

IT Security Governance Innovations: Theory and Research

Daniel Mellado
Rey Juan Carlos University, Spain

Luis Enrique Sánchez
University of Castilla - La Mancha, Spain

Eduardo Fernández-Medina
University of Castilla - La Mancha, Spain

Mario Piattini
University of Castilla - La Mancha, Spain

Managing Director: Lindsay Johnston
Senior Editorial Director: Heather A. Probst
Book Production Manager: Sean Woznicki
Development Manager: Joel Gamon
Development Editor: Myla Merkel
Assistant Acquisitions Editor: Kayla Wolfe
Typesetter: Travis Gundrum
Cover Design: Nick Newcomer

Published in the United States of America by
Information Science Reference (an imprint of IGI Global)
701 E. Chocolate Avenue
Hershey PA 17033
Tel: 717-533-8845
Fax: 717-533-8661
E-mail: cust@igi-global.com
Web site: <http://www.igi-global.com>

Copyright © 2013 by IGI Global. All rights reserved. No part of this publication may be reproduced, stored or distributed in any form or by any means, electronic or mechanical, including photocopying, without written permission from the publisher. Product or company names used in this set are for identification purposes only. Inclusion of the names of the products or companies does not indicate a claim of ownership by IGI Global of the trademark or registered trademark.

Library of Congress Cataloging-in-Publication Data

IT security governance innovations : theory and research / Daniel Mellado ... [et al.], editors.
p. cm.

Includes bibliographical references and index.

Summary: "This book provides extraordinary research that highlights the main contributions and characteristics of existing approaches, standards, best practices and new trends in IT Security Governance"--Provided by publisher.

ISBN 978-1-4666-2083-4 (hardcover) -- ISBN 978-1-4666-2084-1 (ebook) -- ISBN 978-1-4666-2085-8 (print & perpetual access) 1. Computer security--Management. 2. Information technology--Security measures. 3. Business enterprises--Security measures. 4. Data protection. I. Mellado, Daniel, 1980-

QA76.9.A25I894 2013

005.8--dc23

2012014224

British Cataloguing in Publication Data

A Cataloguing in Publication record for this book is available from the British Library.

All work contributed to this book is new, previously-unpublished material. The views expressed in this book are those of the authors, but not necessarily of the publisher.

Chapter 10

Using Indicators to Monitor Security Risk in Systems of Systems: How to Capture and Measure the Impact of Service Dependencies on the Security of Provided Services

Olav Skjelkvåle Ligaarden
SINTEF ICT & University of Oslo, Norway

Atle Refsdal
SINTEF ICT, Norway

Ketil Stølen
SINTEF ICT & University of Oslo, Norway

ABSTRACT

Systems of systems are collections of systems interconnected through the exchange of services. Their often complex service dependencies and very dynamic nature make them hard to analyze and predict with respect to quality in general, and security in particular. In this chapter, the authors put forward a method for the capture and monitoring of impact of service dependencies on the security of provided services. The method is divided into four main steps focusing on documenting the system of systems and IT service dependencies, establishing the impact of service dependencies on risk to security of provided services, identifying measurable indicators for dynamic monitoring, and specifying their design and deployment, respectively. The authors illustrate the method in an example-driven fashion based on a case within power supply.

DOI: 10.4018/978-1-4666-2083-4.ch010

INTRODUCTION

In today's business environment, companies (businesses, enterprises, organizations) co-operate with other parties by providing and/or requiring information and communication technology (ICT) supported services. The ICT-systems facilitating such co-operation are often so-called system of systems (SoS). An SoS may be thought of as a kind of "super system" comprising a set of interconnected systems that work together towards some common goal.

(Allen, 2005) defines governance as "*setting clear expectations for the conduct (behaviors and actions) of the entity being governed, and directing, controlling, and strongly influencing the entity to achieve these expectations.*" In an SoS setting, a company is often expected to provide services fulfilling requirements to security. If the services are not provided according to their security requirements, then it may have severe consequences for the company providing them. Thus, the company needs to govern the security of the provided services. Risk assessment is a necessity for ensuring that risks to security of provided services are at an acceptable level. However, it is not straight-forward to assess risk to security of provided services in an SoS. Firstly, the exchanged services may require other services in order to function. Such requirements result in so-called service dependencies. Change in the security attributes of one service may easily cause the security attributes of its dependent services to change as well. Secondly, the different systems may be under different managerial control and within different jurisdictions. For the systems that are outside our control, we have limited knowledge of their security risks, structure, and behavior. Thirdly, such a large number of systems, controlled and operated by different parties, evolve rapidly in a manner that may be difficult to predict.

To cope with this situation we propose the use of detailed dependency models to capture the

impact of services dependencies, trust relations as a basis for analysis in the case of insufficient documentation, and monitoring to cope with evolution. Our main result is a method facilitating the set-up of such monitoring. This method can be used in security governance for the purpose of assessing to what extent the security expectations to the provided services are achieved.

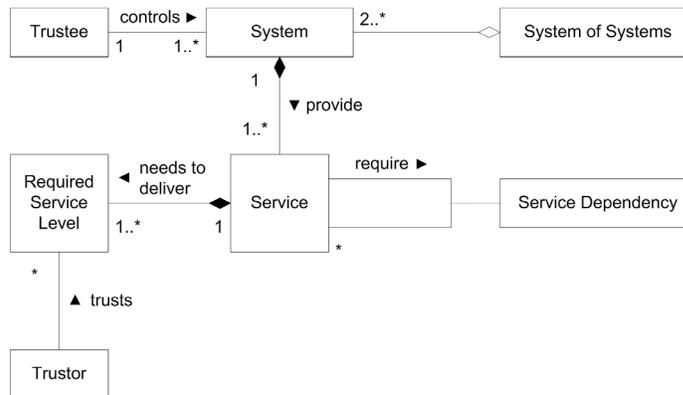
The method is divided into four steps. Service dependencies and trust relations are identified and documented in the first step. In the second step we conduct a security risk analysis to capture the impact of service dependencies on risk to security of provided services. The identified trust relations are used when analyzing service dependencies involving systems of which we have insufficient documentation. In the third step we identify the security risks to be monitored, as well as measurable indicators for monitoring their risk values. In the fourth and final step we specify how these indicators should be designed, i.e., how they should be calculated, and deployed in the SoS, i.e., how data needed in the calculations should be extracted and transmitted within the SoS in question. The result of applying the method is a security risk picture parameterized by indicators, each defined by design and deployment specifications.

The rest of the chapter is organized as follows: in the next section (Section 2) we introduce basic terminology and definitions. Section 3 presents the methodological approach, while the four steps of the approach are demonstrated on an example case within power supply in Sections 4 – 7. In Section 8 we present related work, while we conclude and indicate further research in Section 9.

BASIC TERMINOLOGY AND DEFINITIONS

In this section we provide basic terminology, definitions, and conceptual models for system of systems, risk, and related concepts.

Figure 1. Conceptual model relating system, system of systems, and other concepts



System of Systems and Related Concepts

Our definition of SoS is based on the definitions of (Office of the Under Secretary of Defense for Acquisition, Technology, and Logistics, 2002) and (Jamshidi, 2008): “A system of systems (SoS) is a set or arrangement of systems that are related or connected to fulfill common goals. The different systems may be controlled, operated, and maintained by different parties and within different jurisdictions. The loss of any system may seriously impact the other systems and the process of fulfilling the common goals.”

An SoS may arise naturally from the interconnection of individual systems, or it may be built specifically for the purpose of achieving goals that the individual systems cannot achieve alone. An example of the former is the interconnection of critical infrastructures, while a sensor network, constructed for the purpose of gathering low-level data to be aggregated, is an example of the latter.

We focus on SoS where the systems are interconnected through the exchange of services. In Figure 1 is a conceptual model, in the form of a UML (OMG, 2004) class diagram, relating system, system of systems, and other concepts. The associations between the different concepts have cardinalities that specify how many instances of

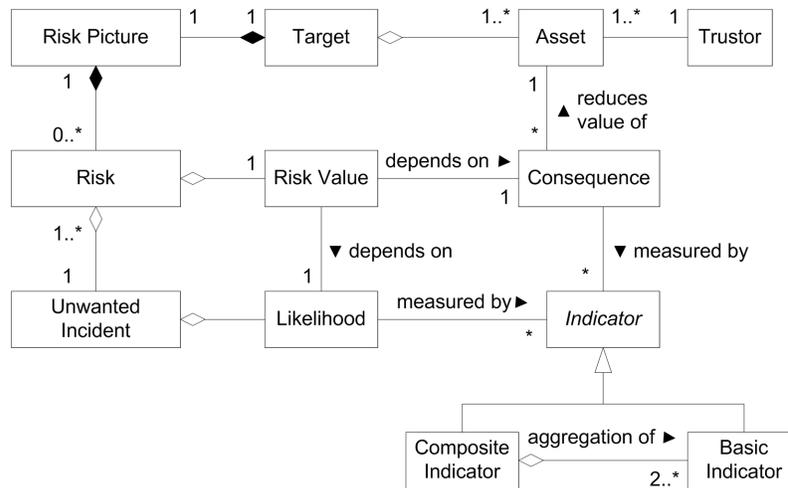
one concept that may be associated to an instance of another concept. The filled diamond specifies composition, while the hollow diamond, used in Figure 2, specifies aggregation.

As shown in Figure 1, a *System of Systems* consists of at least two *Systems*. The different systems may be controlled and operated by different *Trustees*. A system needs to provide at least one *Service*, in order to be recognized as a system. Typically, a service will have a required level of service defined for each area of service scope. These required service levels may for instance be specified in a service-level agreement. Thus, one or more *Required Service Levels* are associated with each service. For each required service level, a *Trustor* may have a certain amount of trust in that the service delivers the required level of service. A service provided by one system may require other services in order to function. Such requirements result in *Service Dependencies*, shown by an association class in Figure 1. These dependencies help us to better understand the importance of the individual services exchanged between the systems.

Risk and Related Concepts

Figure 2 shows a conceptual model for risk and closely related concepts. A *Risk* involves an

Figure 2. Conceptual model for risk and closely related concepts



Unwanted Incident, such as “System operator is unable to control and operate the power plant.” The unwanted incident may occur with a certain *Likelihood*. When the incident occurs, an *Asset* will be damaged (and its value reduced). This is the *Consequence* of the risk. An asset is owned by a *Trustor* and it is something of value that the trustor seeks to protect. It can be a physical thing, e.g., “Power plant,” or conceptual, e.g., “Reputation of trustor.” Since the consequence of an incident depends on the particular asset in question, the same incident may have different consequences for different assets.

By conducting a risk analysis we obtain a *Risk Picture*, consisting of zero or more risks, for the *Target* of analysis. The target characterizes the focus and scope of the analysis. In order to choose and prioritize between treatments, we assign a *Risk Value* to each risk. A risk function calculates the risk value by taking the likelihood of the unwanted incident and its consequence for the asset in question as input. Typically, likelihood is measured in terms of frequency or probability, while the measure of consequence depends on the asset in question.

Zero or more *Indicators* may be used to measure likelihood and consequence values. An

indicator is either basic or composite. Thus, an abstract class (name in italic) is used to represent *Indicator* in the conceptual model. By *Basic Indicator* we mean a measure such as the number of times a specific event generated by the ICT infrastructure has been observed within a given time interval, the average time between each generation of a specific event, the load on the network at a particular point in time, or similar. A *Composite Indicator* is the aggregation of two or more basic indicators.

METHODOLOGICAL APPROACH

An overview of the methodological approach is presented in Figure 3. In the following we describe each of the four main steps as well as their sub-steps in terms of a detailed guideline.

Our intended client enterprise corresponds to the trustor in Figure 2. The trustor controls a fragment of the SoS which we refer to as the target. The target depends on the rest of the SoS that is controlled by other enterprises that may be thought of as trustees of our client enterprise. Our task is to establish a dynamic security risk picture that captures the impact of service dependencies on

Figure 3. Overview of the methodological approach

- Step 1 – Document system of systems**
 - 1.1 – Model system of systems
 - 1.2 – Capture service dependencies
 - 1.3 – Capture trust relations
- Step 2 – Analyze the impact of service dependencies on risk to security of provided services**
 - 2.1 – Identify security assets
 - 2.2 – Construct high-level threat diagrams of the impact of service dependencies on identified security assets
 - 2.3 – Construct detailed threat diagrams of the impact of service dependencies on identified security assets
- Step 3 – Identify indicators for system of systems**
 - 3.1 – Identify security risks to be monitored
 - 3.2 – Identify relevant indicators for the security risks to be monitored
- Step 4 – Specify design and deployment of identified indicators for system of systems**
 - 4.1 – Specify design of indicators for monitoring security risks
 - 4.2 – Specify deployment of indicators for monitoring security risks

risk to the security of the client’s provided services.

Step 1: Document System of Systems

Step 1.1: Model System of Systems

- **Objective:** Model the SoS.
- **Rationale:** To capture the impact of service dependencies on risk to security of provided services, we need to document the services exchanged between the different systems in the SoS. In particular, it is essential to understand the dependencies between the target and the target’s environment, i.e., the rest of the SoS. We also need to document the requirements to the different services, since we are only concerned with the impact of services on risk when they are not delivered according to requirements.
- **How Conducted:** A system model is created by the analysis team based on input documentation provided by the trustor. The system model describes the systems of the target as well as the systems of the target’s environment. It also captures system interactions in the form of service exchanges, and the required service levels. Each required service level is specified for one area of service scope. We can for instance specify the required level of availability, integrity, etc., for a service.
- **Input Documentation:** The trustor provides information on the SoS, their interactions in the form of service exchanges, and the requirements, in the form of required levels of service, for each service exchanged.
- **Output Documentation:** A system model documenting:
 - The systems of the target and environment.
 - The interactions between the systems in the form of service exchanges.
 - The required service levels for each service.

- Modeling Guideline:** The SoS is modeled in the form of a graph, as illustrated by Figure 4. The system elements (vertices) in the graph represent systems, while service relations (edges) represent exchanges of services. The bold rectangular container with rounded corners separates the target from its environment. Each system element is annotated with the trustor/trustee controlling and operating the system represented by the element, while each service relation is annotated with the service exchanged and its required levels of service. In Figure 4 this has only been shown for two service relations, in order to save space. For one of the service relations, a required service level has been specified for one area of service scope, while required service levels have been specified for two areas of service scope for the other ser-

vice. Here, *A* stands for availability, while *I* stands for integrity.

The source of a service relation represents the provider of the service, while the target of the relation represents the consumer of the service. A system may need to consume services in order to provide other services. If one system provides two or more services to another system, then the model is a multigraph, i.e., a graph which allows multiple edges, meaning edges with the same pair of source and target vertices.

