

Enterprise Architecture Models for Cyber Security Analysis

Mathias Ekstedt and Teodor Sommestad

Industrial Information and Control Systems, KTH - Royal Institute of Technology, Sweden

Abstract-- Enterprise architecture is a rising discipline that is gaining increasing interest in both industry and academia. It pays attention to the fact that effective management of business and IT needs take a holistic view of the enterprise. Enterprise architecture is based on graphical models as a vehicle for system analysis, design, and communication. Enterprise architecture is also a potential support for control systems management. Unfortunately, when it comes to security analyses, the architectural languages available are not adapted to provide support for this. This presentation focus on research performed as part of the EU seventh framework program VIKING (Vital Infrastructure, Networks, Information and Control Systems Management) and the Swedish Centre of Excellence in Electric Power Engineering, EKC². The research is focusing on developing and adapting security analyses frameworks to architectural languages on a level where information about control systems' configuration is scarce and thus incomplete and partly unreliable.

Index Terms— Bayesian Networks, Control Systems, Cyber Security, Enterprise Architecture.

I. INTRODUCTION

SOCIETY is increasingly dependent on the proper functioning of the electric power system, which in turn supports most other critical infrastructures: water and sewage systems; telecommunications, internet and computing services; air traffic, railroads and other transportation. Many of these other infrastructures are able to operate without power for shorter periods of time, but larger power outages may be difficult and time consuming to restore. Such outages might thus lead to situations of non-functioning societies with devastating economical and humanitarian consequences.

The operation of the power system is today highly dependent on computerized control systems. These industrial control systems can be resembled to the central nerve system of the power system. At the same time as control systems enables more efficient, qualitative, and safe power systems, their vulnerabilities are also direct vulnerabilities of the power system itself.

In addition to the potential severe consequences of a compromised control system, security management of these

control systems is a really complex issue. First of all security is inherently suffering from a weakest-link syndrome. This means that in principle can just a single misconfiguration in the control system architecture be a vulnerability that jeopardizes the whole power system. This is truly a challenge since control systems are extremely complex: they contain highly advanced functionality; they are heterogeneous and include several third party components; they are extensively networked, both internally and with external systems, and they depend on the human organization that manages and uses them. Moreover, the security level of the control system depends on external factors. For instance, a protocol developed in-house by a vendor can perhaps be considered a relatively secure as long as few people know how it works. But as soon as the protocol specification pops up at Wikipedia or similar, the circumstances have changed; the exact same protocol has shifted from secure to insecure, potentially without its users knowledge. Altogether control system security management can be described as keeping track of a moving target that consists of a great number of details that are interrelated in very complex ways.

Now, how is control system security managed? Since the situation is so complex, it is completely impossible for a single person to keep everything in his or her head. The state-of-the-art management approach to system management in general is to adopt architecture models for systems and its surrounding organizational environment, i.e. enterprise architecture models. So is it for control systems. Perhaps counter-intuitive, but, since our systems are so complex we need simplified descriptions of our systems that provide a holistic view. We need this because without knowledge about the whole, knowledge about the details serves little purpose in a complex world. The challenge is however to make sure that our models contain relevant properties of the systems and organizations. For security oriented models it is for instance vital that information about firewall configurations and physical locations of equipment are included in the models, whereas we are perhaps not as interested in for instance source code length or user interface layout (that would be very relevant for other system properties). Due to the weakest-link syndrome, it is also essential that we are sure that our system models are correct with respect to how the system is really implemented. To develop such a security enterprise architecture model that is correct and relevantly adapted to its purpose is not easy, but if you have it control system security can be effectively managed.

Manuscript received October 24, 2008.

Mathias Ekstedt is with the Royal Institute of Technology, Osquidas väg 12, 100 44 Stockholm, Sweden, weden, phone: +46 87906867; fax: +46 8 7906839; e-mail: mathiase@ics.kth.se.

Teodor Sommestad is with the Royal Institute of Technology, Osquidas väg 12, 100 44 Stockholm, e-mail: teodors@ics.kth.se.

A. Outline

The next chapter mentions some of the important guides and standards in the field of security and cyber security and describes how they can be considered as assessment theories. The third chapter introduces enterprise architecture models. The fourth chapter describes how security theories are combined with architectural models to purpose-oriented architectural models. Finally the paper is concluded in the fifth chapter.

