Utilizing a Risk-Driven Operational Security Assurance Methodology and Measurement Architecture – Experiences from a Case Study

Reijo M. Savola, Teemu Kanstrén,
Heimo Pentikäinen, Petri Jurmu
VTT Technical Research Centre of Finland
Oulu, Finland
e-mail: {Reijo.Savola, Teemu.Kanstren,
Heimo.Pentikainen, Petri.Jurmu}@vtt.fi

Mauri Myllyaho
EXFO NetHawk
Oulu, Finland
e-mail: Mauri.Myllyaho@exfo.com

Kimmo Hätönen
Nokia Siemens Networks
Espoo, Finland
e-mail: Kimmo.Hatonen@nsn.com

***Abstract*—Practical measurement of information security of telecoms services is a remarkable challenge because of the lack of applicable generic tools and methods, the difficult-to-predict nature of security risks, the complexity of the systems, and the low observability of security issues in them. We discuss our experiences in utilizing a risk-driven methodology and associated measurement architecture in a practical case study. Effectiveness and efficiency are of main interest to stakeholders responsible for security. We note, however, that security configuration correctness and compliance with requirements are, in practice, the core objectives from an operational perspective. For these objectives there is more evidence available and it is easier to attain it. Our findings in this case study show a need for a wide range of security metrics to offer sufficient evidence of the design, implementation, and deployment of security controls. The case study also shows how visualization tools can be used efficiently to support the management of collections of these metrics.**

Keywords-Security; metrics; monitoring; risk analysis

I. INTRODUCTION

In the modern world, telecoms services are becoming more and more exposed to security threats. Sufficient and effective operational security is a result of adequate solutions and different stakeholders' activities at various levels, from the overall system administration to end-user applications and their behavior. Systematically obtained and managed evidence of the performance of these systems' security solutions benefits system development and maintenance.

The term *(information) security metrics* has become standard when referring to the security level or performance of a System under Investigation (SuI). A more appropriate term is *security indicators*, given the unpredictability of security risks, the complexity of systems, and their low observability in the absence of suitable measurement architectures. However, the former term is used in this study, to follow the most widely used terminology. Examples of security metrics application areas include risk management, comparison of different solutions, security assurance, testing, and monitoring [1]. This study focuses on operational security assurance.

In this study, we assume that there are three fundamental objectives of security measurement: effectiveness, efficiency and correctness. *Security effectiveness* means assurance that the stated security objectives are met in the SuI and the expectations for resilience in the use environment are satisfied, while the SuI does not behave in any way other than what is intended [2, 3, 4]. It is very difficult to measure security effectiveness directly; though activities such as long-term system use and penetration testing give some confidence. The quality of knowledge of risks is vital for security effectiveness. *Security efficiency* is assurance that adequate security effectiveness has been achieved in the SuI, in view of the resource, time, and cost constraints [2]. *Security correctness* is assurance that the security controls defined have been correctly implemented in the SuI, and the system, its components, the interfaces, and the processed data meet the security requirements [2, 3, 4]. Specific requirements, standards and best practices are used as references for security correctness assurance. While most experts agree that 100% secure systems are not possible, security correctness, including legal and standards compliance, is an important and achievable objective in practical security work.

From a security measurement perspective, the optimal ratio of security effectiveness and efficiency is of great interest. The goal of all security work is to ensure adequate security performance with respect to capability of mitigating and/or eliminating actual risks (effectiveness) using resources (e.g., time, money and functional performance) efficiently. We define operational security assurance, in line with [5], as grounds for confidence that security control realization is as expected in the operational system. This definition clearly emphasizes the security controls'

 ISBN: 978-1-61208-186-1

correctness, although it indirectly addresses effectiveness and efficiency too.

In this study, we discuss the role of operational security assurance in aiming at an optimal ratio of security effectiveness to efficiency in a Push E-mail system case study. We describe the security model components and discuss issues that we encountered when we implemented the model in an operational system.

Today smart phones are used in increasing amounts for various Web-based social services, such as Facebook and a variety of email applications. Additionally, the purchase of new music, goods, or software by means of mobile devices has become more common. The phones utilize several types of network connection at the same time. The shift in mobile devices' usage to other than only voice calls or Short Message Service (SMS) messaging has created a need for network and Internet operators to offer these services securely for the mobile devices' users. The demand for "always-on" functionality, especially in hand-held devices, has resulted in Push E-mail systems.

The main contribution of this study is in the benefit and challenge analysis of utilizing risk-driven security metrics and associated measurement architecture in a practical telecoms service case study. The metrics and measurement approach used are introduced in our previous work in [1, 2, 6–12]. The approach enables systematic and practical gathering and management of security evidence for different security related decision-making purposes.

The rest of the paper is structured as follows. Section II discusses the background and summarizes our previous work on this topic. Section III presents related work. Section IV presents the case study, with example metrics, and discusses our experiences of it. Section V addresses the benefits and challenges of utilizing our approach, before Section VI offers conclusions and poses future research questions.

II. BACKGROUND AND PREVIOUS WORK

In the discussion that follows, we offer a brief presentation of our previous work with security metrics and measurements and in the development of measurement architectures for them.