Step 1.2: Capture Service Dependencies

- Objective:** Identify and document service dependencies within the SoS.
- Rationale:** In Step 1.1 we documented the services exchanged between the different systems. In this step we identify the service

Figure 4. System model

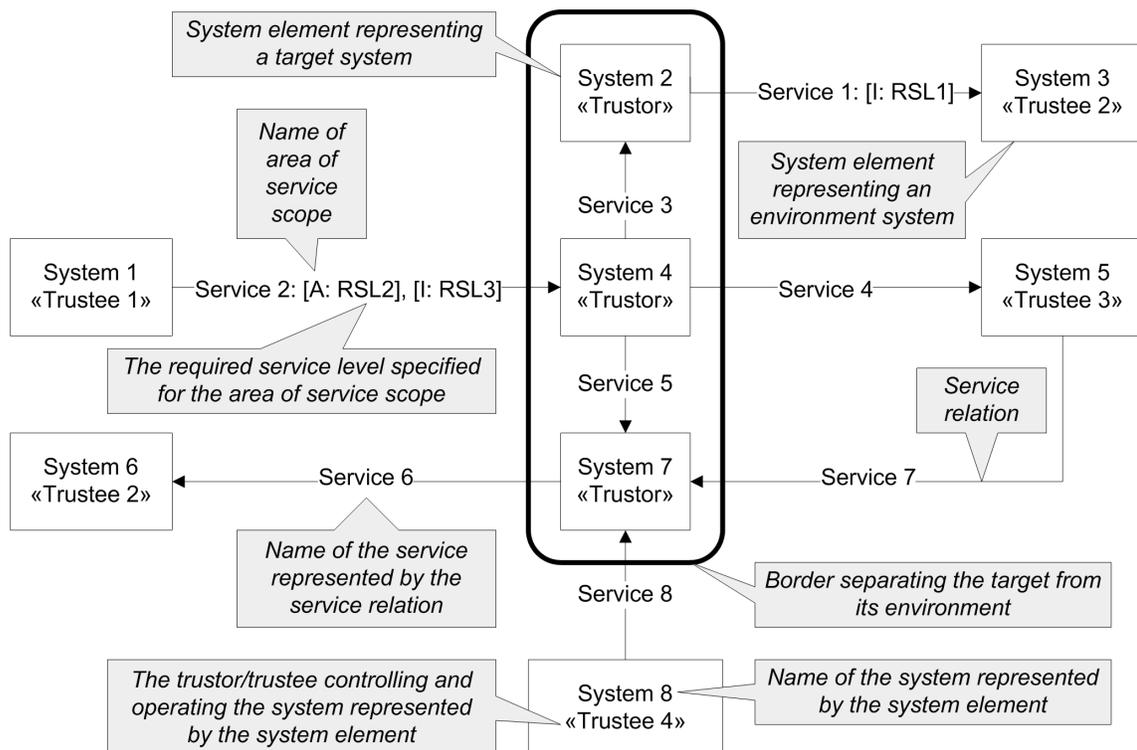
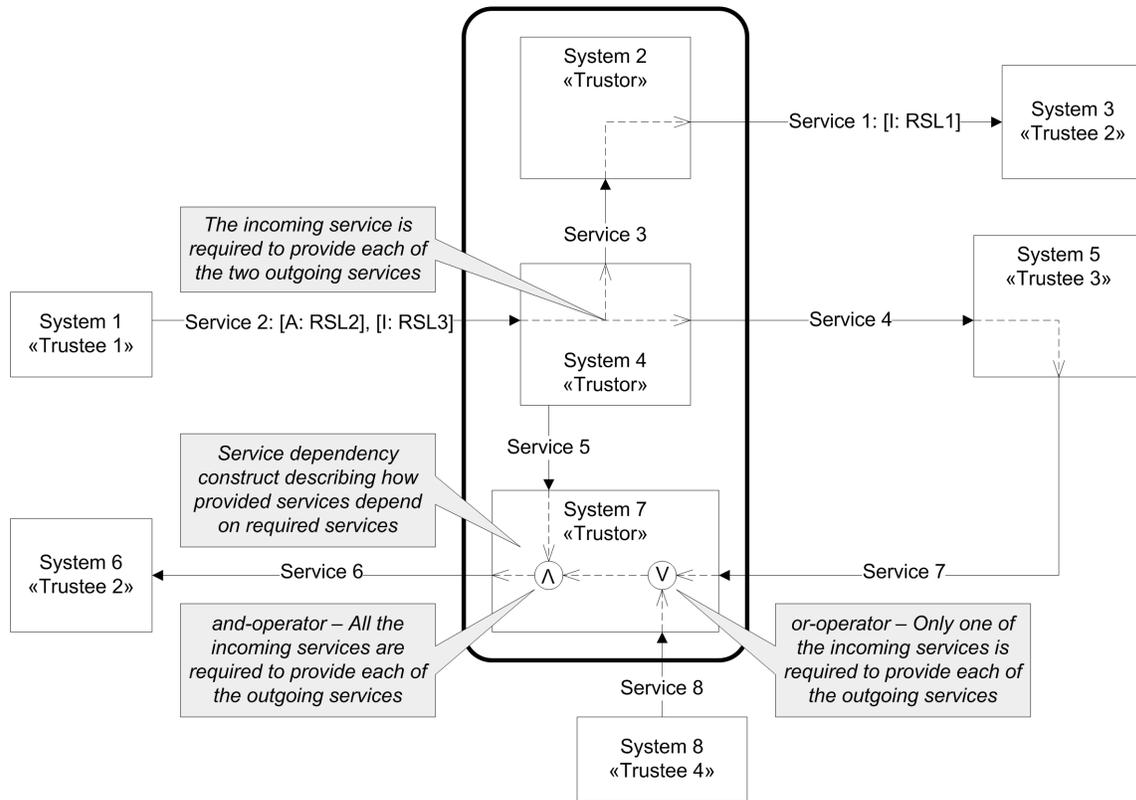


Figure 5. System model annotated with service dependencies



dependencies resulting from the exchange of these services. This enables us to analyze the impact of service dependencies on risk to security of provided services.

- **How Conducted:** The system model, from Step 1.1, is annotated with service dependencies, based on input documentation provided by the trustor. The annotated model shows how provided services depend on required services.
- **Input Documentation:**
 - The system model from Step 1.1.
 - The trustor provides information on the relationships between required and provided services for the different systems documented in the system model.

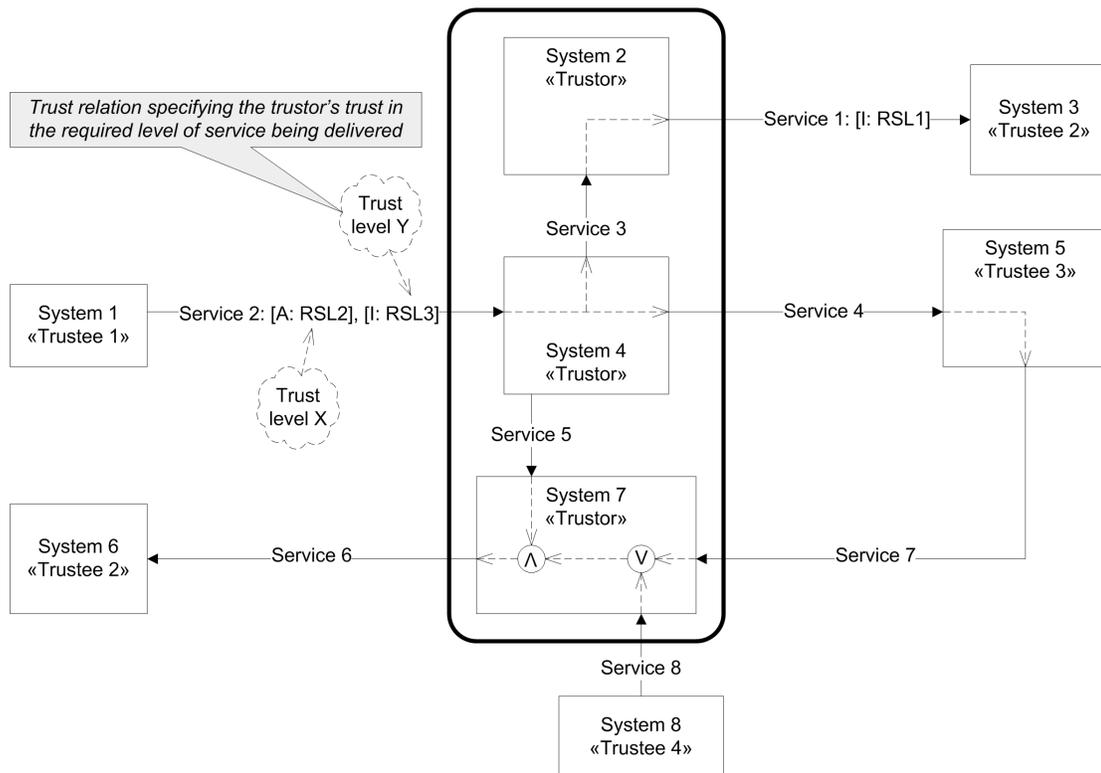
- **Output Documentation:** The system model, from Step 1.1, annotated with service dependencies.
- **Modeling Guideline:** Figure 5 shows the system model in Figure 4 annotated with service dependency constructs. The constructs describe dependencies between provided and required services. Dependencies between required and provided services are combined with “and” (Λ) or “or” (V) operators. The meaning of the and-operator is that all the incoming services are required to provide each of the outgoing services, while the meaning of the or-operator is that only one of the incoming services is required to provide each of the outgoing services. As seen in Figure 5, operators may be combined to express dependencies that cannot be expressed by a single operator alone.

Step 1.3: Capture Trust Relations

- **Objective:** Document the trustor’s trust in the required levels of services being delivered by its trustees.
- **Rationale:** A trustor will normally not have detailed knowledge of the interior of systems owned by its trustees. Moreover, they may be changed and updated in a manner not controlled by the trustor. Hence, services provided by environment systems are difficult to analyze due to lack of documentation as well as control. To cope with this lack of knowledge we use trust levels to reason about the failure of environment systems to provide their services to target systems according to required service levels. Each trust level states the degree to which the trustor trusts the required service level to be delivered.

- **How Conducted:** The system model, from Step 1.2, is annotated with trust relations. Each trust relation relates a trust level (in the interval [0,1]) determined by the trustor to a required service level of a service provided by an environment system to a target system.
- **Input Documentation:** The system model from Step 1.2.
- **Output Documentation:** The system model, from Step 1.2, annotated with trust relations.
- **Modeling Guideline:** Figure 6 shows the system model in Figure 5 annotated with trust relations. The trust relations are shown with dotted clouds. Each cloud is assigned to a required service level of a service provided by an environment system to a target system.

Figure 6. System model annotated with trust relations



Step 2: Analyze the Impact of Service Dependencies on Risk to Security of Provided Services

Step 2.1: Identify Security Assets

- **Objective:** Identify the security assets for which impact of service dependencies should be analyzed.
- **Rationale:** By identifying the security assets we restrict the identification of risks caused by service dependencies to only those risks that may harm the security assets of the trustor, i.e., reduce their values. By doing so, we ensure that the available time and resources are spent identifying the most critical and important security risks for the trustor in question.
- **How Conducted:** For each provided service, the trustor identifies the security assets for which protection is required. Zero or more security assets may be identified for each provided service. The value of the security asset is reduced if the service level is less than the required service level.
- **Input Documentation:** None.
- **Output Documentation:** Zero or more security assets for each provided service.

Step 2.2: Construct High-Level Threat Diagrams of the Impact of Service Dependencies on Identified Security Assets

- **Objective:** Achieve an initial high-level understanding of the impact of service dependencies on the identified security assets by schematically constructing threat diagrams from the system model.
- **Rationale:** In order to conduct a detailed analysis of the impact of service dependencies on risk to security of provided services, we first establish an initial high-level understanding of how the failure of

individual systems to deliver their services according to requirements may lead to the failure of other individual systems to deliver their services according to requirements, as a result of the service dependencies. Moreover, we establish how this eventually may lead to unwanted incidents that harm the identified security assets if they occur. Such an initial high-level understanding is achieved by schematically constructing a threat diagram for each provided service with security assets.

- **How Conducted:** Figure 7 shows the schematic construction of a threat diagram from an excerpt of the system model in Figure 6. The threat diagram provides an initial overview of how the security asset “Integrity of Service 1 delivered to System 3” may be harmed if the systems represented by the referring threat scenarios fail to deliver their services according to their required service levels.

We use CORAS (Lund, Solhaug, & Stølen, 2010), which is a model-driven approach to asset-oriented risk analysis, for the modeling and analysis of risk. The threat diagram is expressed in the CORAS language. Examples of the semantics (as defined in (Lund et al., 2010)) of elements and relations are given in Step 2.3 of the demonstration of the methodological approach on the example case. As can be seen in the diagram, referring threat scenarios, vulnerabilities, and the referring unwanted incident have been given names following the conventions “Services not delivered by System X according to requirements,” “System Y depends on Service Z,” and “Incident with impact on the A,” (where A is the name of the asset) respectively.

A threat diagram is constructed from the system model resulting from Step 1.3 for each provided service where security assets were identified in Step 2.1 by using the following schematic procedure¹:

1. Create an excerpt of the system model by removing model elements in the following order:
 - a) All service relations representing services that are different from the provided service in question and that:
 - i. Are provided by target systems to environment systems.
 - ii. Are both provided and required by environment systems.
 - iii. The provided service in question does not depend directly or indirectly on.
Modify/remove the service dependency constructs of system elements representing target systems that were connected to the removed service relations.
 - b) All system elements that are no longer connected to other system elements through service relations.
 - c) The system element representing the environment system that requires the provided service in question, unless the environment system provides services that the provided service in question depends directly or indirectly on.
 - d) All trust relations that are no longer associated with service relations, and all service dependency constructs of system elements representing environment systems.
 - e) The border separating the system elements representing the target systems from the system elements representing the environment systems.
 2. Replace each system element with a referring threat scenario. This scenario represents the failure of the system to provide its services according to their required service levels.
 3. Insert the security assets of the provided service into the excerpt, and insert a referring unwanted incident for each security asset. Insert an impacts relation between the referring unwanted incident and its security asset.
 4. Replace the service relation representing the provided service with as many leads-to relations as there are referring unwanted incidents. Let the source of each leads-to relation be the referring threat scenario representing the system delivering the provided service and let the target of each leads-to relation be one of the referring unwanted incidents.
 5. Replace each service relation between two referring threat scenarios with a leads-to relation, and assign a vulnerability to each relation. The vulnerability denotes that the system represented by the target scenario depends on the service represented by the replaced service relation.
For all leads-to relations in the threat diagram, the source and target of the relation is an out-gate and in-gate, respectively. The gates are connected to referring threat scenarios and unwanted incidents. Moreover, the source of each impacts relation is an out-gate, where the out-gate is connected to a referring unwanted incident. In-gates and out-gates are explained in more detail in Step 2.3 of the demonstration of the methodological approach on the example case.
- **Input Documentation:**
 - The system model from Step 1.3.
 - The identified security assets from Step 2.1.
 - **Output Documentation:** One high-level threat diagram outlining the impact of service dependencies on the security assets for each provided service where security assets were identified in Step 2.1.

Step 2.3: Construct Detailed Threat Diagrams of the Impact of Service Dependencies on Identified Security Assets

- **Objective:** Achieve a detailed understanding of the impact of service dependencies on the identified security assets.
- **Rationale:** The threat diagrams from Step 2.2 provide only a high-level outline of the impact of service dependencies on the identified security assets. To establish a security risk picture that can be monitored, we need to detail those diagrams.
- **How conducted:** In Figure 8 is a threat diagram (where some of the details have been suppressed) that shows part of the result of detailing the threat diagram in Figure 7. The referring threat scenarios have been decomposed by creating referenced threat scenarios. Each referenced threat scenario describes the internal threat behavior of the system represented by a referring threat scenario. Moreover, the referring unwanted incident has been decomposed by creating a referenced unwanted incident. The referenced unwanted incident documents the different unwanted incidents that may arise due to the service dependencies. Examples of referenced threat scenarios and unwanted incidents are given in Step 2.3 of the demonstration of the methodological approach on the example case. The two vulnerabilities “V1” and “V2” are the results of the decomposition of the high-level vulnerability “System 2 depends on Service 3,” documented in Figure 7. Symbolic names have been used for the two vulnerabilities in Figure 8. By describing the internal threat behavior of “Services not delivered by System 4 according to requirements,” these symbolic names can be replaced by names of real vulnerabilities. For each of these two vulnerabilities, a con-

ditional likelihood has been assigned to the same leads-to relation as the vulnerability is assigned. We can also see that the vulnerability “System 4 depends on Service 2” has been decomposed into the two vulnerabilities “System 4 depends on availability of Service 2” and “System 4 depends on integrity of Service 2”; one for each of the required service levels associated with “Service 2.” As a result of the decomposition of the high-level vulnerabilities, the referring threat scenarios, and the referring unwanted incident, the high-level in-gates and out-gates and the impacts relation have been decomposed, and likelihood values and consequences values have been assigned to the gates and impacts relations, respectively.