II. SECURITY THEORY

Within the field of security, a large number of initiatives have resulted in practical guides and standards for how to achieve security. These works are however varying in nature. Some of them focus on technical product assessments such as the ISO/IEC 15408 (Common Criteria) [1], and some on organizational issues such as the ISO/IEC 27002 (previously 17799) [2] and NIST SP 800-53 [3] OCTAVE [4]. A few are oriented towards security for control systems such as the NIST SP 800-82 [7], ANSI/ISA 99.00.01 [5], and the procurement language [7]. Even more specifically there are work focusing on the power industry. E.g., IEC TC 57 has published technical reports [8][9] describing how security can be managed within the domain of power system control and associated communications and the NERC-CIP [10] is acting both as a regulatory standard as well as a guide.

A common denominator for all these guides and standards is that they can all be considered as theoretical frameworks for how to achieve security. None of these works are ensuring “complete” security, but if the guides are followed the idea is that the security of the addressed system or organization will be increased. Looking at them from this point of view, they can thus be seen as theory for how to achieve security. A problem with them is however that it is not clear, neither internally in a standard nor externally between standards, how all the different promoted features and mechanisms are related to each other and if some are more important than others. As a simple example most works promotes the idea that firewalls will increase, or in other words causally positively affect, the level of security. But are firewalls more or less important than say a security awareness program when it comes to achieving security? And isn't it so that an awareness program increases the chances that the firewalls are correctly configured? These kinds of phenomena and the strength and structure of the causal relations are typically not addressed in the standards and guides.

III. ENTERPRISE ARCHITECTURE MODELS

Enterprise architecture is an approach to management of information systems, including control systems, that relies on models of the systems and their environment. The main idea is very old, instead of building the systems from scratch using trial and error; a set of models is created to predict the behavior and effects of changes to the system. The models allow reasoning about the consequences of various scenarios

and thereby support decision making. A large number of enterprise architecture frameworks have been proposed in recent years, including the Zachman Framework [11] the Department of Defense Architecture Framework (DoDAF) [12] and the Open Group Architecture Framework (TOGAF) [13]. In the power industry, the most prominent framework is that developed by EPRI within the Intelligrid project [14]. However, when considering the suitability of the metamodels, i.e. the modeling language, proposed in these frameworks to the theoretical analysis discussed in the preceding chapter, there are significant difficulties. First, a number of the proposed metamodels are not detailed enough to provide the information required for the analyses. Secondly, many metamodels do not systematically propose attributes that are useful for the analysis. Finally and perhaps most importantly, many of the frameworks do not contain the classes that would be required to model the control systems. If we for example are interested in protection system reliability, the models need to answer questions regarding for instance hardware redundancy, component coupling, and reliability of communication links. What kind of information is contained in a model is given by its metamodel, so it is important that enterprise architecture metamodels are properly designed. In order to determine if a metamodel is useful for the analysis of cyber security, it would be helpful with a structured description of that analysis given the metamodel.

Looking to software and security engineering we do indeed find a number of modeling languages that have been tailored for security, such as UMLsec [15], secure UML [16] and Misuse cases [17]. These kinds of languages are providing good support for detailed modeling of concerns such as access control formal validation of security design. However, the drawback is that they are lacking in holistic scope and are neither representing the broad spectrum of security. Moreover are they neither aligned with other system topics of interest such as maintainability, performance, functionality, and business alignment.

The modeling language Coras [18] takes a more general approach and describes how threat scenarios can be modeled with a specific notation. Coras do however not provide any direct support for deriving scenarios based on system models, and does not declare how analysis can be performed based on the created models. Another more holistic approach is described in [19] where enterprise models are coupled with dependencies to analyze security threats. This approach is quite similar to the one presented in this paper but it does not account for the uncertainty that relates to security analysis. Another difference is that our approach is focusing on cyber security whereas [19] is intended for enterprise security analysis.

IV. COMBINING SECURITY THEORY AND MODELS

The following chapter briefly outlines the structure of the work carried out by the VIKING project [20] and the Swedish competence center for electric power engineering [21] on the topic of cyber security analysis and modeling. It combines

attack- and defense graphs with Bayesian statistics and enterprise architecture modeling.

