DYNAMIC VALIDITY PERIOD CALCULATION OF DIGITAL CERTIFICATES BASED ON AGGREGATED SECURITY ASSESSMENT

Alexander Beck, Jens Graupmann, Frank Ortmeier

(alexander.beck1@volkswagen.de, jens.graupmann@djg-consulting.com, frank.ortmeier@ovgu.de)
Computer Systems in Engineering, Otto-von-Guericke-University, Magdeburg, Germany

ABSTRACT

The paper proposes a method based on different security-related factors to dynamically calculate the validity period of digital certificates. Currently validity periods are most often defined statically without scientific justification. This approach is not sufficient to objectively consider the actual need for security. Therefore the approach proposed in this paper considers relevant security criteria in order to calculate a meaningful validity period for digital certificates. This kind of security assessment can be executed periodically in order to dynamically respond to changing conditions. Especially in the context of complex systems and infrastructures that have an increased need for security, privacy and availability this issue is highly relevant.

KEYWORDS

digital certificates, validity period, crypto period, security assessment, authentication, risk assessment, security engineering, security metrics and measurement

1. INTRODUCTION

An important component for cryptographic protection of IT infrastructures in large companies is digital certificates. Certificates can be used to secure communication channels, mutual authentication of IT-systems (or people) and also for digital signatures. A common property of digital certificates is their predefined validity period. Assuming that at the time of the creation of a certificate current cryptographic methods have been used, however, the algorithms may be broken before the expiry of the certificate validity period. This validity period is relatively often defined by companies to four or five years.

A similar loss of trust regarding the certificate occurs if the secret key of the certificate is compromised due to other circumstances. The resulting need for a dynamic validity period calculation of digital certificates based on relevant risk factors is an integral part of this paper. After the definition and analysis of this topic a calculation method is presented based on several security related criteria in order to determine the optimal validity period.

Finally it is shown how such a process can be integrated into a certificate lifecycle management system. Certificate lifecycle management system denotes a system that manages and monitors digital certificates starting from their creation and delivery up to their expiration or revocation – comparable to other lifecycle management systems.

DOI:10.5121/ijcis.2012.2401

2. RELATED WORK

Numerous works deal with the topic of PKI and all related topics. Of particular interest are the critical works such as “The Inconvenient Truth About Web Certificates” [5] and “Ten Risks of PKI” [6], dealing with actual rather practical problems in the context of a PKI that have to be taken into account. In the context of this paper approaches that attempt to quantify security are particularly relevant. This category includes the works "A calculus of trust and its application to PKI and identity management" [7] and "Trust-rated Authentication." [8].A good overview of PKI-related implementation issues provides “NIST Special Publication 800-63 Problem” [9].

3. CHOOSING THE RIGHT CRYPTO PERIOD

The security level of IT systems should always be in relation to their actual threat. The current threat to an IT system does not only rely on the system vulnerability but also, for example, on the interest in the system for unauthorized persons. The resulting (composite) threat level can be very dynamic and can change over time frequently.

For this reason, not only a dynamic initial calculation but also a periodic reassessment is necessary. When such kind of assessment is performed certificates are often not taken into account in detail. Certificates are for pragmatic reasons usually created for a fixed term. This term, the validity period, is usually chosen quite arbitrarily and is generally only adapted according to the path length from root certificate to issued certificate.

That means the lifetime of a root certificate is much longer than of an issued SSL certificate for example. The reason for this is that the lifetime of an issuing certificate should never end before the lifetime of an issued certificate.

Since the relationship between the validity period and the actual threat potential is generally not considered when issuing a certificate, a dynamic computation will be proposed in this paper. Normally, certificates will not be replaced before their expiry.

A good example of changing security assessments over time are some popular hash functions. Here, too, cryptographic algorithms are used as a basis , which can be broken in the course of time. A famous example is the MD5 hash function, which was published in 1991 by Ron Rives t[4]. MD5 was also introduced as a improvement of the MD4 hash function published in 1990 by Rivest to compensate the weaknesses of the MD4 algorithm. The basic problem of a128-bit long message digest was not solved with that improved algorithm. So it is not surprising that on 17 August 2004 Wang, Feng, Lao and Yu showed collisions for MD5 hashes [4].

As a consequence recommendations to stop using MD5 in IT systems were issued [11]. This famous example of a very popular cryptographic function shows how short-lived cryptographic algorithms can be that serve as the basis for issued certificates. In this case, MD5 has been used for 13 years in diverse IT systems.