In [6], we introduced an iterative hierarchical security metrics development methodology, shown in simplified form in Fig. 1. This methodology is aimed at producing a balanced and detailed collection of security metrics, along with an associated measurement architecture. The measurement architecture includes the technical, administrative, legal and other means for gathering evidence. Note that a measurement architecture can be considered to be *risk-driven* if it is designed on the basis of risk-driven Security Objectives (SOs). The figure is identical to the description given in [5] apart from removal of the Quality-of-Service (QoS) metrics branch and replacement of the term "threat and vulnerability analysis" by "Risk Analysis" (RA). Term "threat and vulnerability analysis" has been used in the industry in referring to technical-level (or "architectural-level") RA. The term "RA" better represents the starting point; RA (either company-level or technical-level) as a holistic activity is the best choice as a foundation for security measurement goals. In [7], we integrated the above process into an industrial pilot study to match an iterative RA process and Agile software development. Experiences from the pilot showed the potential of security metrics in offering early visibility of security effectiveness and efficiency during security-critical phases of R&D. It became evident also that individual security metrics do not offer enough benefits; instead, collections of them are needed. Not much security effectiveness evidence is available during the first iterations of RA, when the need for it is at its highest.

In References [8] and [9], we discussed *Base Measures* (BM), *Derived Measures* (DMs), *measurement probes* and *measurement points*. BMs are abstract measurable properties of the SuI. Basic Measurable Component (BMC), discussed in [5], is a similar concept to BM. The difference between BMs and BMCs is that the latter represent the measurable properties *that are components of the decomposition* of SOs, whereas BMs can be standalone measures. It is possible that a property described by a BMC cannot be fully measured (through unavailability or unattainability of the evidence needed). DMs are interpretations of the BMs. In practice, one or multiple DMs can represent each BM. In generic models, the DMs that will be available in the future are not known, so only BMs can be presented. Development of detailed metrics, or DMs, for both of them may mean utilization of different measurement architectures. A measurement probe is a tool for performing checks of infrastructure objects in order to provide the information needed for purposes of measurements as defined by metrics. A measurement point is a point in the SuI, where one or more measurement probes are deployed.

In [8] and [9], we introduced a reference architecture for building a general monitoring framework, which can be utilized in Stage 5 in Fig. 1 for obtaining automated technical evidence for the purposes of continuous operational security assurance. This approach consists of four layers: (i) at the bottom, the Base Measure Layer, (ii) next, above it, the Data Collection layer, (iii) the Measurement Control and Processing Layer, and, at the top, (iv) the Presentation, Evaluation, and Management Layer.

As collections of security metrics can grow rather large, their management is a challenge. Moreover, aggregation of measurement values has pitfalls: relying blindly on an aggregated value can result in loss of important information and can lead to a false sense of security. There is no optimal weighting among branches, since many security problems arise from weakest links, which can be present in any sub-hierarchy. The benefits of visualization for human cognition can be utilized to increase the manageability of security metrics collections. In [10], we introduced a modeling and visualization tool called the Metrics Visualization System, or MVS, for the management of hierarchical security metrics and measurements. In the MVS *security metrics model* (SMM), the basic building block is a *security metrics node* (SMN). In an SMM, SMNs form a hierarchy. Same (or slightly customized) sub-hierarchies can be attached to different security controls at the higher level, because similar Security Controls (SCs) often mitigate or remove different security risks. All SMNs in the SMM have the same default

 ISBN: 978-1-61208-186-1

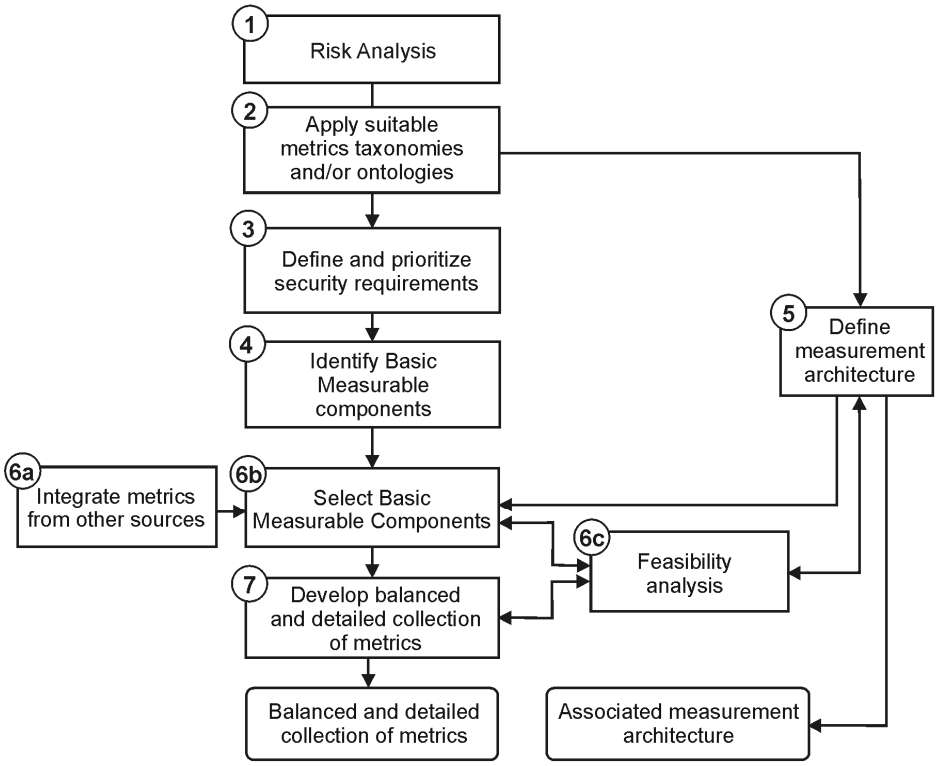

Figure 1. A security metrics development approach based on [5].

property fields: a distinctive name, metric confidence value (range 0…1), operation specification (logical expression), threshold criteria and associated visualization, polling frequency field for automated measurements, and enable/disable flag for operation value evaluation. The metrics in SMNs can be defined in terms of logical operations. All nodes can be colored or left blank. The default coloring scheme of the MVS imitates traffic lights: red stands for insufficient level, yellow for intermediate level, and green for sufficient level [10].