A. Attack Trees and Defense Trees

Attack trees are a graphical notation evolved from fault trees, where the main goal of an attacker is depicted as the root of a tree [22]. The steps to reach this goal are broken down into sub-goals of the attack through “AND” and “OR” relationships, which represent mandatory or optional steps an adversary faces when attacking a system. This is a standard, intuitive way of modeling threats and security. The attack trees can be used to answer questions about the current security status and facilitate comparison with previous measurements, but does not answer questions about how to improve the security status. A natural extension of attack graphs is to include not only attacks, but also countermeasures. From the perspective of the person developing and maintaining the systems, this amounts to adding controllable elements to the tree. The concept of including countermeasures in the tree structure has been used in [23] to create something called “defense trees”, illustrated in Figure 1.

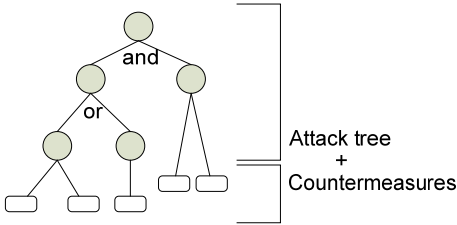


Figure 1. The defense tree concept.

Attack- and defense trees can easily grow extensively as several different attacker goals are of relevance which create a forest of attack paths. To represent attack structures more compactly and to illustrate how well different countermeasures protect against the various attacks, statistical mathematics in terms of Bayesian networks have been proposed and used in attack and defense trees (graphs) [24][25][26][27].

B. Bayesian Networks and Extended Influence Diagrams

Influence diagrams are a powerful modeling approach, used to depict and analyze complex causal interplay between properties [28]. The diagrams [29][10] are an enhancement of Bayesian networks (cf. [30][31]). In influence diagrams, random variables graphically represented as chance nodes may assume values, or states, from a finite domain (cf. Figure 2).

Influence diagrams employ the same mathematical rigor for describing relations between random variables. Given a domain of random variables, chance nodes, X_1, \dots, X_n . Each chance node, X_i , may assume a value x_i from the finite domain $\text{Val}(X_i)$.

Influence diagrams do, in addition to chance nodes, include variables representing deterministic decisions alternatives using decision nodes; and the possibility to represents goals

(or value) using so called utility nodes. A decision could for example be “scenario A” or “scenario B” and a utility node can be “Cost of data loss”. Decision nodes can be used as chance nodes: to either influence another node or to be influenced by it. A utility node can however on be influenced by other types of nodes.

The mathematical inference engine that influence diagrams provide allows modeling of complex decision problems. Coupled to this mathematical engine is also a graphical notation where the variables are represented as a graph. The advantage of the graph representation is that it provides a compact way of expressing the dependency relations between the variables, i.e. which variables influence each other. See Figure 2.

Extended influence diagram syntax

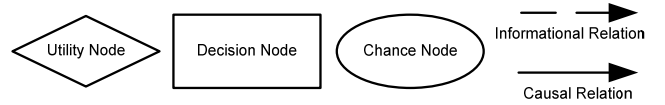


Figure 2. Syntactic elements of extended influence diagrams.

As illustrated in the example diagram of Figure 3, extended influence diagrams can be used to represent defense trees. A utility node can be used to represent the consequence of successful attacks and the steps required for their success can be decomposed into a number of sub steps. Attack steps will then assume the state “Success” or “Failure”, depending on the states of its parents. The states of countermeasures influence the probability that an attack will be successful. Thus, they are modeled as causal parents to the attack steps. Finally, depending on the scenario chosen, the states of countermeasures will differ. This can be represented by decision nodes that influence the state of countermeasures. [25].

Example diagram

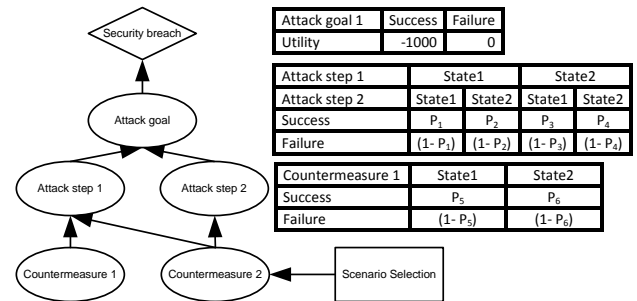


Figure 3. Syntactic elements of extended influence diagrams and a simple example.