Because many CA certificates had been issued with a lifetime longer than those 13 years they were still actively used at the time the algorithm was broken.

As a consequence all valid certificates issued by these CA certificates had to be revoked and reissued in order to ensure an adequate level of security.

As the number of used certificates continuously grows the topic discussed here will gain more and more relevance in the near future.

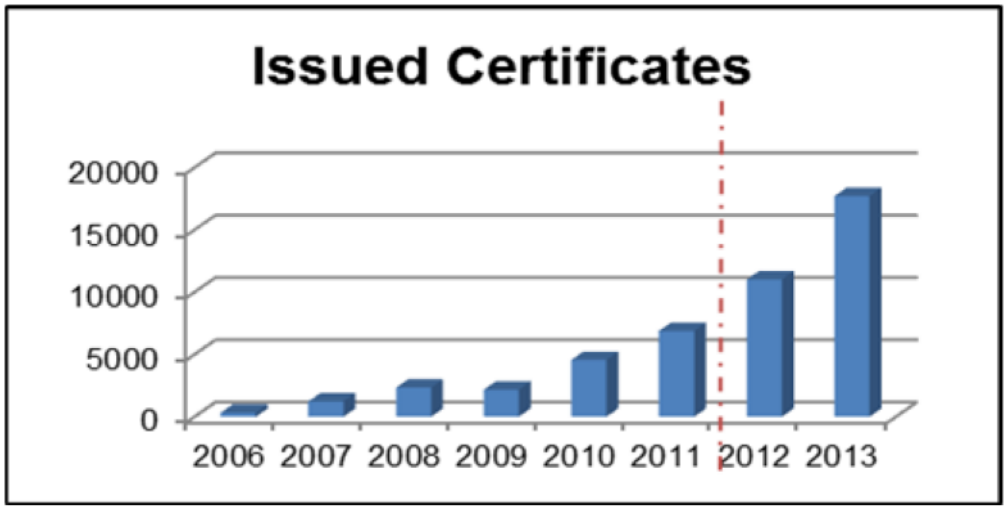

Figure 1. Number of certificates of a leading automotive company

The figure above shows the number of issued certificates and the expected increase in the next years of a leading automotive company.

4. CALCULATION BASED ON AGGREGATED SECURITY RISK ASSESSMENT

In order to determine an appropriate certificate lifetime considering the system security a number of factors should be taken into account. Quantification and aggregation of these factors provides a measure that can be leveraged to derive a certificate lifetime in a meaningful way.

A specific calculation always has to be adopted with regard to the actual PKI context. Therefore no "universal" calculation rule can be provided. The following example calculation outlines a typical calculation based on common factors. Such a calculation can become quite complex because usually some factors affect other factors. The resulting value of each factor should lie in the interval [0; 1]. In the aggregation formula each factor should be weighted in accordance to its significance with regard to the actual PKI context that is considered. Although this kind of normalization and weighting is technically not necessary it makes things easier and more transparent.

4.1. Vulnerability of IT-Systems

The calculation of the vulnerability of an IT system is an estimate based on recognized factors. It can be determined in various ways ranging from simple automated computations to the individual manual assessments of each single system and component. Below some basics of manual assessments and a new concept for a pragmatic automated process is discussed.

4.1.1. Manual Security Assessment

Manual security assessment is often based on the following modules

- General Security Audit

The purpose of an audit is a documented status of the lived level of IT security, the detection of defects and security vulnerabilities as well as the assessment by an

independent auditor of existing security measures in technical and organizational areas.

- IT System Audit

Considered aspects:
  - operating system hardening
  - software versions
    - authorization (roles & permission) and passwords
  - safety related configurations

- Vulnerability Scanning

Considered aspects:
  - Open ports
  - Used services
  - Installed operating system and software
  - versions
  - Lack of security updates (patches)
  - Shared volumes
  - Password and user policies

- Penetration test

  A penetration test is a test in which a penetration tester tries with appropriate programs or methods to penetrate a system and to exploit vulnerabilities that were identified.

The resulting value should be between 0 and 1 (system is absolutely safe). The above list should exemplary show the possible extent of such assessment. The actual components and weights have to be aligned to the current processes and policies of the PKI-operating company.