An important challenge is that many objects of the SuI system architecture are *unmanaged* [13]: they are not within the Administration Domain (AD) of the stakeholder carrying out security management and/or measurement of the SuI. Direct security measurements are not possible for an unmanaged object. However, a *trust value*, a certain value representing the amount of trust that the security of the object is adequate, can be associated with the object [11]. In practice, in many cases direct measurement of the SuI is not possible. Assessment of the properties essential to the security level can be used to replace direct measurements to achieve enough *indication* of the security level. In [12], a taxonomy of quality metrics for assessment of security correctness was introduced. The taxonomy uses a presentation inspired by the Common Criteria (CC) [14]. The quality metric families comprise (i) coverage, (ii) rigor, (iii) depth, and (iv) independence of verification. Any of six different quality levels discussed in [12] can be assigned to each family.

III. Related Work

Security quantification has been studied in the research already for several years now. Comprehensive overviews of security metrics approaches and objectives are found in, for example, [15–17]. Critical discussions and surveys are available in [18–20]. Skeptics often consider the current state of the art of security so low that any attempt to measure it would not be as success [17]. Evidently, the poor level is a result of the lack of usable tools and methods capable of systematizing security work and obtaining evidence of it. Furthermore, systematic security methods have not been emphasized enough in software engineering. The problem in the many research efforts aimed at security quantification has been that there is a lack of their validation in real or realistic case studies. Although no exhaustive validation has been carried out in this study, it offers many practical insights towards defining an industrial-strength security measurement framework. Among the major attempts to standardize security evaluation are the ISO/IEC 15408 Standard (CC) [14], the ISO/IEC 27000 series of standards [21], and many similar standards preceding them. A severe shortcoming in these efforts is that they are generic in nature and do not focus enough on security risk. Risk-driven and practical frameworks such as the one discussed in this paper offer new potential also for standardization.

 ISBN: 978-1-61208-186-1

IV. CASE STUDY: A PUSH E-MAIL SYSTEM

In this section, we briefly present the SuI of our case study, a company's Push E-mail service. We also present its identified security risks at high level, SOs and examples of metrics and measurements. The aim is three-fold: (i) to give an implicit example of the application of the approach discussed in Section II, (ii) to gather findings addressing the potential of security metrics and measurements, and (iii) to investigate shortcomings in the approach.

During the case study, different components of the SuI were integrated in a laboratory environment and their security risks and associated controls investigated, along with the metrics modeling and development. These activities were carried out in co-operation between the research and industrial partners of the BUGYO Beyond Eureka CELTIC cluster project. Project's main advances are summarized in [5].

A. The System under Investigation

The Push E-mail [22] functionality is situated at the last hop of the e-mail system, from the Receiver's E-mail Server to the Receiver's Client, which is today often a smartphone. Assume that a Sender would like to send an e-mail message to the Receiver at address name@a-company.com. The sequence of e-mail transfer consists of the following steps [11]:

1. The Sender asks from an E-mail Client called a Mail User Agent (MUA) to send an e-mail message to a Mail Transfer Agent (MTA) on the E-mail Server run by the Sender's Internet Service Provider (ISP).
2. The MTA requests the IP address corresponding to the "to"-address of the e-mail message from the Domain Name System (DNS).
3. The DNS responds with the address resolution information.
4. The Sender's MTA sends the message to the Receiver's MTA using the Simple Mail Transfer Protocol (SMTP).
5. The Receiver's MTA sends the message to his MUA using Post Office Protocol version 3 (POP3) or the Internet Message Access Protocol (IMAP).
6. In the case of an e-mail address managed by a local server, the message is passed to the Mail Delivery Agent (MDA) of the server instead of the MTA.

B. Risk Analysis and Security Objectives

Prioritized SOs for the SuI are agreed upon according to the RA. As concluded in [7], RA should be iterative throughout the system lifecycle. The major categories of risk identified are listed in Table I. Note that the risk categories in the table are not quantified and prioritized. A systematic prioritization effort by a group of core stakeholders is needed. In the table, "C" represents for confidentiality, "I" integrity, "A" availability, and "P" privacy risk.

R1 can result from exploitation strategies of many types – example cases where include an attacker using social engineering, or malicious insiders' knowledge, discovering critical vulnerabilities (e.g., weaknesses in a core configuration file), utilizing knowledge otherwise acquired, using malware, and exploiting a situation in which authentication is not strong enough and there are problems with security patches. R2 can stem from unintentional configuration problems (low-quality configuration management, security patch problems, and human error) or can be a result of attacker activity. R2 has potential to contribute to R1 too. R3 can be realized via brute-force (e.g., dictionary) attacks, or through network eavesdropping and exploitation of default e-mail user passwords. Loss of availability (R4) can be caused by Denial of Service (DoS) attacks, including Distributed Denial of Service (DDoS). Attack strategies for R5 exploit, first and foremost, low end-user security awareness or too great trust.

