "potential for harmful security event", should be considered as a safety hazard. The common approach presented in this paper, however, starts with activities relating to safety and then analysis security.

This procedure was chosen because:

1. First of all because IEC 61508 specifies a very formal and strict way of receiving requirements. The IEC 61508 procedure is very well adopted and often a legal requirement
2. Depending on the safety integrity level a hardware fault tolerance different from zero is required (see Table 2). This requirement results in a specific physical system architecture (e.g., "two channel architecture") that must be considered while establishing the security environment.
3. Moreover assets to be protected included safety requirements might be comprising the security environment. In most BACS safety is a key application functionality that cannot be readjusted.
4. [20] suggests to integrate security measures as monitoring functions that indicate failures in the safety system. This will limit security functions to a passive subordinate role. In proposed approach security is an active part of the system.

After step 8 of the pre-design phase (Figure 2) safety requirements were specified that are part of the security environment. Additionally, a set of security requirements is available already considering safety requirements. What is still missing at this point is a cross-checking of both sets of requirements. Are there new hazards and are new risks imposed on the BACS due to security? Are the safety and security requirements complementary? Do security requirements contradict safety requirements and vice versa? Finally, what functions are necessary to meet safety and security requirements?

The "overall hazard and risk analysis" uses the hazards and threats already identified in step 3 and step 6 of the pre-design phase respectively. At this point attention is paid to the dependencies of safety and security. First security requirements are checked if they cause not yet identified hazards. If so, another cycle need to be started. New safety requirements are specified and step 5 to 8 in Figure 2 are repeated. In case the "overall hazard and risk analysis" reveals the same hazard due to security requirements at the end of an additional cycle, a clear contradiction between safety and security was identified. Next risk of the hazards and threats are evaluated with regard to the safe and secure BACS. At this stage of the pre-design phase a possibility is given to set the focus of the BACS either on safety or security. Risk allocation depends on the field of application: either safety or security has priority but in many cases safety is dominant.

Regarding safety and security requirements and hazards coming from the overall hazard and risk analysis, so called safety/security requirements are specified. Most of them are equal to the safety and the security requirements they are based on. In case some requirements are contradicting each other, the presented approach foresees that the one that impose a higher level of risk will be selected. This kind of methodology presents a clear and concise way of solving the problem of contradiction between safety and security requirements. To show a way how to deal with contradictions, the following example is given.

In the safety world it is common practice to send "alive messages", so-called heartbeats, between a producer (actuator) and a consumer (sensor). They are sent periodically according to the chosen SIL (Table 1) to check whether the consumer is still running. In addition, the heartbeats must be generated within a defined time frame on a node. Since a consumer should just accept heartbeats from particular defined producers, authentication using a message authentication code (MAC) is applied. Let's assume that generating a heartbeat must be performed within 30 ms due to the chosen SIL; moreover generating a 16-byte MAC

Table 5. Typical reasons for network hazards in BACS [13]

Cross talk	Aging
Broken cable	EMC Failure
Wiring failure	Human failure
Stochastic failure	Temperature
Stuck at failure	Transmission of non-authorized messages

requires 50 ms of time.

If heartbeats are sent in a fire alarm system in office buildings between a fire detector and fire damper, the following problem has to be solved: generating an authenticated heartbeat within 30 ms is impossible because processing a 16-byte MAC takes 50 ms. As in this scenario the risk of sending heartbeats less frequently is much higher (Risk of not detecting a defect sensor imposes a high risk to people in the building.), the safety requirement is preferred.

In case of an access control system to a vault the situation is different. Heartbeats are sent between an actuator to open door and the input screen. If we assume that the door of the vault can be opened manually from the inside – low risk level to people being inside the vault – , authentication (generating a MAC) has priority to avoid unauthorized access to the actuator.

The final step of the pre-design phase in Figure 2 specifies functions derived from the requirements. Methods from the safety or security world that form the functions are applied depending on the prioritization of safety or security.

The following section presents the usage of the integrated pre-design model. Special attention is paid to dependencies between safety and security. As an example for this, the usage of source addresses is taken.

6. Practical Application

Applications for safety use source addresses to prevent message insertion. Security access control can be based on entity identification by source addresses.

According to the developed pre-design model (Figure 2) steps 3 and 4 as well as steps 6 and 8 are the core parts of the joint analysis. Although based on general requirements, the definition of safety scope and security objectives needs to address specific issues of both areas. The different focal position in the pre-design model mainly stem from historical reasons. The security approach is to first check what harm can be done and then define the security measures. In safety a certain requirement is usually given in advance. In practical usage, both cases will have a (preliminary) definition of scope and objectives before the hazard/threat and risk analysis and a reconsideration or definition after this analysis.

Table 7. Typical reasons for network threads in BACS [18]

Trojan horse	Data forgery
Eavesdropping on the net	Address Spoofing
Flooding machines with bogus data	Human failure
Isolating machines by DNS attacks	Impersonation of illegitimate users
Viruses	Transmission of non-authorized messages

Table 5 and Table 7 show typical reasons for hazards and threats for field level communication systems that effects the integrity of source addresses in a message. Data for this analysis have been take from the projects SafetyLON [16] and REMPLI [17].