4.1.2. Automated Security Assessment

In this section an automated calculation of system vulnerability based on a Configuration Management database (CMDB) and a safety assessment database is presented. The CMDB manages all resources and their dependencies. This includes hardware and software including their exact versions and patch levels.

The automated calculation of vulnerability must take into account all individual components and should provide a value that reflects the security level for the overall system.

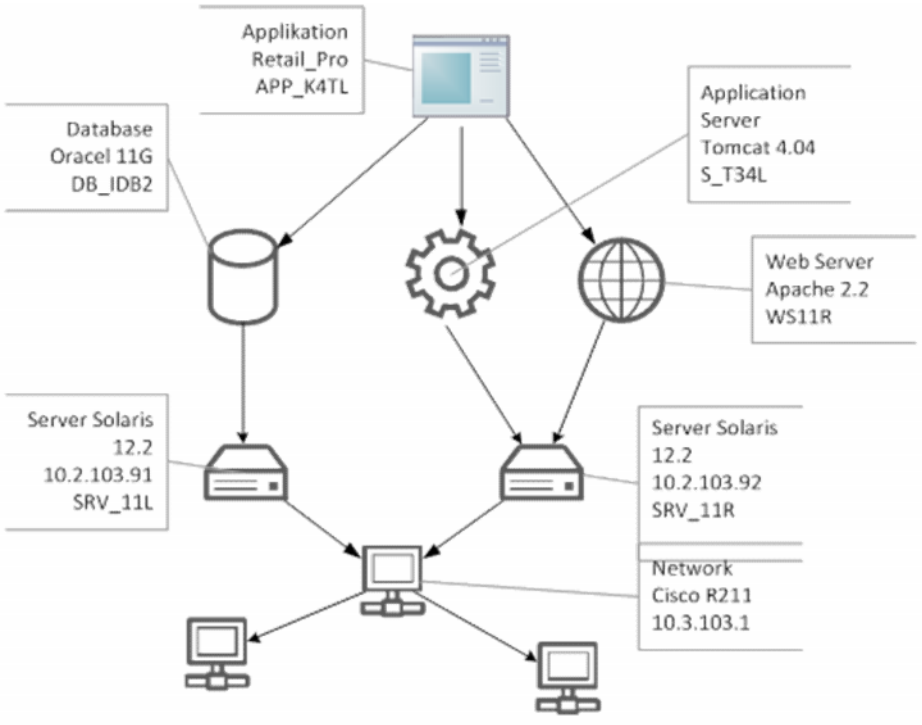

Figure 2. Configuration Management Database (CMDB)

Figure 2 shows a typical section of a CMDB. Here each node corresponds to a system or software component. For each component a corresponding security rating value must be defined in the security assessment database.

This requires a regular assessment of all the resources under consideration by appropriate qualified personnel based on accredited sources. In order to provide a comprehensive security rating all software and hardware components directly connected to the considered resource have to be considered.

The calculation formula has to fulfill the following conditions:

- The resulting value must lie in the interval [0; 1] (1 means: system is completely safe)
- The aggregated value must be less than or equal to the smallest single value.

A simple calculation which meets the above requirements is the multiplication of all individual values. Of course the factors can weighted using a factor in the interval [0; 1] according to their relevance.

Alternatively, however, the lowest safety factor can be chosen as the resulting overall value of the whole system – this calculation also meets the above requirements.

4.2. Criticality of data to be protected

The more valuable the data managed by a system is the more interest by potential attackers exists to obtain possession of this data. As more critical (or sensitive) data requires a higher level of protection, public data has to be rated with a value of 1 (that means: no interest by attackers) The

more critical the data is the lower its rating has to be. Existing data classification schemes can serve as reference points here.

4.3. Key length & algorithms

The key length in dependence of the algorithm is an essential aspect for the determination of a certificates lifetime. So it is not surprising, that the relationship between key size and security needs has been studied in many scientific publications. New, however, is the consideration of the key length in the context of an aggregation of a number of relevant factors.