TABLE I. MAJOR RISK CATEGORIES FOR PUSH E-MAIL SERVICE

#	Risk	C/I/A/P
R1	Attacker gaining unauthorized access to the e-mail system as an administrator and potentially seizing it or even a larger system within or outside the AD	C/I/A/P
R2	Unintentional or deliberate misconfiguration of the system, making it vulnerable to attack	C/I/A/P
R3	Attacker gaining unauthorized access to e-mail messages and their content	C/I/P
R4	Attacker causing the e-mail service to crash or causing delays in it	A
R5	Phishing and spam causing indirect losses to the e-mail user	C/I/A/P

Nowadays, the environment in which Push E-mail services are used is vulnerable to the risks discussed above. In comparison to an e-mail service run on personal computers, typical Push E-mail clients run in a more challenging environment, on various types of mobile devices. Nowadays, keeping smart-phones up to date from a security perspective is not a trivial task. These problems seriously affect the operational security level of the SuI. Examples of these problems are the following:

- There are often changes in application and platform SW and in the service concept which the smart phone uses. Consequently, it is difficult to maintain a trusted and consistent up-to-date system configuration.
- Administration responsibilities are often unclear. The end-user might not have the sufficient rights to keep the configuration up-dated, and the administrator possessing those rights might not be able to maintain an up-to-date configuration. This is because the smart phones under any given company's administration may feature a myriad of network protocols with varying security levels.
- Some smart-phone models have advanced automated functionality, and the case of the end-user having enabled the wrong mode, these functions can cause security risks.

Note that typical Service Level Agreements (SLAs), which can be used to set requirements for companies' international services also, emphasize availability (R4). Other security risks are typically not addressed. In practice, this challenge contributes to difficulties in communicating the risks throughout the development, implementation and operation of services.

 ISBN: 978-1-61208-186-1

TABLE II. EXAMPLES OF SECURITY OBJECTIVES

#	SO on which an associated high-level SC is based	Risk
1	Authenticity and authorization of administration users and e-mail service users • End-user authentication (e-mail service, end-user role within the AD, and device) • Authentication of administration users • Client/server authentication • Access control in the AD	R1, R2, R3, R4
2	Up-to-date and secure configuration and SW versions for all relevant infrastructure objects • Clear responsibilities • Client: operating system, anti-virus and E-mail SW Client/server authentication • Authentication, Authorization and Accounting (AAA) Server SW within the AD • SW outside the AD (E-mail Server of ISP, MTA, and DNS)	R2, R3
3	Confidentiality and integrity of traffic and messages • Server/client traffic • Secure Sockets Layers/Transport Layer Security (TSL/SSL) channel • Authentication and authorization traffic • Wireless Local Areas Network (WLAN) channel	R3, R4
4	Up-to-date and effective anti-spam, anti-phishing and malicious attachment removal solutions	R5

In addition to the results of RA, the high-level objectives contributing to SOs can be based on suitable best practice, such as the ISO/IEC 27000 series of standards [21], which defines generic confidentiality, integrity, availability and privacy goals. Moreover, company-level security requirements and guidelines can be used. Table II presents examples of specific SOs of the SuI, and their connection to the risks specified in Table I. The actual security solutions of the system, SCs, are based on the SOs. Note that the list presented here is not complete. In the table, "AD" refers to the e-mail service AD of a company.

C. Modeling Metrics' Relationships to Security Objectives

Following the process of Fig. 1, security metrics models are constructed on the basis of the RA results and identified SOs in a prioritized manner. Fig. 2 shows a screenshot from the MVS tool depicting the highest levels of an SMM for an Authorization SC (SC1). The SMM includes the relevant SCs as SMNs immediately below the highest level entity, SuI node. Four other controls are shown in the figure, but suppressed for space reasons: SC2: Secure configuration and versioning, SC3: Confidentiality management, SC4: Spam filtering and malicious-attachment removal, and SC5: Service availability. Suppression of lower-level details is denoted by "+" at the bottom of a metrics node. It is important to note that alternative hierarchical classifications of SCs and, consequently, sub-hierarchies of the SMM, are possible. The choice of risks to be shown at the highest-level depends on the priority of them [7]. For example, secure configuration can be incorporated into separate sub-hierarchies. In the example SMM, we chose to emphasize it as a separate SC and associated SMM branch because of its importance in a typical Push E-mail use environment. In addition to the general confidentiality management sub-hierarchy (SC3), confidentiality concerns of other SCs are emphasized under the relevant sub-hierarchy.

In the example SMM, authorization is divided into two main branches, authentication and access control. The Fig. 5 in the Appendix 1 shows the MVS sub-hierarchy SMM for authentication. The leaves in the SMM represent BMCs with no expansion possibility or components for which a further breakdown is possible. Authentication is further divided into end-user authentication, administration personnel authentication and client/server authentication branches.