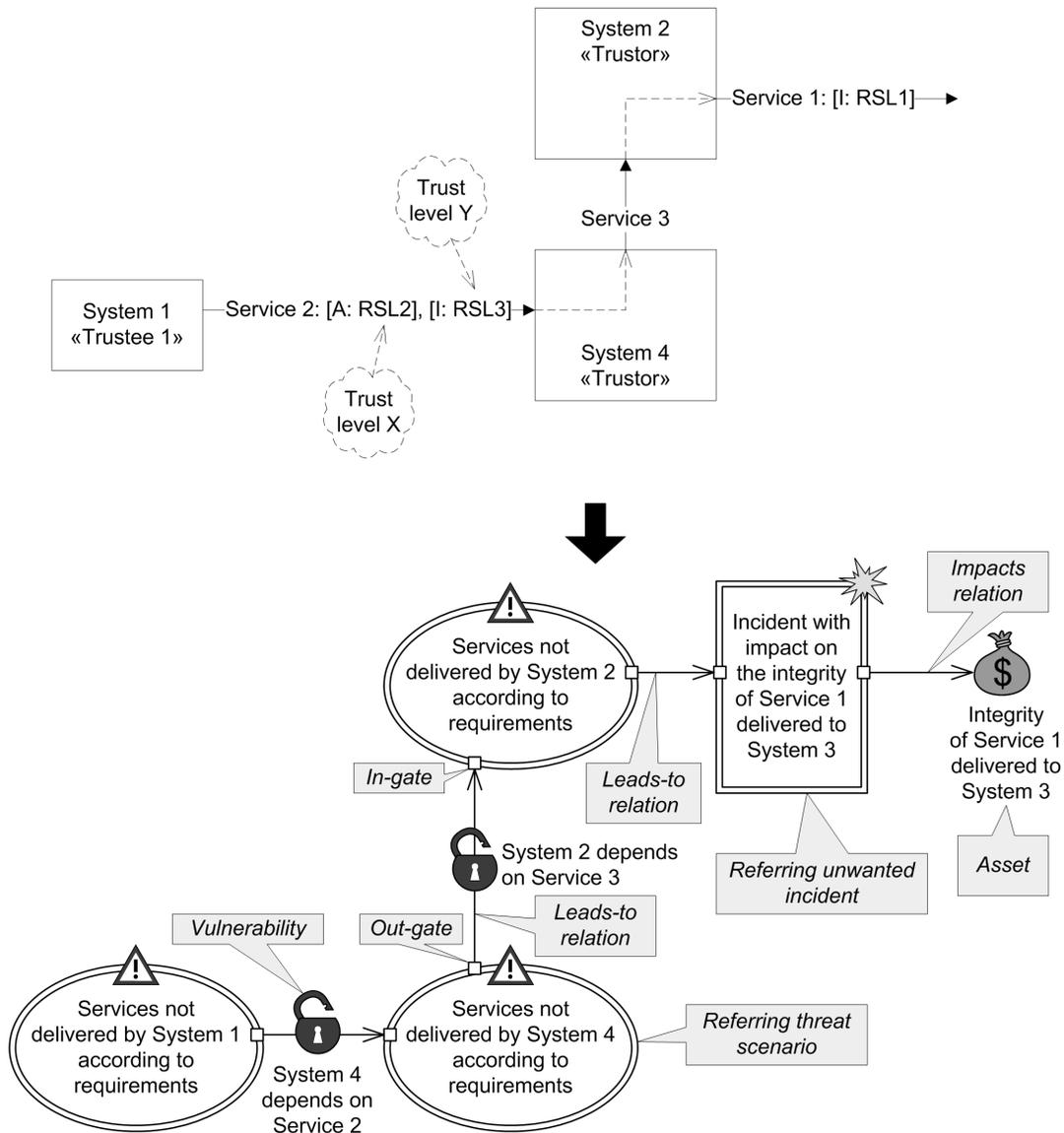
In the following we provide a more detailed description of how to detail the high-level threat diagrams resulting from Step 2.2. Each high-level threat diagram is detailed by:

- Decomposing each referring threat scenario and its high-level in-gates and out-gates by creating a referenced threat scenario. The referenced threat scenario describes the internal threat behavior of the system represented by the referring threat scenario.

For target systems the internal threat behavior describes how the failures of other systems to deliver their services according to requirements may affect the ability of the target system to deliver its services according to requirements. It also describes how internal failures of the target system may affect its ability to deliver services according to requirements, if the referring threat scenario may lead to other referring threat scenarios. The decomposition also results in likelihood values being assigned to the in-gates and out-gates of the referring and referenced threat scenario.

Using Indicators to Monitor Security Risk in Systems of Systems

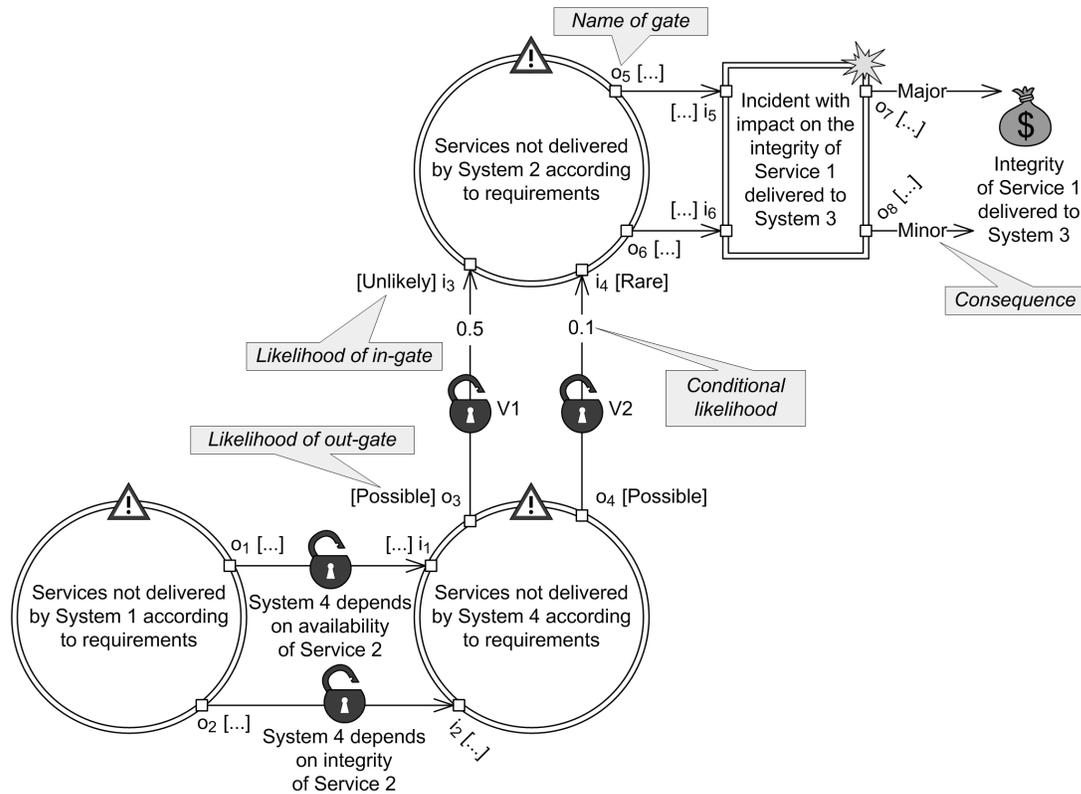
Figure 7. Threat diagram, constructed schematically from an excerpt of the system model in Figure 6, which provides a high-level outline of the impact of service dependencies on the security asset “Integrity of Service 1 delivered to System 3”



For an environment system we only provide a high-level description of its internal behavior. One or two threat scenarios are used to describe this behavior. Two scenarios are used if the environment system is not explicitly required to fulfill all the requirements

associated with its provided services. The first threat scenario represents the failure of the system to deliver its services according to requirements that it needs to fulfill, while the second represents the failure of the system to deliver its services according to

Figure 8. Threat diagram that shows part of the result of detailing the threat diagram in Figure 7



- requirements that the system are not explicitly required to fulfill. For the latter we are referring to requirements to services that are not part of the contract, e.g., service-level agreement, between the environment system and the target systems, but that are of importance to the trustor controlling and operating the target systems.
- Decomposing each referring unwanted incident and its high-level in-gates and out-gates by creating a referenced unwanted incident. Each referenced unwanted incident documents different unwanted incidents that may arise due to the service dependencies. The decomposition also results in likelihood values being assigned to the in-gates and out-gates of the re-

- ferring and referenced unwanted incident. We also decompose the impacts relation of the referring unwanted incident, and we assign consequence values to the new impacts relations.
- Decomposing high-level vulnerabilities assigned to leads-to relations between referring threat scenarios representing target systems. A conditional likelihood is assigned to each leads-to relation for which a vulnerability resulting from the decomposition of a high-level vulnerability has been assigned.
- Decomposing high-level vulnerabilities assigned to leads-to relations where the source of the relation is a referring threat scenario representing an environment system, while the tar-

get is a referring threat scenario representing a target system. The high-level vulnerability is decomposed into as many vulnerabilities as there are required service levels associated with the service represented by the high-level vulnerability. Each of the vulnerabilities resulting from the decomposition specifies that the target system depends on the particular area of service scope associated with the required service level. To estimate the likelihood of the required service level not being delivered, we first calculate the worst-case service level of the particular area of service scope. The worst-case service level specifies our minimum expectation to the particular area of service scope. It is calculated based on the required service level and the trust level calculated in Step 1.3. The likelihood is then estimated based on the difference between the required service level and the worst case service level. The estimated likelihood values are assigned to out-gates of the referring and referenced threat scenarios representing the environment systems.

As part of this step, we also specify scales for measuring likelihood and consequence, and functions for calculating risk values. The risk functions are used after we have created the detailed threat diagrams to determine the risk values of the different risks to security of provided services. A risk value is determined based on the likelihood of an unwanted incident and its consequence with respect to a security asset.

- **Input Documentation:** The high-level threat diagrams from Step 2.2.
- **Output Documentation:**

- Detailed threat diagrams documenting the impact of service dependencies on the security assets.
- Worst-case service levels.
- Scales for measuring likelihood and consequence.
- Risk functions for calculating risk values.
- A list of risks to security of provided services.

Step 3: Identify Indicators for System of Systems

Step 3.1: Identify Security Risks to be Monitored

- **Objective:** Identify the risks to security of provided services that should be monitored.
- **Rationale:** A risk analysis will often result in a number of identified risks to security of provided services. We need to identify the security risks that should be monitored, since it is often not in the trustor's interest to monitor all the security risks. Moreover, there may be security risks for which monitoring is not feasible.
- **How Conducted:** For each security risk resulting from Step 2.3, we must decide whether it should be monitored. Typically, a risk to security of provided services is selected for monitoring if it is believed that the likelihood and/or consequence value determining its risk value is likely to change in a manner that will considerably harm the trustor. A security risk may also be selected for monitoring if we are uncertain about the risk value.
- **Input Documentation:**
 - The detailed threat diagrams from Step 2.3.
 - The list of risks to security of provided services from Step 2.3.

- **Output Documentation:** A list of risks to security of provided services to be monitored.

Step 3.2: Identify Relevant Indicators for the Security Risks to be Monitored

- **Objective:** Identify relevant indicators for monitoring the risk values of the security risks to be monitored.
- **Rationale:** To monitor changes in risk values we need to identify indicators. The indicators are calculated from measurable properties of the SoS.
- **How Conducted:** For the security risks identified to be monitored in Step 3.1, we identify relevant indicators. Indicators for monitoring consequence are related to impacts relations between unwanted incidents and security assets. On the other hand, indicators for monitoring likelihood may not only be related to unwanted incidents, but also to vulnerabilities and threat scenarios leading up to an incident, since the likelihoods of vulnerabilities being exploited and threat scenarios occurring will affect the likelihood of the unwanted incident occurring.
Basic indicators are identified for the different likelihood and consequence values to be monitored. If more than one basic indicator is needed for monitoring a consequence or likelihood value, then a composite indicator, aggregating the basic indicators, is also identified.
- **Input Documentation:**
 - The list of risks to security of provided services to be monitored from Step 3.1.
 - The detailed threat diagrams from Step 2.3.
- **Output Documentation:** A set of relevant basic and composite indicators for monitoring likelihood and consequence.

Step 4: Specify Design and Deployment of Identified Indicators for System of Systems

Step 4.1: Specify Design of Indicators for Monitoring Security Risks

- **Objective:** Specify how basic and composite indicators for monitoring likelihood and consequence values should be designed.
- **Rationale:** We need to specify how the identified basic and composite indicators from Step 3.2 should be designed, i.e., how they should be calculated, in order to be useful for monitoring.
- **How conducted:** A design specification, in the form of an algorithm, is provided for each indicator identified in Step 3.2. It specifies the data needed for calculating the indicator, how the indicator should be calculated, and the output from the calculation. Assuming the likelihood and consequence intervals obtained in Step 2.3 are correct, the algorithm should yield likelihoods and consequences in these intervals when applied to the basic indicator values at the time these intervals were determined.
- **Input Documentation:**
 - The list of risks to security of provided services to be monitored from Step 3.1.
 - The relevant indicators identified in Step 3.2.
 - The detailed threat diagrams from Step 2.3.
 - Basic indicator values from the time when the detailed threat diagrams were constructed.
- **Output documentation:** A design specification for each indicator identified in Step 3.2.

Step 4.2: Specify Deployment of Indicators for Monitoring Security Risks

- **Objective:** Specify how basic and composite indicators for monitoring likelihood and consequence values should be deployed in the SoS.
- **Rationale:** We need to specify how the identified basic and composite indicators from Step 3.2 should be deployed in the SoS, i.e., how the data needed to calculate the different indicators is extracted and transmitted within the SoS, in order to be useful for monitoring.
- **How Conducted:** A deployment specification is provided for each indicator identified in Step 3.2. It specifies how the data used by the design specification is extracted and transmitted within the SoS.
- **Input Documentation:** The design specifications from Step 4.1.
- **Output Documentation:** A deployment specification for each indicator.

DEMONSTRATION OF STEP 1: DOCUMENT SYSTEM OF SYSTEMS

We consider an SoS consisting of an electrical power production infrastructure (EPP), a public telecom infrastructure (PTI), and an electrical power grid (EPG). In the following we assume that we have been hired by the company in charge of the electrical power production infrastructure, Client EPP, to help capture and monitor the impact of service dependencies on the security of its provided services.