The value of utility nodes is defined in terms of the states of the influencing nodes. In order to specify the joint probability distribution of the nodes in the model, the respective conditional probabilities that appear in the product form (1) must be defined.

$$P(X_1, \dots, X_n) = \prod_{i=1}^n P(X_i | Pa(X_i)) \quad (1)$$

The second component P describes distributions for each

possible value x_i of X_i , and $\text{Pa}(X_i)$, where $\text{Pa}(X_i)$ is the set of parent nodes of X_i . These conditional probabilities are typically represented in tables. Using an influence diagram, it is possible to answer questions such as what is the probability of variable X being in state x_1 given that its parents Y and Z are in states y_2 and z_1 ($Y = y_2$ and $Z = z_1$).

Together with the graphical representation, Figure 3 depicts examples of conditional probability tables representing the probabilities of success in various attacks.

One important feature of the Bayesian formalism is the possibility to learn from previous data and create powerful statistical models for accurate assessments of for instance cyber security. Since influence diagrams, as opposed to mere Bayesian networks, include the notions of decision and utility nodes, predicted losses from successful attacks can be included in the models, thus enabling a more holistic view of the cyber security problem.

C. Abstract Models for Cyber Security

Thus far in the paper the focus has been on the analysis framework that is setting up the structure for assessing cyber security. We now turn to the models and describe how the influence diagram defense tree can be combined with enterprise architecture models into so called abstract models [34]. The purpose of abstract models is to ensure alignment between what is included in the modeling language (metamodel) and the property we are interested in analyzing, i.e. cyber security in this case.

An abstract model comprise of four components: entities, entity relationships, attributes and attribute relationships. The first three of these components can be recognized from standard modeling languages such as the class diagrams of the UML. Entities are a central component in most modeling languages and can as in class diagrams be used to represent concepts of relevance for the model. These can be either physical artifacts, such as “computer” and “person”, or more concepts such as “data” and “procedure”. Entities are represented in a similar way as classes in UML are: a rectangular box with the name of the entity specified at its top.

Entities can in abstract models be connected through entity relationships. These entity relationships are depicted as lines spanning between the entities with roles names and multiplicities declared at the endpoints.

Attributes of abstract models are as in UML held by entities and are depicted as squared boxes within the entity belong to. However, unlike in UML, they are random variables or utility variables of finite domains. In other words, the attributes of the abstract model are the chance and utility nodes of the extended influence diagram.

Finally, and thus naturally, abstract models in addition to other modeling languages have the attribute relationship. This is relationship is the same as the relationship in the extended influence diagrams. If these attribute relationship span between two entities, it is always associated with a particular entity relationship, which is denoted by the dashed line, for indicating which entity relationship that is the reason why the

attribute relationship exists. Cf. Figure 4.

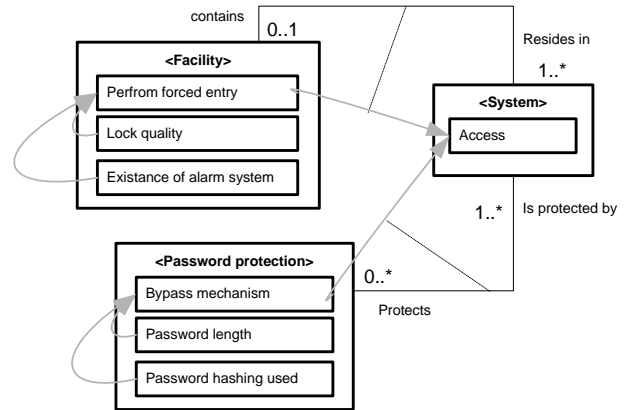


Figure 4 – Example of an abstract model.

Abstract models can thus be seen as metamodels enhanced with extended influence diagrams. This enhancement is not as straightforward as it perhaps seem. The reason for this is that the extended influence diagram does not differentiate between the instantiated and abstract modes. For instance, as a result of the multiplicities of entity relationships, the number of parents an attribute has may differ between instances of the abstract model. One way to handle this when describing the “instantiated extended influence diagram” is to use aggregation functions to specify the conditional decency an attribute has on its parents. Examples of such aggregation functions are “AND”, “OR”, “AVERAGE” and “MAX”.

D. Generating Abstract Models from Defense Trees

Schechter [22] points out that the structure of attack trees depends on the system architecture and the choice of countermeasures. For example, in architectures that contain confidential data, attacks compromising the access control of this data are relevant. The countermeasures included in the architecture are also of relevance since the attack vectors that are available depend on these. The attacks bypassing an access control mechanism based on biometrics does for instance differ from attack against an access control mechanism based on passwords. Also, the multiplicity of countermeasures is of importance since additional countermeasures introduce additional hinders for adversaries.