Because of the serious consequences of attacks could have for the administration of the e-mail service, a separate authentication metrics collection is used, with requirements stricter than the end-user requirements. Note that, depending on the smart phone device, the device authentication solutions can differ. Furthermore, the administration domain and e-mail service require dedicated authentication solutions

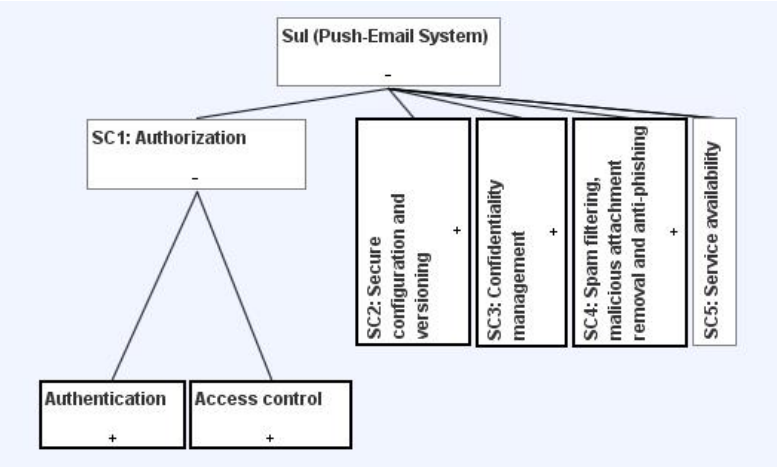

Figure 2. An example high level SMM for the Push E-mail case system. A screenshot from the MVS tool.

(that can be federated). All authentication branches mentioned incorporate ID Strength and Mechanism Strength sub-hierarchies, following the taxonomy shown in [6]. The ID Strength branch is opened under "End-user Authentication," and Mechanism Strength is shown under "Administration Personnel Authentication," to BMC level.

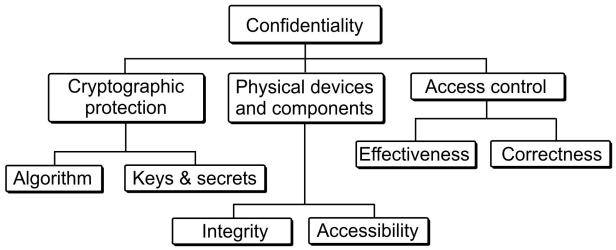

Figure 3. Confidentiality decomposition [6].

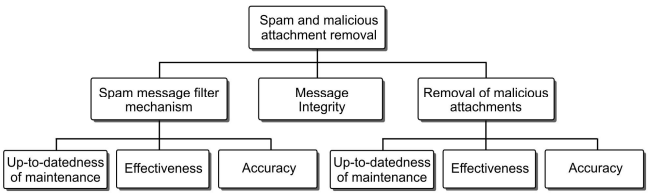

Figure 4. Spam and malicious attachment removal breakdown [11].

Similar sub-hierarchies can be constructed for other SCs via the MVS tool. The SMM sub-hierarchy for SC2 includes configuration, version control and testing and monitoring results, with different infrastructure objects relevant to the security solutions of the SuI forming sub-hierarchies. The infrastructure objects important for the Push E-mail service include the Mail Server, AAA Server, mobile device(s), Push E-mail Client SW, Spam Filter, and firewalls. In addition, several important infrastructure objects reside outside the AD (relevant for such purposes as identity management). Fig. 3 shows a decomposition from [6], which can be applied for metrics modeling associated with SC3. An example metrics hierarchy for spam filtering and malicious attachment removal (SC4) is shown in Fig. 4. SC4 includes security awareness metrics also, to reflect the level of the end-users' capability to withstand phishing attacks. Use of the availability sub-hierarchy (SC5) emphasizes the system's effectiveness in combating DoS and DDoS attacks, and evidence from robustness testing.

The example SMM is risk-oriented. However, it can be arranged in other ways, so as to match the needs of the users of the metrics better. For example, server administration personnel utilize various server programs. A metrics view showing these programs and their main configurations at high level would be beneficial for them.

D. Example Operational Metrics for Configuration Correctness and Deployment of Security Controls

Below, we discuss the difference between BMs and DMs by using some metrics examples from the Push E-mail system. Table III shows how some security metrics of operational BMs for the Push-Email service studied were defined. The examples listed here emphasize configuration correctness and adequate deployment of security controls. Examples of metrics for effectiveness of authentication, authorization, integrity, confidentiality, and availability are given in [6], and for spam filtering and malicious attachment removal in [11].

TABLE III. EXAMPLES OF BASE MEASURES IN PUSH-EMAIL SERVICE

BM	**SC**	**DM**
Mail Server: User authentication mode	1	1.1
Mail Server: Denial of plaintext authentication without encryption	1	1.2
Mobile device: AD password strength	1	
Mobile device: Device password strength	1	
Push E-mail Client SW: User authentication mode	1	1.3
Mail Server: Operating System UTD	2	2.1
Mobile device: Operating System UTD	2	2.2
Push E-mail Client SW: Service SW UTD	2	
Mail Server: IMAP server encryption activated mode	3	3.1
Mail Server: IMAP minimum encryption key length	3	3.2
Mobile device: WLAN encryption configuration	3	
Push-Email Client SW: Encryption algorithm mode	3	3.3
Mail Server: Spam filter SW UTD	4	
Mail Server: Malicious attachment removal SW UTD	4	
Mail Server: Mail backup UTD	5	5.1