Step 1.1: Model System of Systems

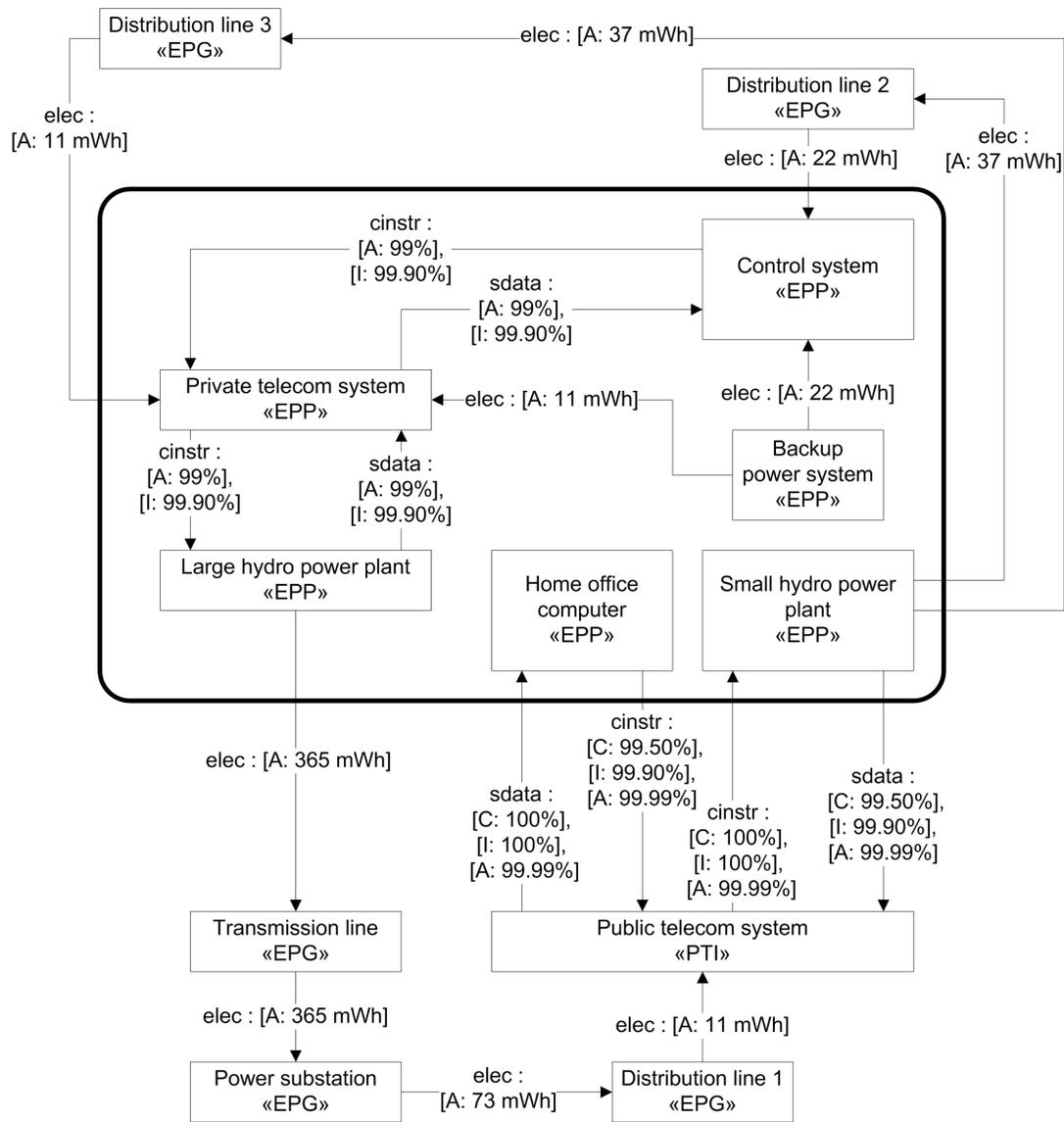
Figure 9 documents the electrical power production infrastructure and its environment. The different systems provide and/or require electricity

(*elec*), control instructions (*cintr*), and sensor data (*sdata*). All the services with the exception of electricity service are data services. The required service levels for the electricity services state the amount of electricity that needs to be delivered (availability) in the period of one year. For the data services, the required service levels (also for the period of one year) state the percentages of sensor data/control instructions messages that needs to be delivered (availability), be confidential, and have integrity.

In the electrical power production infrastructure there is a “Large hydro power plant.” The electrical power produced by this plant is transmitted on a high-voltage “Transmission line” to a “Power substation.” Here, the power is transformed to low-voltage power by a transformer, before being distributed to its end-users by distribution lines. “Distribution line 1” provides electrical power to the “Public telecom system.” The infrastructure also consists of a “Small hydro power plant.” This power plant distributes power directly to its end-users by the use of “Distribution line 2” and “Distribution line 3.” “Private telecom system” and “Control system,” both located within the electrical power production infrastructure, are two of the end-users that receive electrical power from these two distribution lines. These two systems share a “Backup power system,” which is used when the electrical power grid fails to provide electricity to one or both systems.

The “Control system” is used to operate the “Large hydro power plant.” By the use of the “Private telecom system” it sends control instructions to the plant, while sensors at the plant send data to the “Control system” through the same telecom system. The “Control system” responds to errors arising at the plant. If it cannot resolve the errors, it will shut down the plant to protect equipment. If the connection to the “Control system” is lost, the plant will automatically shut down if it cannot resolve errors by itself. The required service level with respect to availability is 99% for all the data services exchanged between the “Control system”

Figure 9. System model for electrical power production infrastructure and its environment



and the “Large hydro power plant,” since the plant has some ability of operating independently of the “Control system.” Moreover, the required service level with respect to integrity is 99.90% for all the data services.

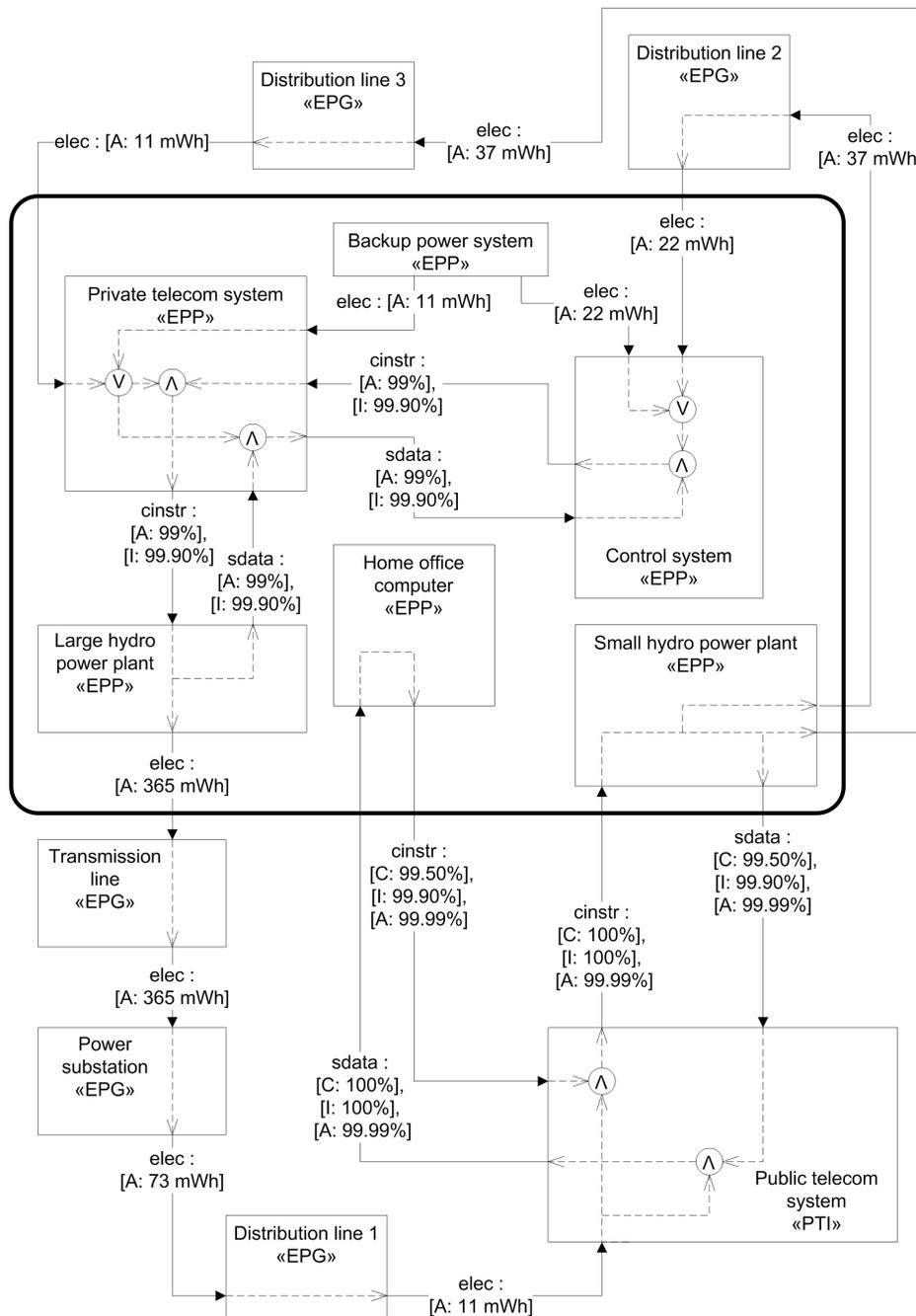
Due to its size, the “Small hydro power plant” is operated by a system operator from his “Home office computer.” The operator uses a computer that is dedicated to this task. He sends encrypted control instructions to the plant through the “Public

telecom system,” while the sensors at the plant sends encrypted data to the operator through the same telecom system. The system operator responds to errors arising at the plant. If he cannot resolve the errors, he will shut down the plant to protect equipment. If the connection to the “Public telecom system” is lost, the plant will automatically shut down to protect equipment. This is done as a precautionary step, since the plant is not able to resolve errors by itself. Since the availability of

the data services exchanged between the “Small hydro power plant” and the “Home office computer” are crucial for the operation of the “Small hydro power plant,” the required service level for

all the data services with respect to availability is 99.99%. It should be noticed that the required service levels for integrity and confidentiality for data services provided by “Public telecom system”

Figure 10. System model annotated with service dependencies



to “Home office computer” and “Small hydro power plant” do not specify explicit requirements that “Public telecom system” needs to fulfill when providing the data services. It is more correct to say that these requirements are to the data messages themselves. As can be seen in Figure 9, Client EPP requires that the confidentiality and integrity of data messages should not be changed while at “Public telecom system” or during transmission to its destinations.

Step 1.2: Capture Service Dependencies

In Figure 10, the system model in Figure 9 is annotated with the service dependencies. Most of the service dependencies are self-explanatory, but note especially that “Small hydro power plant” depends on the availability of control instructions, provided by “Home office computer,” to produce electricity. The “Large hydro power plant” is less dependent on control instructions than the “Small hydro power plant,” but since it depends on control instructions in situations where it cannot resolve errors, there is a dependency between the required control instructions service and the electricity service provided to “Transmission line.” It should also be noticed that both “Private telecom system” and “Control system” can require electricity from the “Backup power system” if the electrical power grid fails to provide electricity, and that incoming sensor data messages may affect the outgoing control instructions messages, and vice versa. The dependencies between incoming and outgoing messages are a result of control instructions messages often being created based on the incoming sensor data messages, and that control instructions messages affect the operation of “Small hydro power plant” and its data sensors, which again affect the outgoing sensor data messages.

Step 1.3: Capture Trust Relations

In Figure 11, the system model in Figure 10 is annotated with trust relations. As can be seen in the figure, trust levels have been assigned to the required service levels for those services that are provided by systems of the environment to systems of the target.

All the services for which trust levels should be assigned are considered very reliable by Client EPP. Thus, it is expected that they should achieve their required service levels. Even so, Client EPP is aware that the services can fail. After having considered both the high reliability of the services and the possibility of service failures, Client EPP assigns high trust levels to the different required service levels.

For the control instructions services provided by “Public telecom system” to “Small hydro power plant,” Client EPP has a trust of:

- 0.95 in that the control instructions messages are delivered according to the confidentiality requirement;
- 0.90 in that the control instructions messages are delivered according to the integrity requirement; and
- 0.99 in that the control instructions messages are delivered according to the availability requirement.

DEMONSTRATION OF STEP 2: ANALYZE THE IMPACT OF SERVICE DEPENDENCIES ON RISK TO SECURITY OF PROVIDED SERVICES

Step 2.1: Identify Security Assets

For the sake of simplicity, we demonstrate the method by only identifying security assets for one of the provided services. In an industrial case we would consider all the provided services when identifying security assets.

Figure 11. System model annotated with trust relations

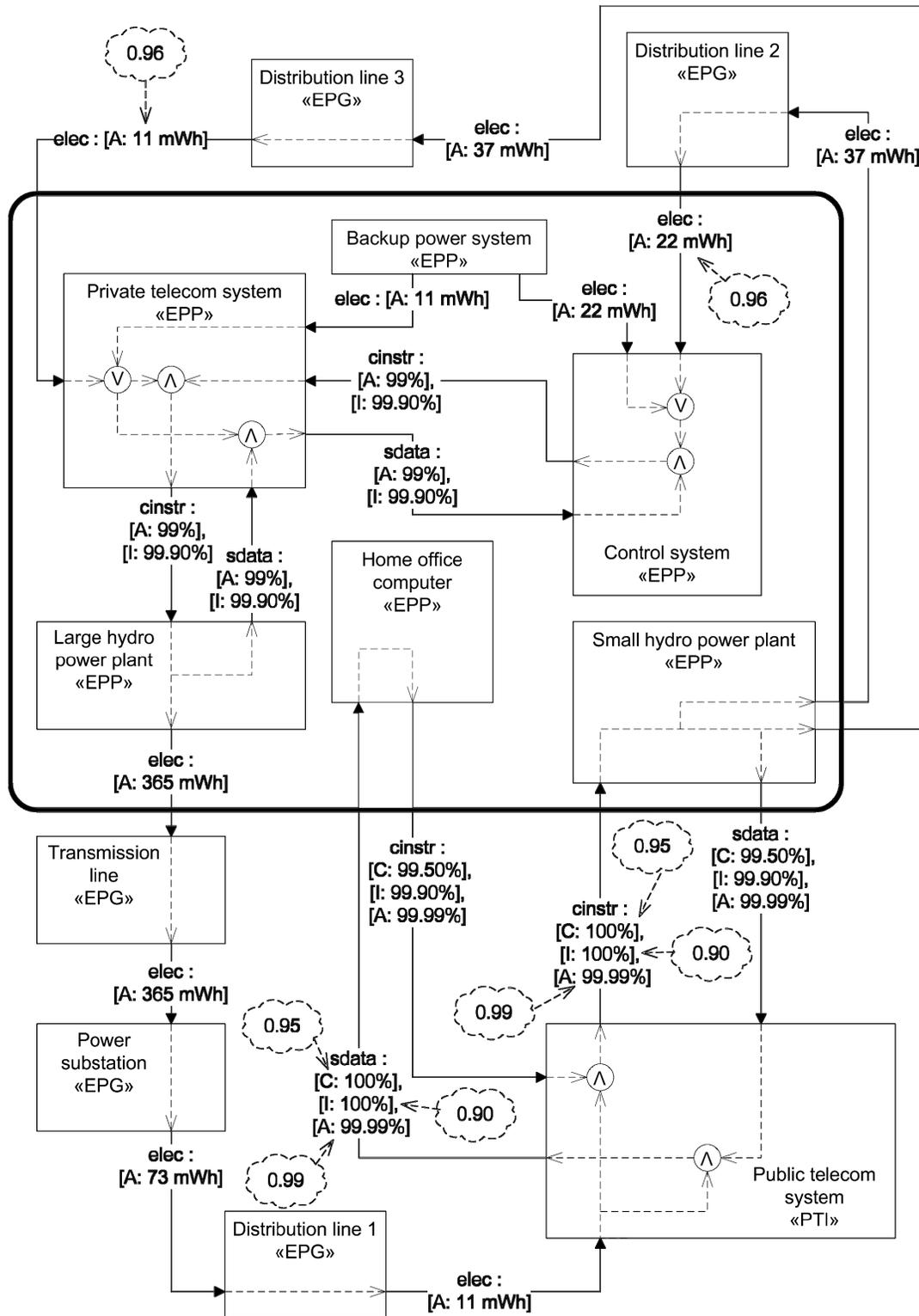
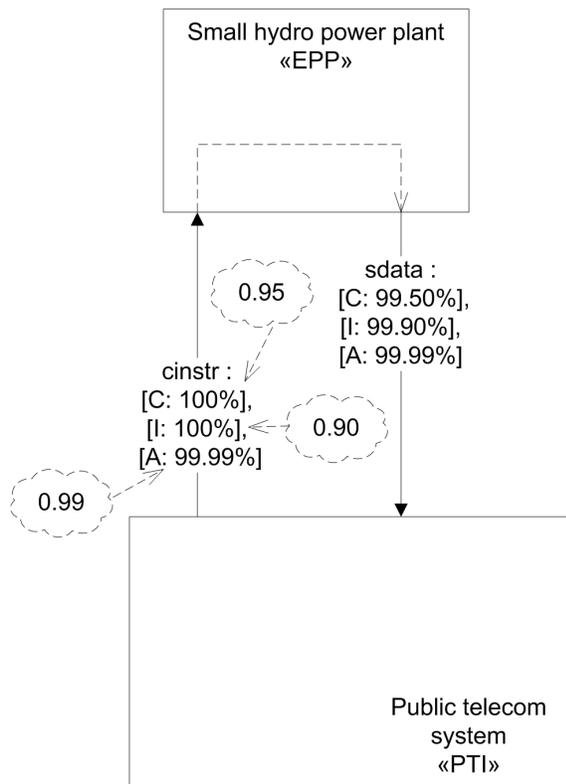