In general, the longer the key length is, the longer the lifetime of a certificate can be chosen [3]. The influence of the key length on various aspects has been investigated by the NIST and defined as follows [2]:

A suitably defined crypto period:

1. Limits the amount of information protected by a given key that is available for cryptanalysis.
2. Limits the amount of exposure if a s ingle key is compromised
3. Limits the us e of a particular algorithm to its estimated effective lifetime
4. Limits the time available for attempts to penetrate physical, procedural, and logical access mechanisms that protect a key from unauthorized disclosure
5. Limits the period within which information may be compromised by inadvertent disclosure of keying material to unauthorized entities
6. Limits the time available for computationally intensive cryptanalytic attacks (in applications where long-term key protection is not required)

In order to choose the appropriate key length many aspects and different methods have to be taken into account. A helpful interactive collection of different recommendations and regulation was created by Damien Giry [1]. It permits the comparison of different algorithms and key lengths in an interactive way. Based on such information that of course needs to be verified and updated on a regular basis , a corresponding fact table for all the algorithms and related key lengths is stored in the database.

Every meaningful combination of algorithm and key length must be assessed with a value between 0 and 1 with respect to safety. A value of near 0 implies that the algorithm is known to be broken and easily computable whereas a value of near 1 rates a combination that is considered to be safe for a long time.

4.4. Revocation status (CRL / OCSP)

Another factor to determine a suitable lifetime for certificates is related to the handling of revoked certificates. The revocation status can be checked using an OCSP service or certificate revocation lists (CRL).

As OSCP provides more timely information regarding the revocation status is has to be rated higher in comparison to CRLs in the context of our calculation. Based on the above calculation rules, this factor can be quantified trivially:

- usage of an OCSP service: 1
- usage of certificate revocation lists (CRL): 0.75
- no revocation checking: 0.5

4.5. Key storage of CA certificates and length of certificate chain

Usually certificates are not issued by a root CA, but by a Sub-CA. Depending on the size and structure of the PKI-operating company the path length from the root CA to the Sub -CA can significantly differ. The longer the path the more intermediate CAs can potentially be compromised. It is often the case that the safety level of a Sub-CA (or the corresponding organizational unit, respectively) is lower than that of each higher level.

For this reason both, the path length and the key storage of the most "uncertain" instance on the certificate path are of relevance. Since the actual security of Sub-CAs based on assessment of the infrastructure and key storage can hardly be rated, only the path length will be considered at this point. One possible (trivial) calculation is: 1 / path length.

4.6. Certificate Distribution (automated distribution, personal delivery, number of recipients)

The delivery process of a certificate and potentially of a corresponding passphrase is a non-negligible potential target for attackers.

Delivery: Automatic (SCEP or similar protocol)

Automated methods (SCEP, CMP), in which the certificate using resource generates the keys itself and issues a certificate request are considered to be safe as no critical key material is sent over insecure channels.

Delivery: Manual (e-mail)

Much more critical is the manual delivery of a particular certificate including the private key within a container (PKCS#12) via e-mail.

In case of manual (e-mail) delivery the following aspects should be considered:
If the certificate is delivered manually:

- Does the private key material retain by the applicant and is only the public part signed (PKCS#10)? Or is the private and the public key generated by the CA and delivered within a container (PKCS#12)?
- Is the email encrypted?
- Are the pin and the certificate container (PKCS#12) communicated through independent channels (e.g. phone and Email)?
- How large is the number of recipients of the certificate (e.g. size of used email distributing list)?

These aspects can be rated using a checklist

4.7. Aggregation

The following table outlines a classification into mandatory and optional factors. Furthermore, the value range and the meaning of the respective upper and lower limits of the corresponding factors is given.

Table 1. Factors of an aggregated security assessment

Factor	Mandatory	Value Range	Boundary value
Vulnerability of the IT-Systems	X	0 – 1	No protection at all
			Maximum protection
Criticality	X	0 – 1	Top secret
			Public
Key length & algorithm	X	0 – 1	Can be computed easily
			Assumed to be secure for > 20 years
Revocation status checking		{0,5; 0,75; 1}	No checking
			OCSP
Length of certificate chain		0 – 1	Infinite length
			Issued by root
Key distribution		0 – 1	Unencrypted via email
			Securely encrypted on secure token.

The Security Risk Assessment uses the factors described above to perform the computation of an optimal certificate lifetime (validity period) from a security perspective. Based on this calculation certificates obtain individual lifetimes based on their specific application and security assessment. The following condition must be met for the calculated runtime:

$$Calculated\ Lifetime \leq Requested\ Lifetime \leq CA\ Lifetime \leq Policy$$

It is recommended to establish this kind of Security Risk Assessment within a Certificate lifecycle management system (CLM). The construction of a CLM will be described in a paper to appear in the context of the automotive industry. The automotive industry is a well suited application scenario due to its complexity, heterogeneity and speed of innovation.