TABLE IV. EXAMPLES OF DERIVED MEASURES

DM	**Example expression**
1.1	Configuration command check: `auth_mechanisms = plain login cram-md5`
1.2	Configuration command check in Dovecot configuration file: `disable_plaintext_auth`
1.3	Configuration command check: `auth_mechanisms = plain login cram-md5`
2.1	Version information query – example reply from the system: *Linux webrouter 2.6.32-24-generic #43-Ubuntu SMP Thu Sep 16 14:17:33 UTC 2010 i686 GNU/Linux*
2.2	Version information query – example reply from the system: *Linux Nokia-N900 2.6.28-omap1 1 PREEMPT Fri Aug 6 11:50:00 EEST 2010 armv7l unknown*
3.1	Configuration command check: `ssl = required`
3.2	Configuration command check: `ssl_cipher_list = ALL:!LOW:!SSLv2`
3.3	Configuration directive check: `SSLCipherSuite AES256-SHA:AES128-SHA`
5.1	Checking appropriate use of the `rsync` application, example: `rsync –a /home/user/Maildir /media/backupdrive/mail`

 ISBN: 978-1-61208-186-1

The BM examples listed in Table III originate from different abstraction levels in the metrics hierarchy. For example, "Denial of plaintext authentication without encryption" is a more detailed BM than "AD password strength." The SC number associated with the BM is shown. Furthermore, the table gives a reference to a DM derived from the BM, utilizing OpenSSL [23] and Dovecot Secure IMAP Server [24] commands, listed in Table IV. "UTD" refers to *Up-To-Datedness*. Mapping DMs from BMs is a 1-to-*N* process: one or several DMs represent each BM. In Table IV, only one example DM is given for each BM.

It is evident that the *SO representativeness* of scattered DMs is not enough for sufficient security evidence at the SO level. The situation is due to the information being missing, or the evidence needed being either unavailable or unattainable. Comparison of the examples in Table III and IV shows that, especially during the process of interpretation of BMs, a lot of information is lost. First of all, it is important that this kind of measurability challenges be duly kept on track in the metrics hierarchy properties.

Certain evidence requires other types of information-gathering than technical measurement architectures. For example, anti-phishing (SC4) metrics require measurement of end-users' security awareness. Puhakainen [25] investigates factors contributing to this awareness. He introduces theories based on training, awareness campaigns and punishment/reward.

The problem of missing information gaps can be mitigated by means of suitable assessment methods to give evidence of the situation. The quality metrics from [12] can be utilized for this purpose. The level of investigation preferred for assessment is the BMC level, but, depending on the granularity of metrics, the level may be higher or lower. The following questions form the basis for assessment:

1. Coverage: How widely has the BMC been investigated?
2. Rigor: Has there been enough rigor in the investigation?
3. Depth: In what depth has the BMC been investigated?
4. Independence: Has the investigation been carried out independently of system development and/or operation?

Measurement intervals for security-related measurements depend on various factors. The most important ones are the type of evidence needed, critical changes in the SuI, its availability, its attainability, and the efficiency constraints affecting measurements. Measurement needs can change in response to any combination of these factors.

V. Discussion of Benefits and Challenges

This section discusses the results from the application of our approach in the case study in a more general context.

A. Benefits

Today, state-of-practice activities in operational security assurance for telecoms services are largely based on *ad hoc* practices. Obviously, systematic evidence-driven security approaches bring several advantages.

By utilizing metrics, one can make more evident the potential bias between the security implementation and its specification [7]. This enables decision-makers to make informed decisions about investments in security countermeasures and risk mitigation. Visibility and constant evaluation of the status of operational security assurance highlight areas of potential problems and allow addressing them before risks are actualized. Security level in new R&D efforts and system operation will improve if factors contributing to security effectiveness and efficiency, along with their relationships, can be analyzed and documented.

The traceability of the objective requirement chain from the outcome of the first iterations of RA to SOs, and further to design and operational requirements, is systematized and better managed via the collection of metrics. Systematic risk-based thinking throughout the system lifecycle supports more effective and efficient security solutions. Feedback to R&D activities from the system's operation, utilizing metrics and measurements, is a powerful tool in assisting the future R&D efforts to focus on relevant security issues.

B. Challenges

As can be seen from the SMM of Figs. 2 and 3, obtaining sufficient evidence of security issues in a realistic system requires a wide collection of metrics, measurements and assessments. The need for wide metrics collections can result in a burden for practical service administration if there are no usable tools offering the right type of information.

Despite advances, such as the MVS tool, there are still many question marks in efficient metrics management. Simple measurement result aggregation, in combination with poor representativeness of the metrics used, results in the problem that the model does not express security phenomena in a full enough and credible way. Moreover, ensuring the correctness of metrics, i.e., that they represent the correct aspects relevant to SOs – still remains a challenge.

For many security issues, automated measurement is not possible; some of the required information is simply not available or attainable. Therefore, to increase the representativeness of metrics (i.e., fill the gaps between RA results, SOs, SCs, BMCs, BMs and DMs), one should use assessments. Credible assessment techniques still require advances. Moreover, common agreements on *trust value* management are needed.