Figure 12. Excerpt, from the system model in Figure 11, for schematic construction of the threat diagram for the sensor data service provided to “Public telecom system”



A concern of Client EPP is that services dependencies in the SoS may affect the ability of “Small hydro power plant” to provide the sensor data service according to the security requirements associated with the service. If this service is affected, then the ability of “Home office computer” to control and operate the “Small hydro power plant” may be affected as well, which again may impact the electricity services provided to “Distribution line 2” and “Distribution line 3.” Client EPP therefore seeks to protect the security assets “Confidentiality of sensor data delivered to Public telecom system,” “Integrity of sensor data delivered to Public telecom system,” and “Availability of sensor data delivered to Public

telecom system,” and wants to identify the service dependencies’ impact on these security assets.

Step 2.2: Construct High-level Threat Diagrams of the Impact of Service Dependencies on Identified Security Assets

For the sensor data service provided to “Public telecom system,” the excerpt in Figure 12 of the system model in Figure 11 and the threat diagram in Figure 13 have been created by following the schematic procedure. The threat diagram provides a high-level description of the impact of service dependencies on the security of the sensor data service provided to “Public telecom system.”

Step 2.3: Construct Detailed Threat Diagrams of the Impact of Service Dependencies on Identified Security Assets

Before we perform the detailed security risk analysis of how target systems may fail to provide services according to requirements, we need to establish how to measure likelihood and consequence, as well as defining the risk function. Table 1 shows how likelihood is measured, while Table 2 shows how consequence is measured for the different security assets. To calculate the number of sensor data messages not sent, sent with lack of confidentiality, or sent with lack of integrity, it is helpful to have an estimate of the number of sensor data messages sent from “Small hydro power plant” in the period of one year. Client EPP estimates this number to be 5000.

For all risks, the risk is classified as acceptable or unacceptable as shown in Exhibit 1.

Here, the *Maximum service level* is the highest achievable service level for the area of service scope associated with the security asset in question. For example, the highest achievable service level for the integrity of the sensor data service is 5000. This means that all the 5000 sensor data

Figure 13. Threat diagram constructed schematically from the excerpt in Figure 12

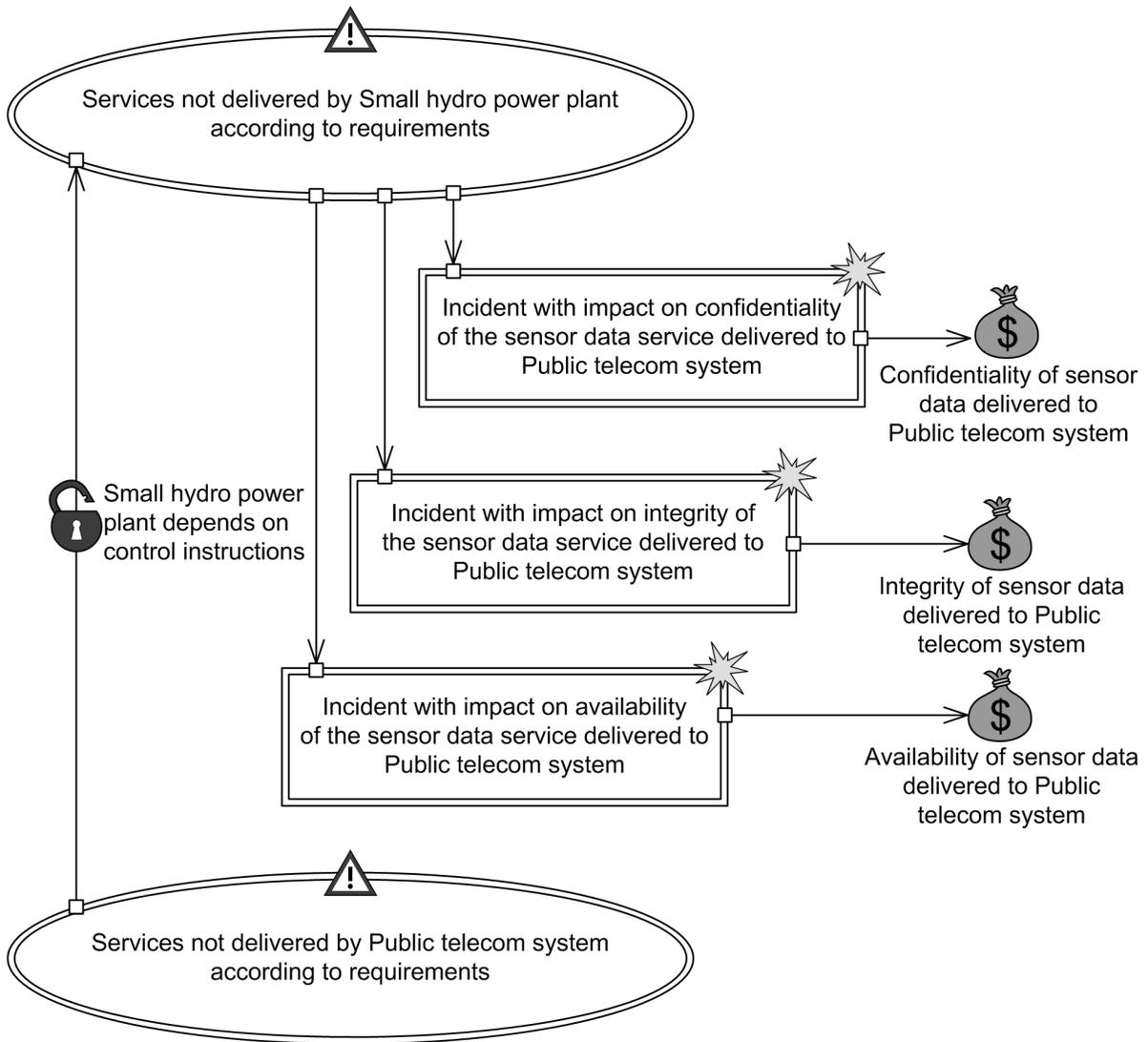


Table 1. Likelihood scale

Likelihood	Description
Certain	Fifty times or more per year [500, ∞): 10 years
Very likely	Ten to fifty times per year [100, 499]: 10 years
Likely	Five times to ten times per year [50, 99]: 10 years
Possible	Two to five times per year [20, 49]: 10 years
Unlikely	Once a year [6, 19]: 10 years
Very unlikely	Less than once per year [2, 5]: 10 years
Rare	Less than once per ten years [0, 1]: 10 years

Table 2. How consequence is measured for the three security assets

Availability of sensor data delivered to Public telecom system
Number of sensor data messages not sent
Confidentiality of sensor data delivered to Public telecom system
Number of sensor data messages sent with lack of confidentiality
Integrity of sensor data delivered to Public telecom system
Number of sensor data messages sent with lack of integrity

Exhibit 1.

```
Expected service level = Maximum service level - (Likelihood · Consequence) (1)
```

```
if Expected service level ≥ Required service level then (2)
    Risk value = Acceptable
else
    Risk value = Unacceptable
endif
```

messages sent during the period of one year have integrity. A risk associated with a security asset is *Unacceptable* if the *Expected service level* is less than the *Required service level*.

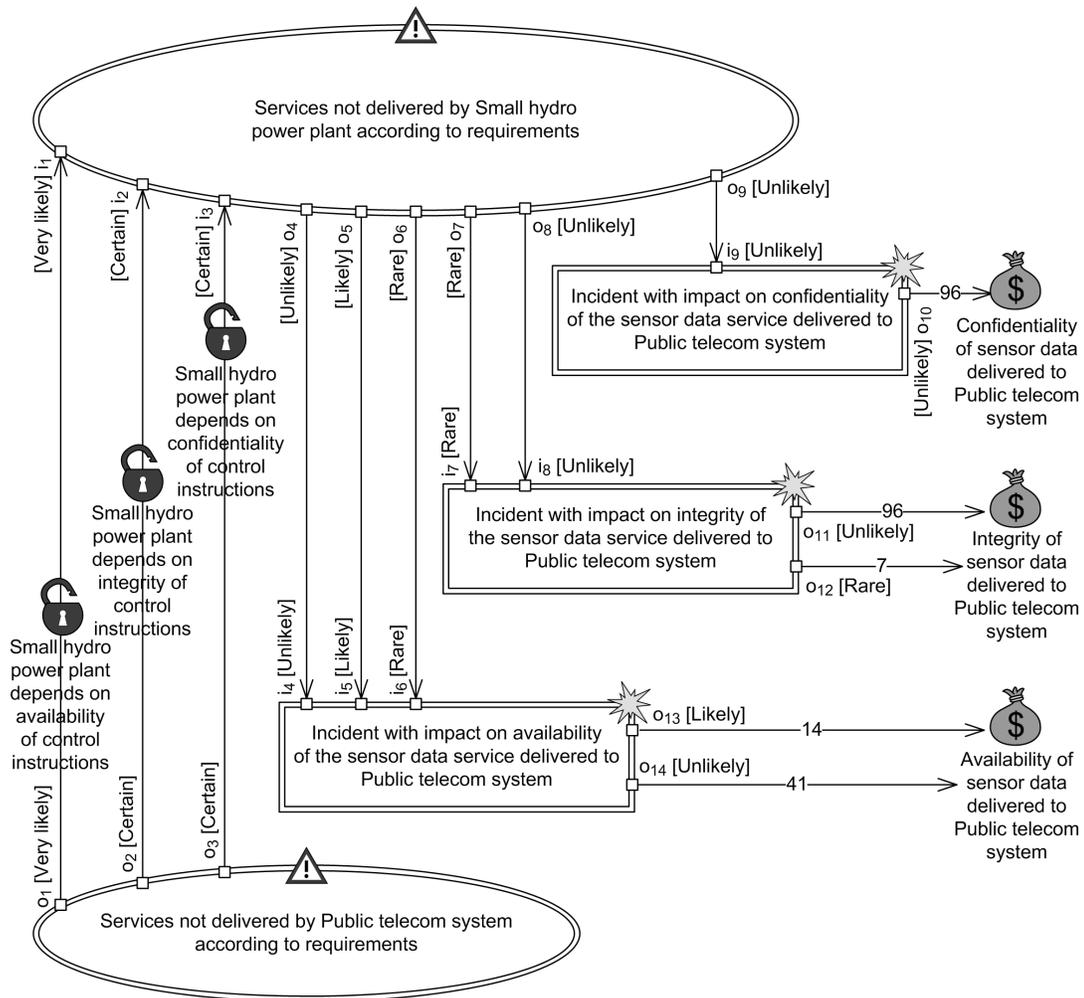
In Figure 14 is the detailed version of the high-level threat diagram in Figure 13. The referring elements in the diagram refer to the referenced threat scenarios provided in Figures 15 and 16, and the referenced unwanted incidents provided in Figure 20. Moreover, the referenced threat scenario in Figure 16 contains three referring threat scenarios, which refer to the referenced threat scenarios provided in Figures 17-19. Client EPP has estimated all the likelihood and consequence values in the different figures.

We refer to i_x and o_y of the referring threat scenarios and unwanted incidents as in-gate and out-gate, respectively. Relations to an element inside a referenced threat scenario must go through an in-gate, while relations to an element outside the referenced threat scenario must go through an out-gate. The likelihood value of an in-gate i_x documents the contribution of an element outside the referenced threat scenario via gate i_x to the likelihood of an element inside the referenced threat scenario, while the likelihood of the out-gate o_y documents the contribution of the likelihood of an element inside the referenced threat scenario via gate o_y to the likelihood of an element outside the referenced threat scenario.

Below we provide some examples of the semantics of elements and relations in the different figures. For more information on the semantics of the CORAS language, see (Lund et al., 2010).

- **Threat Scenario:** Threat scenario “Control instructions message is not delivered” occurs with likelihood “Very likely” (Figure 17).
- **Leads-To Relation (with Conditional Likelihood):** “Control instructions message is not delivered” leads to “Missing control instructions are needed for correcting a serious error” with conditional likelihood “0.5” (Figure 17).
- **Leads-To Relation (with Vulnerability):** “Control instructions message with lack of integrity is delivered” leads to “Retransmission of control instructions message is not requested” with undefined conditional likelihood, due to vulnerability “Possible that checksum algorithm fails to detect integrity violations” (Figure 18).
- **In-Gate (with Likelihood):** i_l is an in-gate with likelihood “Very likely” (Figure 14).
- **Out-Gate (with Likelihood):** o_l is an out-gate with likelihood “Very likely” (Figure 14).
- **Leads-to Relations (Between Elements of Referenced Threat Scenarios):** “Services not delivered by Public telecom system according to requirements that Public telecom system are required to fulfill” leads to “Control instructions message is not delivered” via gates o_l , i_l , and i_{l0} , due to vulnerability “Small hydro power plant depends on availability of control instructions” (Figures 14, 15, 16, and 17).
- **Unwanted Incident:** Unwanted incident “Confidential sensor data is sent in

Figure 14. Detailed version of the high-level threat diagram in Figure 13



plain text from Small hydro power plant to an outsider” occurs with likelihood “Unlikely” (Figure 20).

- **Impacts Relation (Between Element of Referenced Unwanted Incident and Asset):** “Confidential sensor data is sent in plain text from Small hydro power plant to an outsider” impacts “Confidentiality of sensor data delivered to Public telecom system” via gate o_{10} with consequence “96” (Figures 14 and 20).