At this stage little commonalities can be identified. Although similar tools (e.g. methods to identify risk with a risk matrix) and approaches for analysis are taken each area is investigated on its own.

Table 6 and Table 8 show the effects of hazards and threats. At this stage of the process the commonalities can already be identified. E.g., corruption of a message can stem from a stochastic failure, but also from an intentional manipulation by an attacker.

If safety and security measures are used jointly and not installed in parallel, a potential for synergies can be acquired. The measures and synergies gained can be classified in three groups:

1. There are measures that *directly match* such as time stamps or sequence numbers for delayed or repeated messages. Usually there will be no problem to commonly use them. High potential for synergies exists since measures are easily combinable.
2. There are measures that require *different efforts*, e.g. in terms of computational power or consumed bandwidth, such as CRC (Cyclic Redundancy Check) or MAC (Message Authentication Code).

Table 6. Effect of network hazards and resulting safety requirements [8]

Hazard	Safety requirements
Corruption of data	CRC, duplication of message
Loss of a message	Use of a watchdog
Insertion of a message	Use of safe source addresses
Repetition of a message	Use of a time stamp
Wrong sequence of messages	Use of a time stamp
Delay of a message	Use of a time stamp
Non safety related message	Use of a specific header, safe source addresses

Table 8. Effect of network threats and resulting security requirements

Hazard	Security requirements
Modification of data	MAC, Signature
Loss of a message	protocol timeout
Insertion of a message	sequence number (protected against modification)
Replay of a message	sequence number/time stamp
Wrong sequence of messages	sequence number
Eavesdropping	Message encryption
Spoofing of source address	Inclusion of source address in MAC or signature

Both of these measures protect the integrity of the message, but the execution time (e.g., 10-100µs for CRC and 8-15ms for MAC) and bandwidth (e.g., length 2 byte for CRC and 16 byte for MAC) differs. Gains need to be judged on application.

3. There are measures that are unique for safety and security (such as a watchdog timer) and needs to be implemented separately. No synergies possible.

E.g., in a safety related system a source-addressing model is used to guarantee message exchange between safe nodes only and to avoid message insertion. Therefore each safe node is assigned an additional unique address, a so called *safe address*. The receiving nodes check this safe address against their access list and only allows reception of packets in the list. The safety is given by a CRC checking and the transfer of a safe address within the safe message that can only be generated by safe nodes.

In a secure system similar techniques are used. An access control is also based on the node address [17], but instead of the CRC a cryptographic message authentication code (MAC) is used that cannot be recalculated without the knowledge of the appropriate key.

A matching of the requirements and measures can lead to synergies in the design of an integrated safe and secure system. In our example the CRC is replaced by the MAC which allows to remove the safe address. Access control is now managed by normal addresses and the requirement to identify nodes belonging to the safe group is realized by a particular key only available to nodes in this (safe) group. Timestamps and replay counters are unchanged since they have equal tasks in both areas. In this case also the overall hazard and risk analysis will indicate no new risks, since all requirements are covered by the new solution.

In general, security measures will replace safety measures since measures designed for safety do not withstand intentional attacks. E.g., a CRC protects the integrity of a message, but can be recalculated online. Hence stochastic failures are discovered, but an attacker is not impeded to manipulate the targeted information (asset) as well as the CRC.

Another important issue to consider is the resource consumption of the applied measures. Introducing safety and security measures will increase the overhead in computational resources as well as network bandwidth consumption to achieve. Selection and trade-off of different measures inside the fields of safety and security is set out of scope for this paper. E.g., if a CRC or a message duplication is used will not be analyses since this is included in the analysis of the individual security requirements and often demanded by normative regulations.

In particular the adding of security measures to a safe system should be analysed since this is a common case: Table 9 shows the overhead of typical security measures used in embedded systems. If there are no synergies this overhead is directly added and can even double, e.g., if messages are duplicated. In case of synergies such as the replacement of the CRC (typically 2, 4, or 8 byte) the network overhead is 6, 4 and 0 byte for the 8 byte MAC. Other typical MAC functions have a length of 16, 20 or 32 bytes. Concerning the computational power security measures will usually show no synergies since other types of checksums are much faster (factor 100) and/or can be implemented with a negligible effort in hard- or software.

7. Conclusion

The possibilities to gain synergies by taking an integrated approach towards safety and security in BACS is given in many areas. This fact is well known, e.g., “Safety and security […] are closely related, and their similarities can be used to the advantage of both in terms of borrowing effective techniques from each to deal with the other.” [12]. Yet little effort to combine these fields is given since applications are usually either safety or security.