The cost and effort in creation, maintenance and evolution of metrics and measurement architectures is a challenge. The advantages of using metrics and measurements should be compared with the added burden. Since the present study was a laboratory research effort, cost-effectiveness was not investigated. Today, proper administration of complex servers connected to the Internet requires personnel to follow their status constantly. In practice, resourcing can be troublesome. Accordingly, existing infrastructure, functionality and processes should be exploited as much as possible, to incur minimal overhead. Daily manual follow-up of logs is not feasible for visualization of every security aspect, or even every relevant one. The timing of responses to security problems is an issue too. For example, if a security risk related to a server configuration is detected in time, the question remains of whether it can be dealt with right away or only during off-peak usage hours. Live server setups require frequent updates to address security concerns and integration with

 ISBN: 978-1-61208-186-1

management tool updates. Otherwise, the tools would only provide snapshots of certain situations. If information-gathering is too detailed or frequent, the measurement approach can affect performance of the actual SuI. Moreover, the logs can grow so big that they can be stored for only short measurement periods.

VI. Conclusions and Future Work

We have discussed our experiences from development of security metrics and corresponding measurements in a Push E-mail service. The approach used is based on risk-driven hierarchical security metrics development, and utilization of a visualization tool and associated measurement and assessment approaches. Through the use of security metrics and measurements, the differences between security design and its implementation can be made evident, enabling informed decision-making. Moreover, the security objectives and requirements can be managed and traced throughout the system lifecycle.

Our experiences from the modeling of the case system showed that sufficient and credible security evidence consists of a wide collection of metrics, which should be managed in such a way that the relationships extending from high-level risk-driven security objectives and detailed measurements can be traced. In practice, the detailed measurements' correspondence with security objectives is often poor. Consequently, assessment and careful utilization of the available evidence is needed if we are to be able to fill the information gaps.

The cost-effectiveness of metrics and measurements was not addressed in this research effort. Our future work will include cost-effectiveness analysis of the proposed approach in real-world scenarios. Further evolution of the approach is planned in connection with this work.

Acknowledgements

The work presented here has been carried out in three European and Finnish national research projects: the BUGYO Beyond Eureka CELTIC cluster project (2008–2011), GEMOM EU FP7 ICT project (2008–2010), and Cloud Software Program (2008–2013) launched by the Finnish Strategic Centre for Science, Technology and Innovation TIVIT Plc. We wish to thank our colleagues involved in these projects for their related work and helpful discussions that made this study possible.

References

[1] R. Savola, "A taxonomical approach for information security metrics development," NORDSEC '07, 2007.

[2] R. Savola, "A security metrics taxonomization model for software-intensive systems," Journal of Information Processing Systems, Vol. 5, No. 4 (Dec. 2009), pp. 197–206.

[3] W. Jansen, "Directions in security metrics research," U.S. National Institute of Standards and Technology, NISTIR 7564, Apr. 2009, 21 p.

[4] Information Technology Security Evaluation Criteria (ITSEC), Version 1.2, Commission for the European Communities, 1991.

[5] S. Haddad, S. Dubus, A. Hecker, T. Kanstrén, B. Marquet and R. Savola, "Operational security assurance evaluation in open infrastructures," Proceedings of CRiSIS 2010, Sept. 26-28, 2011, Timisoara, Romania, pp. 100–105.

[6] R. Savola and H. Abie, "Development of measurable security for a distributed messaging system," Int. Journal on Advances in Security, Vol. 2, No. 4, pp. 358–380.

[7] R. Savola, C. Frühwirth and A. Pietikäinen, "Risk-driven security metrics in agile software development – an industrial pilot study," Submitted (2012).

[8] T. Kanstrén, R. Savola, A. Evesti, H. Pentikäinen, A. Hecker, M. Ouedraogo, K. Hätönen, P. Halonen, C. Blad, O. López and S. Ros, "Towards an abstraction layer for security assurance measurements (invited paper)," Proceedings of ECSA: Companion Volume, pp. 189–196.

[9] T. Kanstrén, R. Savola, S. Haddad and A. Hecker, "An adaptive and dependable distributed monitoring framework," Int. Journal on Advances in Security, Vol. 4, Nos. 1 & 2, 2011, pp. 80–94.

[10] R. Savola and P. Heinonen, "A visualization and modeling tool for security metrics and measurements management," Proceedings of ISSA 2011, Johannesburg, South Africa, 8 p.

[11] R. Savola, H. Pentikäinen, and M. Ouedraogo, "Towards security effectiveness measurement utilizing risk-based security assurance," Proceedings of ISSA 2010, Aug. 2–4, 2010, Sandton, South Africa, 8 p.

[12] M. Ouedraogo, R. Savola, H. Mouratidis, D. Preston, D. Khadraoui, and E. Dubois, "Taxonomy of quality metrics for assessing assurance of security correctness," Software Quality Journal, Online First, Nov. 30, 2011, 30 p.

[13] M. Ouedraogo, D. Khadraoui, B. de Rémont, E. Dubois, and H. Mouratidis, "Deployment of a security assurance monitoring framework for telecommunication service infrastructure on a VoIP system," Proceedings of NTMS '98.

[14] ISO/IEC 15408-1:2005: "Common Criteria for information technology security evaluation – Part 1: Introduction and general model," ISO/IEC, 2005.

[15] D. S. Hermann, "Complete guide to security and privacy metrics – measuring regulatory compliance, operational resilience and ROI," Auerbach Publications, 2007, 824 p.

[16] A. Jaquith, "Security metrics: Replacing fear, uncertainty and doubt," Addison-Wesley, 2007.

[17] N. Bartol, B. Bates, K.M. Goertzel and T. Winograd," Measuring cyber security and information assurance: A state-of-the-art report," Information Assurance Technology Analysis Center (IATAC), May 2009.