As can be seen in Figure 14, the vulnerability “Small hydro power plant depends on control instructions” in Figure 13 has been decomposed into three vulnerabilities. “Public telecom system” is only required to deliver the control instructions service according to the availability requirement. The referenced threat scenario in Figure 15 is therefore made up of two threat scenarios, since “Public telecom system” is not explicitly required to fulfill the confidentiality and integrity requirements of the control instructions service.

Client EPP estimates the number of control instructions messages delivered each year to

Figure 15. The referenced threat scenario “Services not delivered by Public telecom system according to requirements,” referred to in Figure 14

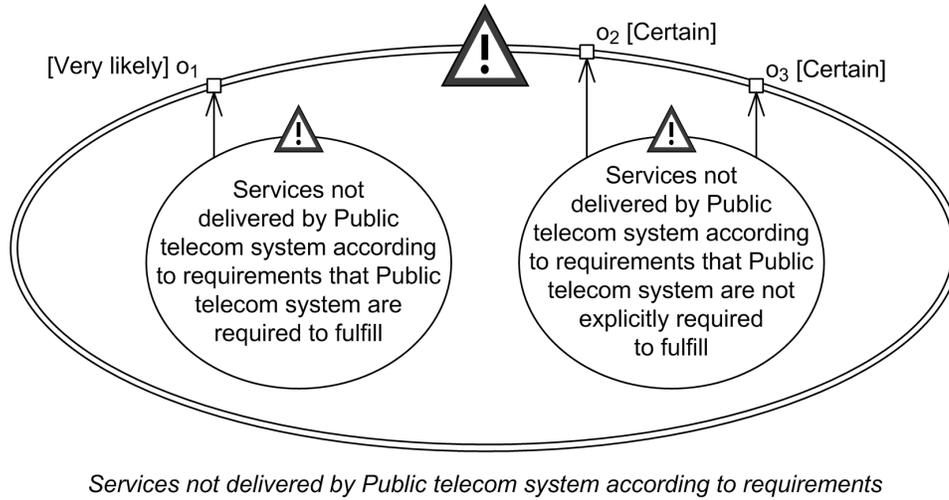


Figure 16. The referenced threat scenario “Services not delivered by Small hydro power plant according to requirements,” referred to in Figure 14

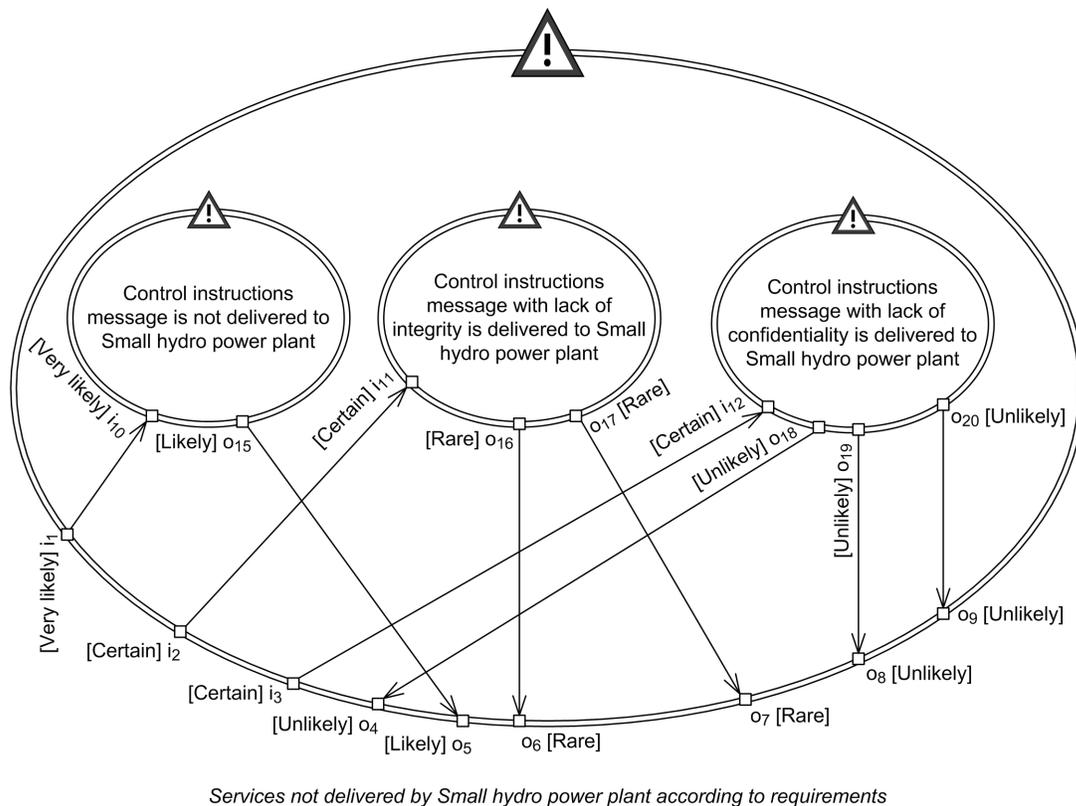
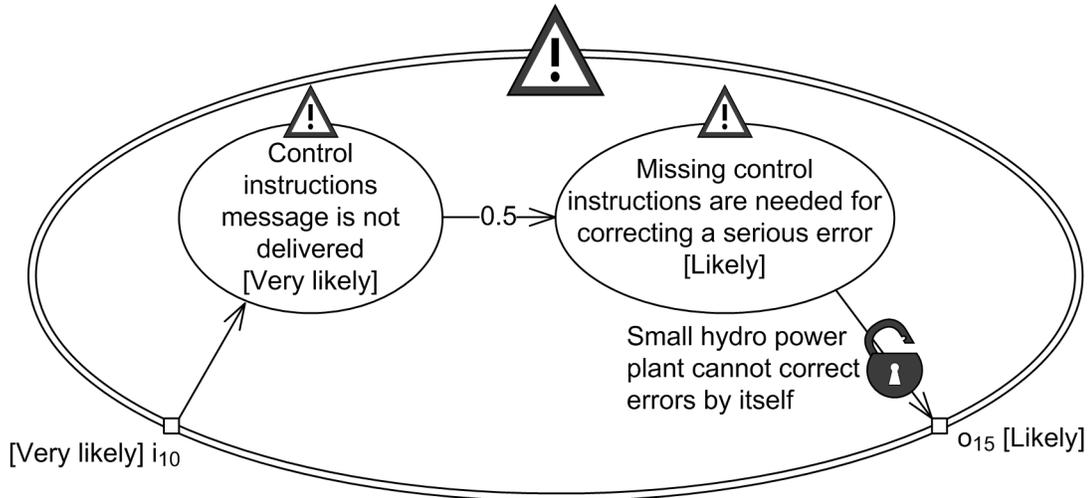
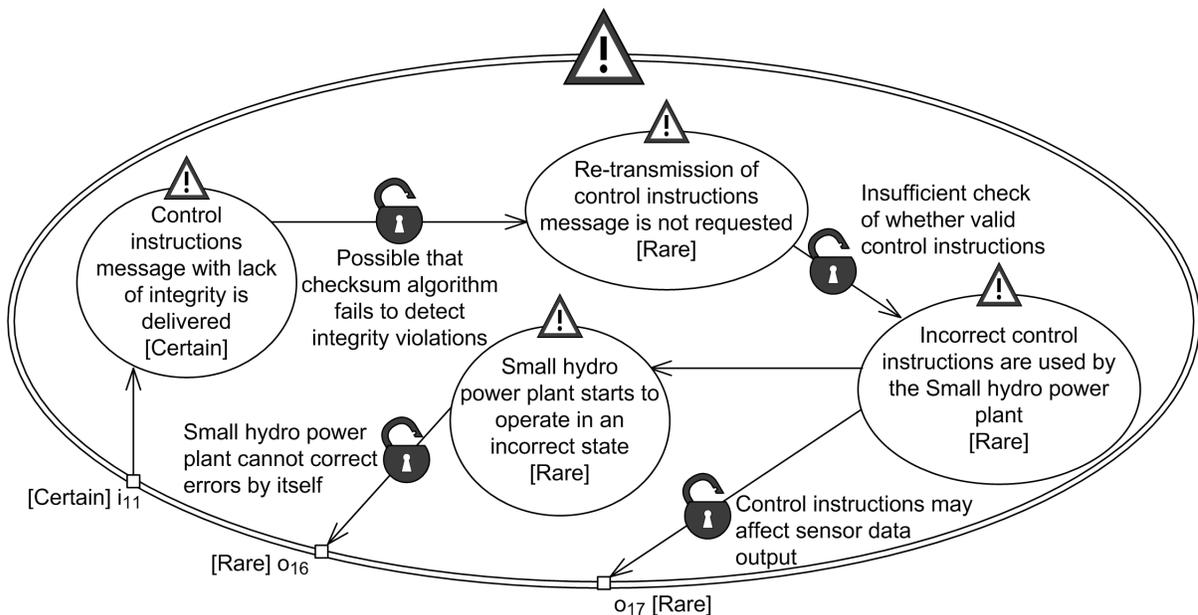


Figure 17. The referenced threat scenario “Control instructions message is not delivered to Small hydro power plant,” referred to in Figure 16



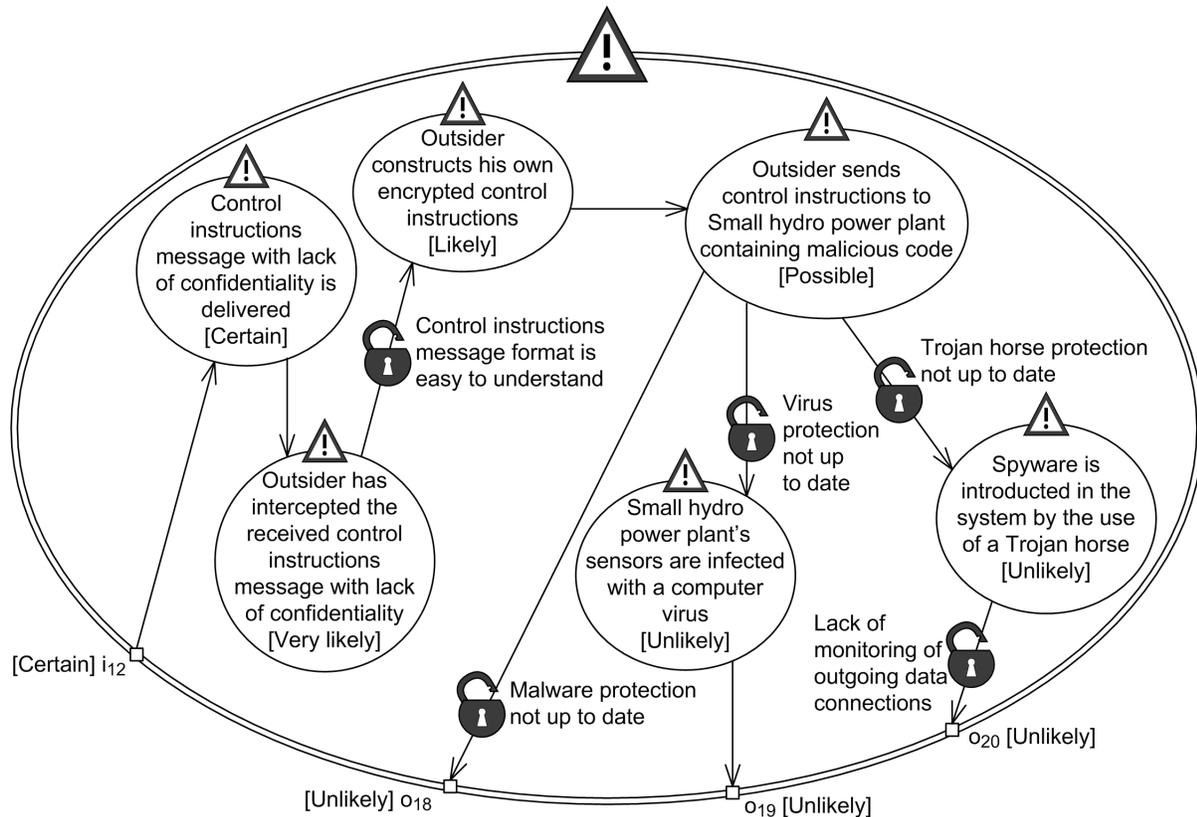
Control instructions message is not delivered to Small hydro power plant

Figure 18. The referenced threat scenario “Control instructions message with lack of integrity is delivered to Small hydro power plant,” referred to in Figure 16



Control instructions message with lack of integrity is delivered to Small hydro power plant

Figure 19. The referenced threat scenario “Control instructions message with lack of confidentiality is delivered to Small hydro power plant,” referred to in Figure 16



Control instructions message with lack of confidentiality is delivered to Small hydro power plant

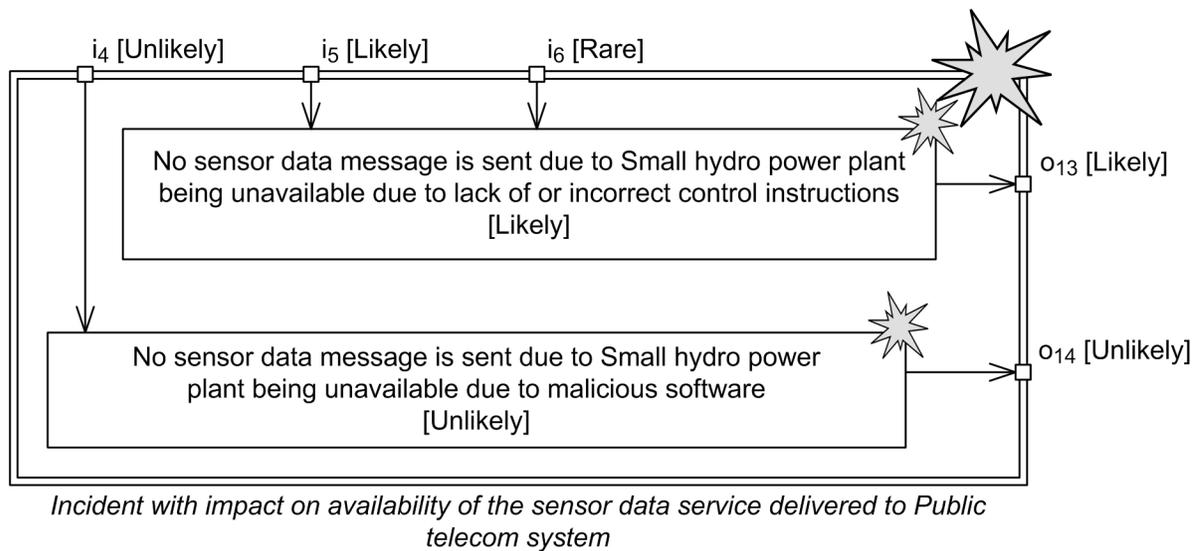
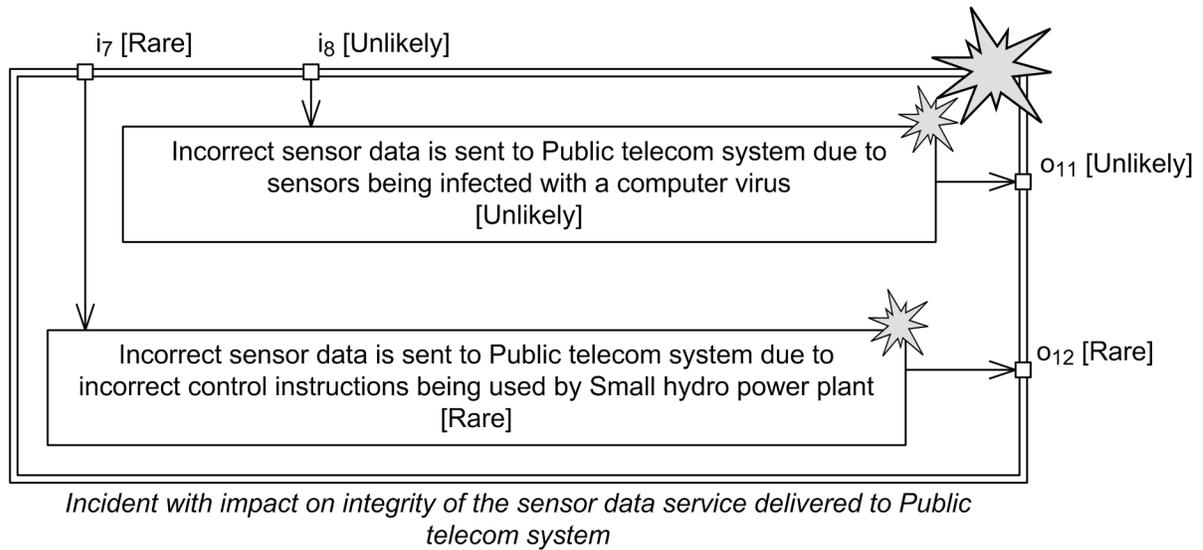
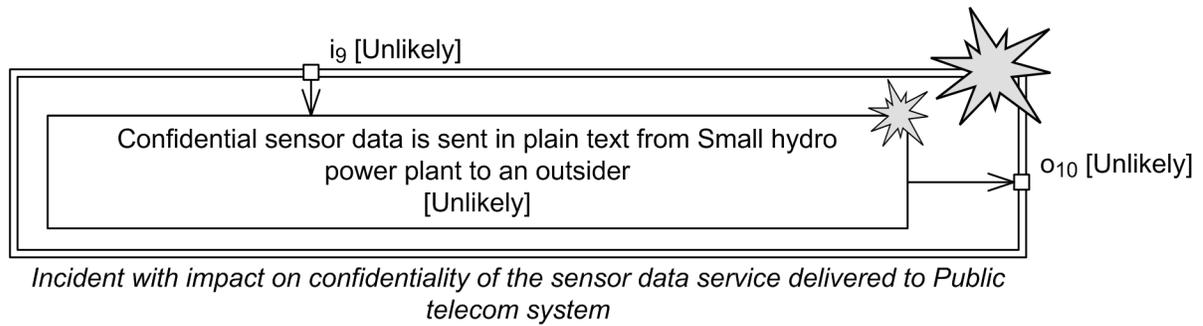
“Small hydro power plant” to be 1000. Before we can estimate the likelihoods of the control instructions service not being delivered according to the confidentiality, integrity, and availability requirements, we need to calculate the worst-case service levels of the control instructions service delivered by “Public telecom system.” These are as follows:

- 95% (100% · 0.95) of the delivered control instructions messages are confidential.
- 90% (100% · 0.90) of the delivered control instructions messages have integrity.
- 98.99% (99.99% · 0.99) of the control instructions messages are delivered.

To estimate the likelihoods we use the estimated number of control instructions messages sent each year in combination with the required and worst-case service levels of the control instructions service delivered by “Public telecom system.” According to the required service levels, 999.9 (99.99% of 1000), 1000 (100% of 1000), and 1000 (100% of 1000) of the control instructions messages should be delivered, should be confidential, and should have integrity, respectively. On the other hand, our expectations according to the worst-case service levels are: that 990 (999.9 · 0.9899) out of the 999.9 required control instructions messages are delivered; that 950 (1000 · 0.95) out of the 1000 required control instructions messages are confidential; and that 900 (1000 ·

Using Indicators to Monitor Security Risk in Systems of Systems

Figure 20. The referenced unwanted incidents “Incident with impact on confidentiality of the sensor data service delivered to Public telecom system,” “Incident with impact on integrity of the sensor data service delivered to Public telecom system,” and “Incident with impact on availability of the sensor data service delivered to Public telecom system,” referred to in Figure 14



0.90) out of the 1000 required control instructions messages have integrity. Thus, the likelihood of control instructions messages not being delivered according to the availability requirement is “Very likely” ($999.9 - 990 \approx 10$ per year), while the likelihoods of control instructions messages not being delivered according to the confidentiality and integrity requirements are “Certain” ($1000 - 950 = 50$ per year) and “Certain” ($1000 - 900 = 100$ per year), respectively.

The referenced threat scenario “Services not delivered by Small hydro power plant according to requirements” is given in Figure 16. The internal threat behavior of “Small hydro power plant” is described by the referenced threat scenarios in Figures 17-19. The different referenced threat scenarios describe how “Small hydro power plant” may fail to provide the sensor data service according to its requirements as a result of “Public telecom system” failing to provide the control instructions service according to requirements.

Figure 20 contains the referenced unwanted incidents referred to in Figure 14. For each of the unwanted incidents, Client EPP believes that more than one sensor data message is affected by the incident. For the incident “No sensor data message is sent due to Small hydro power plant being unavailable due to lack of or incorrect control instructions,” Client EPP estimates a down time of one day, while a down time of 3 days is estimated for the incident “No sensor data message is sent due to Small hydro power plant being unavailable due to malicious software.” For the incident “Incorrect sensor data is sent to Public telecom system due to incorrect control instructions being used by Small hydro power plant,” Client EPP estimates that “Small hydro power plant” sends incorrect sensor data messages for a period of 12 hours as a result of using incorrect control instructions. For the incident “Confidential sensor data is sent in plain text from Small hydro power plant to an outsider,” Client EPP believes that this can go on

undetected for at much as seven days. The same is believed for the incident “Incorrect sensor data is sent to Public telecom system due to sensors being infected with a computer virus.” With an average number of 13.7 sensor data messages being sent each day, we get the consequence values documented in Figure 14.

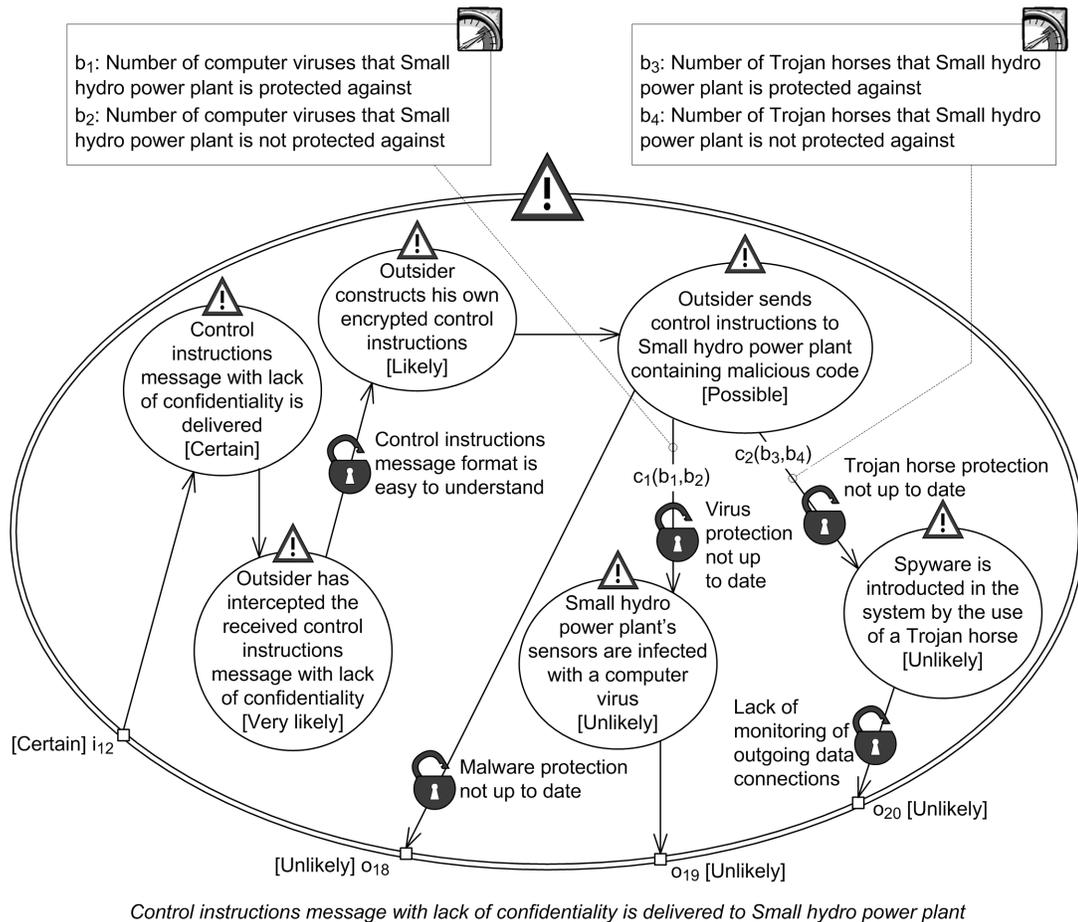
The result of the detailed analysis is five security risks, where each risk consists of an unwanted incident, its likelihood of occurring, and the consequence of the unwanted incident with respect to a security asset. Based on the risk function, defined in eq. (1) and eq. (2), the estimated number of sensor data messages sent each year (5000), and the required service levels for the sensor data service, we can calculate the risk values of the five security risks. All the security risks, with the exception of “Incorrect sensor data is sent to Public telecom system due to incorrect control instructions being used by Small hydro power plant,” are unacceptable. For the acceptable risk the expected service level (4999.3 of the sent sensor data messages have integrity) is higher than the required service level (4995 of the sent sensor data messages have integrity).

DEMONSTRATION OF STEP 3: IDENTIFY INDICATORS FOR SYSTEM OF SYSTEMS

Step 3.1: Identify Security Risks to be Monitored

Client EPP believes that the likelihood values used to calculate the risk values of the risks “Incorrect sensor data is sent to Public telecom system due to sensors being infected with a computer virus” and “Confidential sensor data is sent in plain text from Small hydro power plant to an outsider” may be subject to change. We therefore decide to monitor these security risks.

Figure 21. Relevant indicators, assigned to leads-to relations in the referenced threat scenario in Figure 19, for monitoring the risks “Incorrect sensor data is sent to Public telecom system due to sensors being infected with a computer virus” and “Confidential sensor data is sent in plain text from Small hydro power plant to an outsider”



Step 3.2: Identify Relevant Indicators for the Security Risks to be Monitored

Indicators should be used to monitor likelihood values, since the likelihood values used to calculate the risk values of the two risks may be subject to change. Client EPP does not find it feasible to directly monitor the likelihoods of the unwanted incidents occurring, and has therefore decided to monitor the conditional likelihoods of two leads-to relations in the referenced threat scenario in

Figure 19 that affect the likelihoods of the two unwanted incidents occurring. The relevant indicators for the two leads-to relations are presented in Figure 21. Note that in order to save space we only demonstrate how the conditional likelihoods of these two leads-to relations are monitored based on indicators. In a real-life case we would of course also monitor the resulting likelihoods for the risks identified for monitoring in Step 3.1, which depend on the conditional likelihoods we now address, as well as other factors.

One composite indicator c_1 , which aggregates the two basic indicators b_1 and b_2 , has been identified for one leads-to relation. c_1 makes a prediction about the percentage of computer viruses that “Small hydro power plant” is not protected against. For the other leads-to relation, we have identified the composite indicator c_2 , which aggregates the two basic indicators b_3 and b_4 . c_2 makes a prediction about the percentage of Trojan horses that “Small hydro power plant” is not protected against.

To calculate the indicators, Client EPP relies on data from the security vendor that delivers the security solutions and patches that are used in the control system of “Small hydro power plant.” At the “Small hydro power plant” it may take some time between each upgrade of the security solutions and patching of the control system. This is due to that the updates and patches need to be inspected and tested before they can be introduced into the control system in order to ensure the stability of the control system of “Small hydro power plant.” The consequence is that “Small hydro power plant” may be unprotected for some time against well-known computer viruses and Trojan horses.

DEMONSTRATION OF STEP 4: SPECIFY DESIGN AND DEPLOYMENT OF IDENTIFIED INDICATORS FOR SYSTEM OF SYSTEMS

Step 4.1: Specify Design of Indicators for Monitoring Security Risks

In Figure 21 the composite indicators c_1 and c_2 are associated to one leads-to relation each. No conditional likelihoods were assigned to these leads-to relations during the detailed analysis in Step 2.3. Thus, values are not obtained for the basic indicators from the time when the referenced threat scenario in Figure 19 was constructed.

Table 3. Design specifications, in the form of algorithms, for the basic indicators b_1 and b_2 and the composite indicator c_1

Algorithm for b_1
<p>Input: list₁: “List of names of computer viruses that Small hydro power plant is protected against”</p> <p>$b_1 := 0$ for each name₁ in list₁ do $b_1 := b_1 + 1$ end for each</p> <p>Output: b_1</p>
Algorithm for b_2
<p>Input: list₁: “List of names of computer viruses that Small hydro power plant is protected against,” list₂: “List of names of all computer viruses that the security vendor delivering security solutions and patches to Small hydro power plant offers protection against,” list₃: “List of names of computer viruses that the security vendor delivering security solutions and patches to Small hydro power plant is aware of but does not yet offer protection against”</p> <p>$b_2 := 0$ for each name₂ in list₂ do if name₂ not in list₁ then $b_2 := b_2 + 1$ end if end for each for each name₃ in list₃ do $b_2 := b_2 + 1$ end for each</p> <p>Output: b_2</p>
Algorithm for c_1
<p>Input: b_1, b_2</p> <p>$var_1 := b_2 + ((b_1 + b_2) \cdot [0.001, 0.005])$ $var_2 := b_1 + var_1$ $c_3 := var_1 / var_2$</p> <p>Output: c_1</p>

In Tables 3 and 4 are the design specifications for the different basic and composite indicators. All the specifications have been given in the form of algorithms. The six algorithms are to be used by a risk monitor within the electrical power production infrastructure.

To calculate the two composite indicators, Client EPP takes into account that there may be

Table 4. Design specifications, in the form of algorithms, for the basic indicators b_3 and b_4 and the composite indicator c_2

Algorithm for b_3
<p>Input: list₁: “List of names of Trojan horses that Small hydro power plant is protected against”</p> <p>$b_3 := 0$ for each name₄ in list₁ do $b_3 := b_3 + 1$ end for each</p> <p>Output: b_3</p>
Algorithm for b_4
<p>Input: list₁: “List of names of Trojan horses that Small hydro power plant is protected against,” list₂: “List of names of all Trojan horses that the security vendor delivering security solutions and patches to Small hydro power plant offers protection against,” list₃: “List of names of Trojan horses that the security vendor delivering security solutions and patches to Small hydro power plant is aware of but does not yet offer protection against”</p> <p>$b_4 := 0$ for each name₅ in list₁ do if name₅ not in list₂ then $b_4 := b_4 + 1$ end if end for each for each name₆ in list₃ do $b_4 := b_4 + 1$ end for each</p> <p>Output: b_4</p>
Algorithm for c_2
<p>Input: b_3, b_4</p> <p>$var_3 := b_4 + ((b_3 + b_4) \cdot [0.001, 0.003])$ $var_4 := b_3 + var_3$ $c_2 := var_3 / var_4$</p> <p>Output: c_2</p>

computer viruses and Trojan horses that the security vendor is not aware of. Client EPP thinks it is reasonable to assume that the total number of computer viruses is 0.1 – 0.5% higher than the sum $b_1 + b_2$, and that the total number of Trojan horses is 0.1 – 0.3% higher than the sum $b_3 + b_4$. For both composite indicators we end up with an interval.