Abstract models offer a way of handling these dependencies by dictating the attribute relationships as a consequence of an entity relationship. With this as a basis, it can be expressed how the relationship between attack-goals depend on the entities included in a model, and their relationships to each other.

The nodes of a defense graph, expressed as a extended influence diagram, are typically associated with some entity to which they belong. Based on this, the entities that are relevant for the assessment can be identified and populated with the appropriate attributes. For example, the node “Password Strength” can be interpreted as the entity “Password”, holding the attribute “Strength”.

If an entity relationship shall be included in an abstract

model depend on the structure of the associated extended influence diagram. The entity relationships of relevance are those that determine if an attribute relationship shall exist in an instantiated version of the model. The example abstract model in Figure 4 does for instance have the entity relationship “contains” since this relationship between a “facility” and a “system” would imply that the attribute “perform forced entry” in that facility is required to be true before “access” to the system can be gained..

V. CONCLUSIONS

This paper outlines ongoing work of an approach for cyber security management. The approach has been developed with the purpose providing industrial decision-makers with relevant and credible information. It is based on the assumption that decision-makers today need a better holistic understanding of control systems and their surrounding IT- and organizational environment and is thus suggesting enterprise architecture models as management support. However, the existing (meta)models needs to be adapted to the purpose of cyber security management. In order to do this, we propose so called abstract models built from attack and defense tree relevant for control systems. With this tool the decision maker will be able to make better decisions on a limited knowledge about the details of the control system. With information about the control system stored in a reusable format such as architectural models, a continuous consistent management approach is promoted.

VI. REFERENCES

- [1] ISO15408, Information technology – Security techniques – Evaluation criteria for IT security (Common Criteria), 2005.
- [2] ISO/IEC 27002:2005, Information technology – Security techniques – Information security management systems – Code of practice for information security management
http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=50297
- [3] NIST, SP 800-53 Recommended Security Controls for Federal Information Systems. Revision 1, 2006.
- [4] C.J. Alberts and A.J..Dorofee. Managing Information Security Risks: The OCTAVE. Approach. Addison-Wesley, June 2002.
- [5] Stouffer, K., J. Falco, and K. Scarfone, *Guide to Industrial Control Systems (ICS) security*. Technical Report Special Publication 800-82, NIST, 2008.
- [6] ANSI/ISA–99.00.01–2007, Security for Industrial Automation and Control Systems Part 1: Terminology, Concepts, and Models, USA, 2007.
- [7] G. Finco et al., Cyber Security Procurement Language for Control Systems, Department of Homeland Security, 2008.
- [8] IEC 62210, Power control and associated communications – Data and communications security, 2003
- [9] IEC, Power system control & associated communications - Data & communication security, 62351 part 1-6, TS, 2007.
- [10] NERC, Critical Infrastructure Protection (CIP) 002-009, January, 2006.
- [11] Zachman, J.A.: A Framework for Information Systems Architecture. IBM Systems Journal. 26, 276 – 292, 1987.
- [12] Department of Defense Architecture Framework Working Group: DoD Architecture Framework, version 1.5. Department of Defense, USA (2007)
- [13] The Open Group: TOGAF 2007 edition, Van Haren Publishing, Zaltbommel, Netherlands (2008)
- [14] Hughes, J., The Integrated Energy and Communication Systems Architecture, Electric Power Research Institute, 2004.