The figure below depicts the integration of a Security Risk Assessment (SRA) into a CLM system In addition to the SRA component the security fact database (that has been presented before) is required that contains information on cryptographic algorithms, used software and hardware components. This security-facts-DB (SFDB) is maintained both internally as well as externally based on public information sources.

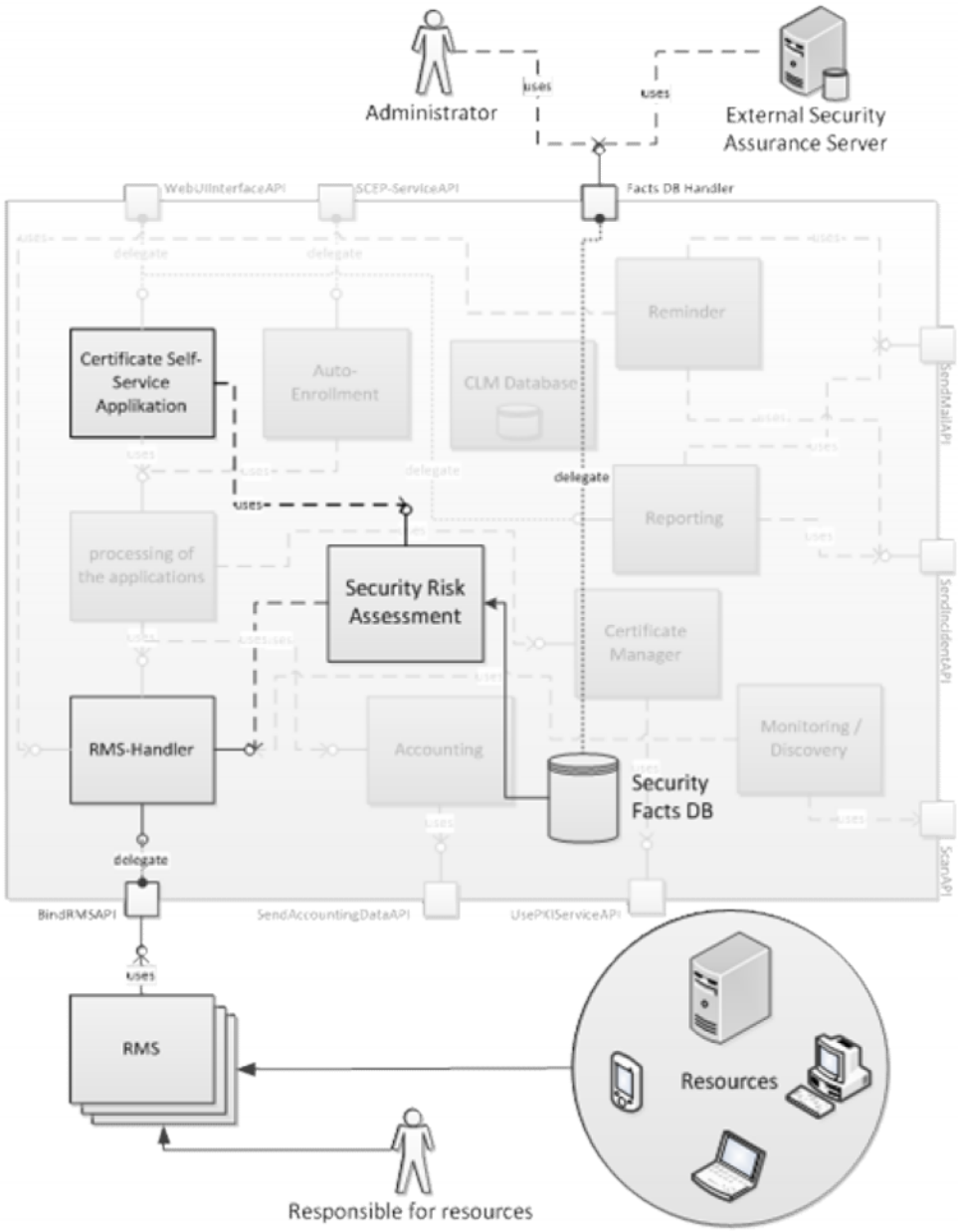

Figure 3. CLM-Architecture with Security Risk Assessment

The system can be maintained by specific administrative departments or other corporate departments that are concerned with security related issues, e.g. the security governance department.