Table 5. Deployment specifications for the basic indicators b_1 and b_2 and the composite indicator c_1

Deployment specification for b_1
<p>Extraction and transmission of list₁: At the start of each week, an automated ICT process creates the list list₁. The list list₁ is created by extracting the names of all computer viruses that “Small hydro power plant” is protected against from the information security database in the electrical power production infrastructure. The process transmits list₁ to the risk monitor by using the internal data network of the electrical power production infrastructure.</p>
Deployment specification for b_2
<p>Extraction and transmission of list₂: At the start of each week, an automated ICT process updates the information security database in the electrical power production infrastructure. The database is updated by retrieving the names of all new computer viruses that the security vendor of Client EPP offers protection against from the security vendor’s database. After having updated the database, the process creates the list list₂ by extracting the names of all computer viruses that the security vendor offers protection against from the information security database. The process transmits list₂ to the risk monitor by using the internal data network of the electrical power production infrastructure.</p>
<p>Extraction and transmission of list₃: At the start of each week, an automated ICT process updates the information security database in the electrical power production infrastructure. The database is updated by retrieving the names of all new computer viruses that the security vendor of Client EPP is aware of but does not yet offer protection against from the security vendor’s database. After having updated the database, the process creates the list list₃ by extracting the names of all computer viruses that the security vendor is aware of but does not yet offer protection against from the information security database. The process transmits list₃ to the risk monitor by using the internal data network of the electrical power production infrastructure.</p>
Deployment specification for c_1
<p>Extraction of b_1 and b_2: The risk monitor calculates the two basic indicators b_1 and b_2 each week. After having calculated the two basic indicators, the risk monitor uses them to calculate the composite indicator c_1.</p>

Step 4.2: Specify Deployment of Indicators for Monitoring Security Risks

In Tables 5 and 6 are the deployment specifications for the basic and composite indicators. The six specifications specify how data needed in the calculations of the indicators should be extracted and transmitted within the SoS. As we can see from the specifications, the security risk picture is updated on a weekly basis.

Table 6. Deployment specifications for the basic indicators b_3 and b_4 and the composite indicator c_2

Deployment specification for b_3
Extraction and transmission of list₄: At the start of each week, an automated ICT process creates the list list ₄ . The list list ₄ is created by extracting the names of all Trojan horses that “Small hydro power plant” is protected against from the information security database in the electrical power production infrastructure. The process transmits list ₄ to the risk monitor by using the internal data network of the electrical power production infrastructure.
Deployment specification for b_4
Extraction and transmission of list₅: At the start of each week, an automated ICT process updates the information security database in the electrical power production infrastructure. The database is updated by retrieving the names of all new Trojan horses that the security vendor of Client EPP offers protection against from the security vendor’s database. After having updated the database, the process creates the list list ₅ by extracting the names of all Trojan horses that the security vendor offers protection against from the information security database. The process transmits list ₅ to the risk monitor by using the internal data network of the electrical power production infrastructure.
Extraction and transmission of list₆: At the start of each week, an automated ICT process updates the information security database in the electrical power production infrastructure. The database is updated by retrieving the names of all new Trojan horses that the security vendor of Client EPP is aware of but does not yet offer protection against from the security vendor’s database. After having updated the database, the process creates the list list ₆ by extracting the names of all Trojan horses that the security vendor is aware of but does not yet offer protection against from the information security database. The process transmits list ₆ to the risk monitor by using the internal data network of the electrical power production infrastructure.
Deployment specification for c_2
Extraction of b_3 and b_4: The risk monitor calculates the two basic indicators b_3 and b_4 each week. After having calculated the two basic indicators, the risk monitor uses them to calculate the composite indicator c_2 .

RELATED WORK

The methodological approach presented in this chapter is a specialization of the approach presented in (Refsdal & Stølen, 2009). The approach in (Refsdal & Stølen, 2009) is general in the sense that it only restricts the risk identification to the identified assets and nothing else. In our approach, the risk identification focuses entirely on risks to security of provided services that have been caused by service dependencies. The approach in

(Refsdal & Stølen, 2009) can of course be used to identify indicators for the purpose of measuring the impact of service dependencies on risk to security of provided services, because of its generality. Compared to our approach, however, it is inferior. The approach in (Refsdal & Stølen, 2009) does not offer any support for dealing with SoS or service dependencies. In addition, it focuses to a much lesser extent on the calculations of indicators, and it cannot be used to specify how the indicator calculations should be embedded in the systems to be monitored.

We are not aware of other approaches targeting the capture and measure of impact of service dependencies on risks to the security of provided services. In (Debar, Kheir, Cuppens-Boulahia, & Cuppens, 2010), which is an approach for constructing formal models of services dependencies in information systems, the dependency models are used in security policy-based management. The dependency models are used to find enforcement points for security rules, which then support countermeasure deployment, and for computing the impact of attacks and countermeasures that propagate over the information system.

Service dependencies are also used in fault analysis (Gruschke, 1998) and dependability analysis (Rugina, Kanoun, & Kaâniche, 2007), as well as in analyses targeting critical infrastructures. A number of the approaches addressing service dependencies within critical infrastructures focus primarily on the consequences of infrastructure services not being provided. One such approach is (Svendsen, 2008). This approach is used to create models of infrastructure systems and their interactions. The models are used in computer simulations where the main purpose is to investigate how the functionality of infrastructure systems and interconnections react to different attack scenarios (“what if” scenarios where one or two systems are removed), and how mechanisms for strengthening the underlying dependency graph can be used. Svendsen’s approach differs, in particular, from our approach in that the likelihoods of incidents

(systems failing to provide services according to requirements) are not considered.

Even though a lot of work has been done within the SoS field, there is still no single accepted definition of what an SoS is. Examples of different definitions may be found in (Jamshidi, 2008). With different understandings of what an SoS is, we also get different understandings of what should be addressed with respect to risk and security. For instance, some definitions state that an SoS only consists of systems that operate independently of each other, i.e., that the different systems do not rely on services from other systems in order in to function. This is quite different from our understanding of an SoS. In the literature, SoS has received relatively little coverage when it comes to security and risk analysis. Papers like (Waller & Craddock, 2011), (Gandhi, Gorod, & Sauser, 2011), (Bodeau, 1994), and (Sage, 2003), focus primarily on the challenges and relatively little on actual approaches.

Dependent CORAS (Lund et al., 2010) is an approach for modular risk modeling, which can be used to document and reason about risk in SoS. It extends the CORAS risk modeling language with facilities for documenting and reasoning about risk analysis assumptions. It was motivated by the need to deal with mutual dependencies in risk analysis of SoS. By employing dependent CORAS we may document risk separately for the individual systems in an SoS. In addition, we document the risk analysis assumptions for the different systems, i.e., how threat scenarios and unwanted incidents, documented for other systems, may lead to threat scenarios and unwanted incidents, documented for the system in question. These assumptions are due to some form of dependencies, not necessarily service dependencies, between the different systems. Thus, dependent CORAS deal with dependencies in a general way compared to our approach, which only focus on service dependencies. The different risk models may be combined in the end, if the dependencies between them are well-founded, i.e., not circular.

There exist a number of approaches for measuring information security. One of those is the NIST Performance Measurement Guide for Information Security (Chew et al., 2008). This approach aims to assist in the development, selection, and implementation of suitable measures. It also provides a number of candidate measures. Unlike our approach, it is not specialized towards using these measures for the purpose of calculating explicit likelihood and consequence values.

CONCLUSION

In this chapter we have addressed the issue of how to capture and measure the impact of service dependencies on risk to security of provided services by the use of measureable indicators. To this end we have put forward a method consisting of four steps, which can be used in security governance for the purpose of assessing to what extent the security expectations to the provided services are achieved. To the best of our knowledge, there exists no similar approach. The applicability of the approach has been demonstrated on an example case within power supply.

In Step 1 of the approach, dependencies due to service exchange between different systems are captured. Their impact on risk to security of provided services is established in Step 2. In Step 3 we identify relevant indicators for monitoring the security risks arising from service dependencies, while in Step 4 we specify how likelihood and consequence values associated with the security risks should be calculated from sets of indicators and how these calculations should be embedded in the SoS. The result of applying the method is a security risk picture capturing the impact of service dependencies on security of provided services that can be dynamically monitored via the specified indicators.

An interesting topic for further research is the use of leading indicators (Jansen, 2010) to monitor the impact of service dependencies on security

risk. Many security indicators can be viewed as lagging indicators (Jansen, 2010). These indicators reflect security conditions that exist after a shift in security, e.g., occurrence of unwanted incidents. Leading indicators, on the other hand, reflect security conditions that exist before a shift in security. In the case of service dependencies, the leading indicators may be used to predict their future impact on security risk. By employing leading indicators, countermeasures may be implemented prior to security risks occurring.

ACKNOWLEDGEMENT

The research on which this chapter reports has been carried out within the DIGIT project (180052/S10), funded by the Research Council of Norway, and the MASTER and NESSoS projects, both funded from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreements FP7-216917 and FP7-256980, respectively.

REFERENCES

- Allen, J. (2005). *Governing for enterprise security* (Tech. Rep.). Pittsburgh, PA: Carnegie Mellon University. (CMU/SEI-2005-TN-023)
- Bodeau, D. (1994). System-of-systems security engineering. In *Proceedings of 10th Annual Computer Security Applications Conference* (pp. 228–235). Los Alamitos, CA: IEEE Computer Society.
- Chew, E., Swanson, M., Stine, K., Bartol, N., Brown, A., & Robinson, W. (2008). *Performance measurement guide for information security* (Tech. Rep.). Gaithersburg, MD: National Institute of Standards and Technology. (NIST Special Publication 800-55 Revision 1)
- Debar, H., Kheir, N., Cuppens-Boulahia, N., & Cuppens, F. (2010). Service dependencies in information systems security. In *Proceedings of the 5th International Conference on Mathematical Methods, Models and Architectures for Computer Network Security (MMM-ACNS'10)* (pp. 1–20). Berlin, Germany: Springer-Verlag.
- Gandhi, S., Gorod, A., & Sausser, B. (2011). A systemic approach to managing risks of SoS. In *Proceedings of 2011 IEEE International Systems Conference (SysCon'11)* (pp. 412–416). Piscataway, NJ: IEEE.
- Gruschke, B. (1998). Integrated event management: Event correlation using dependency graphs. In *Proceedings of Ninth Annual IFIP/IEEE International Workshop on Distributed Systems: Operations and Management (DSOM'98)*. Retrieved February 9, 2012, from <http://www.nm.ifi.lmu.de/pub/Publikationen/grus98a/PDF-Version/grus98a.pdf>
- Hammond, A., Adriaanse, A., Rodenburg, E., Bryant, D., & Woodward, R. (1995). *Environmental indicators: A systematic approach to measuring and reporting on environmental policy performance in the context of sustainable development*. Washington, DC: World Resources Institute.
- ISO (International Organization for Standardization), & IEC (International Electrotechnical Commission). (2005). *ISO/IEC 17799:2005 Information technology – Security techniques – Code of practice for information security management*.
- Jamshidi, M. (2008). System of systems engineering – New challenges for the 21st century. *IEEE Aerospace and Electronic Systems Magazine*, 23(5), 4–19.
- Jansen, W. (2010). *Directions in security metrics research*. Darby, PA: DIANE Publishing Company.

Ligaarden, O. S., Refsdal, A., & Stølen, K. (2012). *Using indicators to monitor security risk in systems of systems: How to capture and measure the impact of service dependencies on the security of provided services* (Tech. Rep.). Oslo, Norway: SINTEF. (SINTEF A22301)

Lund, M. S., Solhaug, B., & Stølen, K. (2010). *Model-driven risk analysis: The CORAS approach* (1st ed.). Berlin, Germany: Springer-Verlag.

Lysemose, T., Mahler, T., Solhaug, B., Bing, J., Elgesem, D., & Stølen, K. (2007). *ENFORCE conceptual framework* (Tech. Rep.). Oslo, Norway: SINTEF. (SINTEF A1209)

Office of the Under Secretary of Defense for Acquisition, Technology, and Logistics. (2002). *SoS and FoS FAQ*. Retrieved February 9, 2012, from [http://www.acq.osd.mil/dpap/Docs/FAQs--SoS & FoS.doc](http://www.acq.osd.mil/dpap/Docs/FAQs--SoS&FoS.doc)

OMG (Object Management Group). (2004). *Unified modeling language specification, Version 2.0*.

Refsdal, A., & Stølen, K. (2009). Employing key indicators to provide a dynamic risk picture with a notion of confidence. In *Proceedings of Third IFIP WG 11.11 International Conference (IFIPTM'09)* (pp. 215–233). Berlin, Germany: Springer-Verlag.

Rugina, A., Kanoun, K., & Kaâniche, M. (2007). A system dependability modeling framework using AADL and GSPNs. In de Lemos, R., Gacek, C., & Romanovsky, A. (Eds.), *Architecting Dependable Systems IV* (pp. 14–38). Berlin, Germany: Springer-Verlag.

Sage, A. (2003). Conflict and risk management in complex system of systems issues. In *Proceedings of 2003 IEEE International Conference on Systems, Man and Cybernetics* (pp. 3296–3301). Piscataway, NJ: IEEE.

Svendsen, N. K. (2008). *Interdependencies in critical infrastructures: A qualitative approach to model physical, logical, and geographical interdependencies*. Doctoral dissertation, University of Oslo, Oslo. (In series of dissertations submitted to the Faculty of Mathematics and Natural Sciences, University of Oslo, No. 748)

Waller, A., & Craddock, R. (2011). Managing runtime re-engineering of a system-of-systems for cyber security. In *Proceedings of 6th International Conference on System of Systems Engineering (SoSE'11)* (pp. 13–18). Piscataway, NJ: IEEE.

KEY TERMS AND DEFINITIONS

Indicator: An indicator is something that provides a clue to a matter of larger significance or makes perceptible a trend or phenomenon that is not immediately detectable (Hammond, Adriaanse, Rodenburg, Bryant, & Woodward, 1995). For example, an unexpected rise in the traffic load of a web server may signal a denial of service attack in progress. Thus, the significance of an indicator extends beyond what is actually measured to a larger phenomenon of interest.

Required Service Level: A service has one or more required service levels. Each required service level describes a requirement to one area of service scope. Availability, integrity, etc., are all examples of areas of service scope.

Security Asset: A security asset is something of value to security that a party seeks to protect.

Security Risk: A security risk is defined by the likelihood of an unwanted incident, and its consequence with respect to a security asset.

Security: Security is the preservation of confidentiality, integrity, and availability of information; in addition, other properties such as

authenticity, accountability, non-repudiation, and reliability can also be involved (ISO & IEC, 2005).

Service Dependency: A service dependency describes a relationship between a service provided by a system and services required by the system. A service depends on other services if it requires the other services in order to be provided.

Service: A service is provided by a system and consumed by a system. It represents the exchange of some commodity or the exchange of information.

System of Systems: A system of systems (SoS) is a set or arrangement of systems that are related or connected to fulfill common goals. The different systems may be controlled, operated, and maintained by different parties and within different jurisdictions. The loss of any system

may seriously impact the other systems and the process of fulfilling the common goals.

Trust: Trust is the subjective probability by which an actor (the trustor) expects that another entity (the trustee) performs a given transition on which its welfare depends (Lysemose et al., 2007). The level of trust may vary from 0 (complete distrust) to 1 (complete trust).

ENDNOTE

- ¹ This procedure is only valid if there are no mutual dependencies. We refer to the full technical report (Ligarden, Refsdal, & Stølen, 2012) for the general procedure.