- [15] Jürjens, j., Secure Systems Development with UML, Springer Berlin Heidelberg, Germany, 2005.
- [16] T. Lodderstedt, D. Basin, and J. Doser. SecureUML: A UML-Based Modeling Language for Model-Driven Security. In Proc. of UML’02, LNCS 2460, pages 426–441. Springer-Verlag, 2002.
- [17] G. Sindre and A. L. Opdahl. Eliciting Security Requirements by Misuse Cases. In Proc. of TOOLS Pacific 2000, pages 120–131. IEEE Press, 2000.
- [18] Hogganvik, I., A Graphical Approach to Security Risk Analysis, PhD thesis, Faculty of Mathematics and Natural Sciences, University of Oslo, 2007.
- [19] Breu, R., Innerhofer-Oberperfler, F., Yautsiukhin, A., Quantitative Assessment of Enterprise Security System, Third International Conference on Availability, Reliability and Security, 2008. ARES 08., vol., no., pp.921-928, 4-7 March 2008.
- [20] Collaborative Project 225643 within EU FP7-ICT-SEC-2007-1: Viking—Vital Infrastructure, networkS, INformation and control systems management.
- [21] Centre of Excellence in Electric Power Engineering, EKC², Royal Institute of Technology, <http://www.comp.ee.kth.se/>
- [22] S.E. Schechter. Computer Security Strength & Risk: A Quantitative Approach. PhD thesis, Harvard University, 2004.
- [23] S. Bistarelli, F. Fioravanti, P. Peretti, Defense trees for economic evaluation of security investments, Proceedings of Availability, Reliability and Security (ARES), Vienna, Austria, 2006, pp. 8.
- [24] Y. Liu and M. Hong, Network vulnerability assessment using Bayesian networks, Proceedings of Data Mining, Intrusion detection, Information assurance and Data networks security, Orlando, Florida, USA, 2005, pp
- [25] T. Sommestad, M. Ekstedt, P. Johnson, Combining defense graphs and enterprise architecture models for security analysis, Proceedings of the 12th IEEE International Enterprise Computing Conference, September 2008
- [26] Sommestad, T., Ekstedt, M., Johnson, P., Cyber Security Risks Assessment with Bayesian Defense Graphs and Architectural Models, to appear in proceedings of Hawaii International Conference on System Sciences (HICSS), 2009.
- [27] X. Qin, and W. Lee, “Attack plan recognition and prediction using causal networks”, in Proceedings of the 20th Annual Computer Security Applications Conference, Tucson, AZ, USA, 2004, pp. 370-379.
- [28] P. Johnson, R. Lagerstrom, P. Narman, M. Simonsson., Enterprise Architecture Analysis with Extended Influence Diagrams. Information System Frontiers, 9(2), Springer Netherlands, 2007, pp. 163-180..
- [29] R. Shachter, Probabilistic inference and influence diagrams. Operations Research, 36(4), Hanover Maryland, 1988, pp. 36-40.
- [30] Neapolitan, R. Learning Bayesian Networks. Prentice-Hall, Inc. Upper Saddle River, NJ, USA 2003.
- [31] Jensen, F.V., Bayesian Networks and Decision Graphs, Springer New York, Secaucus, NJ, USA 2001.
- [32] R. Shachter, Evaluating influence diagrams. Operations Research, 34(6), Institute for Operations Research and the Management Sciences, Hanover Maryland, 1986, pp. 871-882.
- [33] R.A Howard, J.E. Matheson, influence diagrams. Decision Analysis, 2(3), Institute for Operations Research and the Management Sciences, Hanover Maryland, 2005, pp. 127–143.
- [34] P. Johnson, E. Johansson, T. Sommestad, and J. Ullberg, A Tool for Enterprise Architecture Analysis, In Proceedings of Enterprise Distributed Object Computing Conference, 2007, pp. 142-142.

VII. BIOGRAPHIES



Mathias Ekstedt received his M.Sc. in electrical engineering and Ph.D. in industrial information and control systems from the Royal Institute of Technology (KTH) in Stockholm, Sweden in 1999 and 2004 respectively.

He is currently a research associate at the Royal Institute of Technology and the manager of the program IT Applications in Power System Operation and Control within the Swedish Centre of Excellence in Electric Power Engineering. He’s research interests includes information and control systems and enterprise architecture for the power industry.

Dr. Ekstedt is a member of the Task force on Cyber Security of Power Systems within IEEE PES Technical Committee on Power System Analysis, Computing and Economics Subcommittee of Computer and Analytical Methods. He is a member of INCOSE and a former member of the Swedish INCOSE Chapter board. He is also the founder of the enterprise architecture network at the Swedish Computer Society.



Teodor Sommestad (M'08) received a M.Sc degree in computer science at the KTH – the Royal Institute of Technology, Stockholm, Sweden.

He is currently a PhD student at the department Industrial Information and Control systems at KTH. His research interests are security of industrial control system and enterprise architecture modeling, primarily the security SCADA systems used for power network management.

Mr. Sommestad is a member of ISACA and is active within the research committee of the

Swedish Chapter.