[18] J. McHugh, "Quantitative measures of assurance: Prophecy, process or pipedream?" Workshop on Information Security System Scoring and Ranking (WISSSR), ACSA and MITRE, Williamsburg, Virginia, May, 2001 (2002).

[19] D. McCallam, "The case against numerical measures of information assurance," Workshop on Information Security System Scoring and Ranking (WISSSR), ACSA and MITRE, Williamsburg, Virginia, May, 2001 (2002).

[20] V. Verendel, "Quantified security is a weak hypothesis: A critical survey of results and assumptions," New Security Paradigms Workshop, Oxford, U.K., 2009, pp. 37–50.

[21] ISO/IEC 27000:2009: "Information technology – Security techniques – Information security management systems – Overview and vocabulary," ISO/IEC, 2009

[22] R. W. Smith, "LPIC-2: Linux Professional Institute Certification, study guide," Sybex, 2011, 694 p.

[23] ——, "OpenSSL project – Cryptography and SSL/TSL toolkit," Website: www.openssl.org [Accessed Jan. 15, 2012].

[24] ——, "Dovecot – Secure IMAP Server," Website: www.dovecot.org [Accessed Jan. 15, 2012].

[25] P. Puhakainen, "A design theory for information security awareness," PhD thesis, University of Oulu, Finland, 2006.

 ISBN: 978-1-61208-186-1

APPENDIX 1

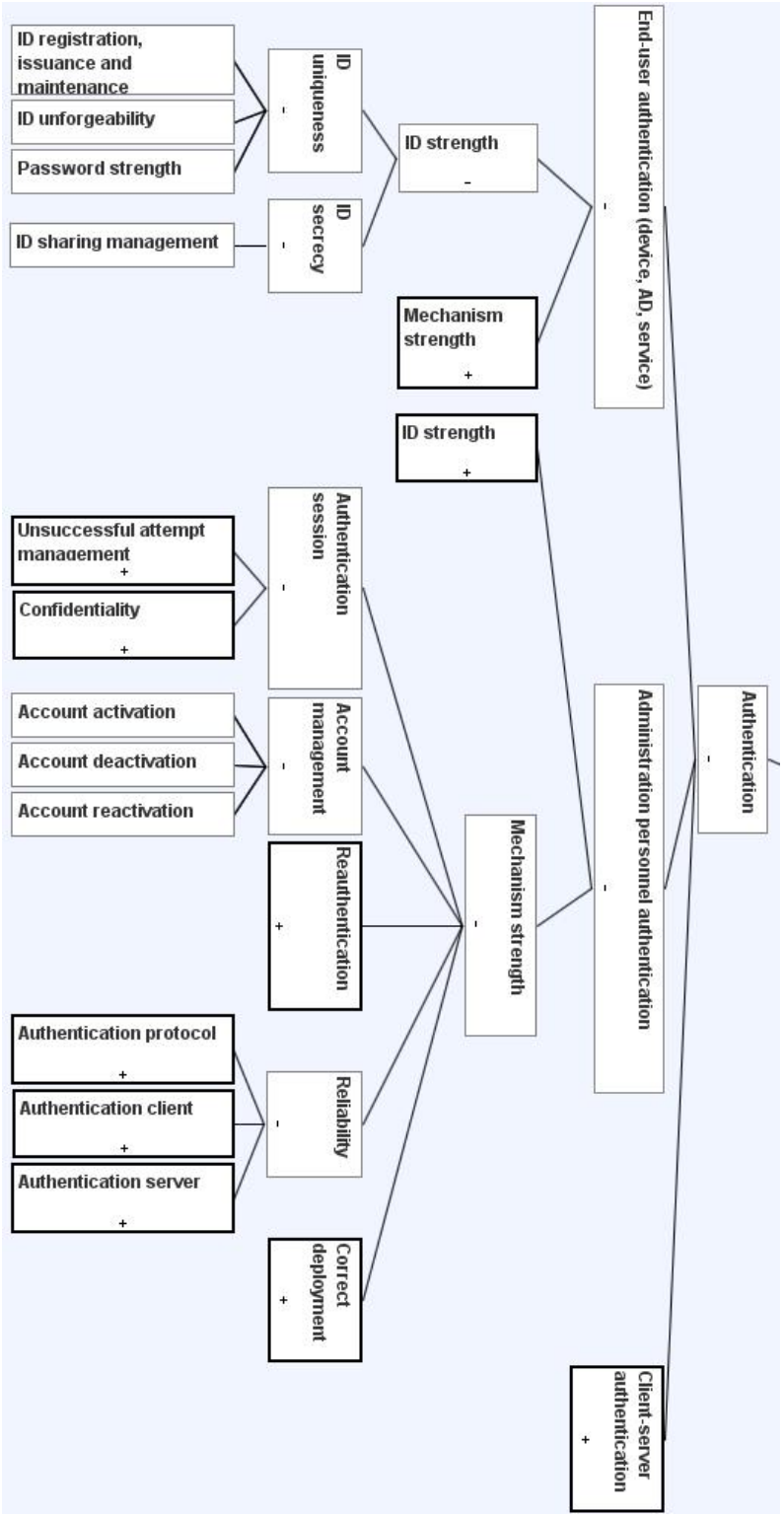

Figure 5. A screenshot of the authentication SMM branch.

 ISBN: 978-1-61208-186-1