Table 9. Performance of key derivation and security functions [17]

Description	Time [ms]
Derive key by 3-DES function	89
Cipher message using 3-DES in outer CBC mode stored in EEPROM/RAM	63/58
Decipher message using 3-DES in outer CBC mode stored in EEPROM/RAM	54/55
Authenticate message with 8-byte MAC using 3-DES in outer CBC mode in EEPROM/RAM	57/53
Verify 8-byte MAC using 3-DES in outer CBC mode stored in EEPROM/RAM	45/47

In building automation this situation changes since mainly for cost reasons a combination of formerly separated networks for safety, e.g., fire alarm system, security, e.g., access control, and operation, e.g. heating, ventilation and air condition, is desired. Combination on the one hand demands for a reliable system and also increase the need for security since systems formerly physically separated are now accessible for a bigger group of users.

This article proposed a common approach for the pre-design phase of such integrated safety and security systems. Techniques such as the risk analysis common in both areas are synchronized to figure out overall hazards that endanger safety or security of a BACS. Since various hazards even put in danger both safety and security of the system, a dual usage of counter measures seem feasible. The life cycle model presented specifies requirements for the different stages in development of a BACS. At the moment the work done focuses on the pre-design phase and the network communication. Inclusion of hardware integrity and the following stages of the life cycle are important topics for further research.

In the area of building automation a certain convergence of systems to safe and secure systems can be noticed, but finally it must be stated that the common approach will only show benefits when both security and safety functions are required by the application. Convergence seem more likely for applications with high security requirements since in this case the overhead given by security is not a hindrance rather a requirement. Moreover contradictions between both areas cannot be completely avoided. The final decision if security or safety is to be preferred within such conflicts is application and environment dependent.

References

[1] C. Schwaiger, A.Treytl, "Smart Card Based Security for Fieldbus Systems", *Proceedings of IEEE International Workshop on Factory Communication Systems*, Vol. 1, pp. 398-406, 2003.

[2] D. P. Eames, J. Moffett, "The Integration of Safety and Security Requirements", *SAFECOMP'99, LNCS 1698*, Springer-Verlag, Berlin, Heidelberg, pp. 468-480, 1999.

[3] V. Stavridou, B. Dutertre, "From Security to Safety and Back", *Proceedings of Computer Security, Dependability and Assurance: From needs to Solutions*, pp. 182-195, 1998.

[4] A. Simpson, J. Woodcock, J. Davies, "Safety through Security", *Proceedings of the 9th International Workshop on Software Specification and Design*, pp. 18-24, 1998.

[5] A. Avizienis, J-C. Laprie, B. Randel, "Fundamental Concepts of Dependability", 2001.

[6] G. Dewsbury, I. Sommerville, K. Clarke, M. Rouncefield, "A Dependability Model for Domestic Systems", *SAFECOMP 2003, LNCS 2788*, Springer Verlag, Berlin, Heidelberg, pp. 103-115, 2003.

[7] "IEC 61508 – Functional safety of electric/electronic/programmable electronic safety-related systems", 1999.

[8] "EN 50126 – Railway applications. The specification and demonstration of reliability, maintainability and safety (RAMS)", 1999.

[9] "IEC 15408 – Information technology – Security technique – Evaluation criteria for IT security", 1999.

[10] W. Royce, "Managing the Development in Large Software Systems", *Proceedings of IEEE WESCOM*, 1970.

[11] G. Müller-Ettrich, *Objektorientierte Prozessmodelle: UML einsehen mit OOTC, V-Modell, Objectory*, Addision-Wesley, 1999.

[12] N.G. Leveson, *Safeware: System Safety and Computers*, Addison-Wesley, 1995.

[13] "EN 50159-1: Railway Applications – Safety-Related Communication in Closed Transmission Systems", 2001

[14] D. Reinert, M. Schaefer (Publisher), *Sichere Bussysteme in der Automation*, Hüthig Verlag, Heidelberg, ch. 4, 2001.

[15] G. Stoneburner, "Toward a Unified Security-Safety Model", *IEEE Computer*, Vol. 39, pp. 96-97, 2006.

[16] T. Novak, T. Tamandl, "Architecture of a Safe Node for a Fieldbus System", *Proceedings of the 5th IEEE International Conference on Industrial Informatics*, Vol. 1, pp. 101-106, 2007 .

[17] A. Treytl, T. Novak, "Practical Issues on Key Distribution in Power Line Networks", *Proceedings of the 10th IEEE International Conference on Emerging Technologies and Factory Automation Proceedings*, Vol. 2, pp. 83-90, 2005.

[18] W. Stallings, *Cryptography and Network Security*, Prentice Hall, 2003.

[19] W. Granzer, W. Kastner, G. Neugschwandtner, F. Praus. "Security in Networked Building Automation Systems", *Proceedings of IEEE International Workshop on Factory Communication Systems*, pp. 283-292, 2006.

[20] K. Sørby, "Relationship between security and safety in a security-safety critical system: Safety consequences of security threats", *M.S. thesis*, Norwegian University if Science and Technology (NTNU), Department of Computer and Information Science, Norway, Trondheim, ch. 9, 2003.