The Security Risk Assessment uses the information located in the SFDB and in resource management systems (RMS) in order to assess the information contained in the certificate request received via the certificate request application. This assessment provides a certificate lifetime recommendation.

One important aspect that should not be ignored is the fact that computation of the certificate lifetime is based on the information available at creation time. As discussed this information can change very quickly.

For this reason it is useful to perform a periodic re- evaluation using approach presented above and thus respond timely to changing circumstances in particular in the area of algorithms and revised security assessments.

5. CONCLUSION

We have discussed that a fixed certificate lifetime does not correspond to the actual security requirements in a proper way.

In particular, the continuously improved computing performance that nowadays e.g. might be based one extremely large botnets or even “legal” cloud computing networks and an active community - partially legal and partially illegal - searching for security vulnerabilities can change original assumption regarding the security of applied algorithms very quickly. This is also true for vulnerabilities in software components.

Therefore, in this paper, an approach is presented to dynamically compute a proper certificate lifetime based on generally accepted factors and current security ratings. It was shown how this dynamic calculation can be embedded into a Certificate lifecycle management system.

Especially the combination of multiple security aspects in order to compute the certificate lifetime based on a holistic assessment of the certificate application environment has not been considered before. The approach presented in this paper will be evaluated later in the context of a large multinational automotive company.

REFERENCES

[1] Crypto keylength recommendation, www.keylength.com, Damien Giry, Okt. 2011
[2] Recommendation for Key Management, Special Publication 800-57 Part 1 Rev. 3, NIST, 05/2011
[3] Suitable algorithms to meet the requirements SigG Germany,May 2001 in conjunction with Appendix 1, Section I of Ordinance No. 2 of 16 November 2001, FNA, 20 May 2011
[4] Hashfunktionen gebrochen, Datenschutz und Datensicherheit, Rüdiger Weis , Stefan Lucks , 04/2005
[5] The Inconvenient Truth About WebCertificates, Nevena Vratonjic et al., The Workshop on Economics o f Information Security, 2011
[6] Ten Risks of PKI: What You're Not Being Told about Public Key, Ellis on, C. and Schneier, B., Computer Security Journal vol. 16, 2000
[7] A calculus of trust and its application to PKI and identity management, Huang, J. and Nicol, D., Proceedings of the 8th Symposium on Identity and Trust on the Internet, 2009
[8] trust-rated Authentication , R. Holz, H. Niedermayer, P. Hauck, G. Carle, Euro PKI, 2008
[9] Special Publication 800-63 Problem: Choose of the right crypto period, NIST, 2011
[10]NIST moves to stronger hashing, Olsen, F., Federal Computer Week, 2/7/2005

Authors short biography

Alexander Beck

Alexander Beck works at the Volkswagen AG in the information system security organization. Parallel to his work he does his Ph.D. at the Otto-von-Guericke University Magdeburg in the area of security technologies. Before that Mr. Beck did his B.Sc. at the University of Applied Sciences Harz from 2006 to 2010 and from 2010 to 2011 his M.Sc. at the Otto-von-Guericke University in cooperation with Volkswagen AG and ITConcepts Professional GmbH.

Jens Graupmann

Jens Graupmann received his PhD in Computer Science from the Max-Planck-Institut in 2006. His research focussed on search engines, database technologies and information systems. Since 2006 he has been working as consultant and project manager in different international consulting projects, primarily in the automotive and aerospace industry. In this context he extended his expertise to IT security.

Frank Ortmeier

Frank Ortmeier is a professor for Computer Systems in Engineering at the Otto-von-Guericke Universität in Magdeburg. He did his PhD at the University of Augsburg on "Model-based safety analysis" in 2005. His main research interests are in the domain of software challenges for technical applications. This ranges from analysis of safety critical applications to systems engineering of software intensive applications. In particular, he is researching on safety and security of cyber-physical systems.

Frank Ortmeier is responsible for a number European student exchange programs, member of several international program committees, an active member of the pre-standardization group EWICS, leading a number of larger research projects on critical systems and spokesman of the regional chapter of the German computer society (GI). He is also responsible for guidance of students in "Ingenieurinformatik" and "Digital Engineering" at the Otto-von-Guericke Universität.