Report from Dagstuhl Seminar 19451

Biggest Failures in Security

Edited by

Frederik Armknecht[1], Ingrid Verbauwhede[2], Melanie Volkamer[3], and Moti Yung[4]

1 **Universität Mannheim, DE,** armknecht@uni-mannheim.de
2 **KU Leuven, BE,** ingrid.verbauwhede@esat.kuleuven.be
3 **KIT – Karlsruher Institut für Technologie, DE,** melanie.volkamer@kit.edu
4 **Columbia University – New York, US,** moti@cs.columbia.edu

Abstract

In the present era of ubiquitous digitalization, security is a concern for everyone. Despite enormous efforts, securing IT systems still remains an open challenge for community and industry. One of the main reasons is that the variety and complexity of IT systems keeps increasing, making it practically impossible for security experts to grasp the full system. A further problem is that security has become an interdisciplinary challenge. While interdisciplinary research does exist already, it is mostly restricted to collaborations between two individual disciplines and has been rather bottom-up by focusing on very specific problems.

The idea of the Dagstuhl Seminar was to go one step back and to follow a comprehensive top-down approach instead. The goal was to identify the "biggest failures" in security and to get a comprehensive understanding on their overall impact on security. To this end, the Dagstuhl Seminar was roughly divided into two parts. First, experienced experts from different disciplines gave overview talks on the main problems of their field. Based on these, overlapping topics but also common research interests among the participants have been identified. Afterwards, individual working groups have been formed to work on the identified questions.

Seminar November 3–8, 2019 – http://www.dagstuhl.de/19451
2012 ACM Subject Classification Security and privacy, Social and professional topics
Keywords and phrases Cryptography, Hardware, Security engineering, Software engineering, Usability, Human Computer interaction (HCI), Human and societal aspects of security and privacy, Usable security or human factors in security, Security evaluation and certification
Digital Object Identifier 10.4230/DagRep.9.11.1

1 Executive Summary

Frederik Armknecht
Ingrid Verbauwhede
Melanie Volkamer
Moti Yung

General Introduction

In the present era of ubiquitous digitalization, security is a concern for everyone. Consequently, it evolved as one of the most important fields in computer science. However, one may get the impression that the situation is hopeless. Nearly on a daily basis, reports of new security problems and cyberattacks are published. Thus, one has to admit that despite the huge

efforts continuously invested since many decades, securing IT systems remains an open challenge for community and industry.

One of the main reasons is that the variety and complexity of IT systems keeps increasing, making it practically impossible for security experts to grasp the full system. This results into the development of independent and isolated security solutions that at best can close some specific security holes. Summing up, security requires to solve an increasing number of inter- and intradisciplinary challenges while current approaches are not sufficiently effective. The aim of this seminar was to gain an interdisciplinary view on security and to identify new strategies for comprehensively securing IT systems.

Goals

The goals of the seminar was to address the following main challenges and to commonly discuss solution strategies:

Challenge 1: Interdisciplinarity The topic of security is getting more and more complex and already understanding the state-of-the-art within one discipline is highly challenging. This makes it practically impossible to understand the problems and constraints from other disciplines. Moreover, different disciplines often have their own methods and "culture". From our experience, working with colleagues from other disciplines requires at the beginning an enormous effort to understand each other. The complexity grows even further when more than two disciplines are involved.

Challenge 2: Variety of Problems In each discipline, a variety of problems do exist. Naturally, researchers have to single out specific problems that they work on instead of aiming for comprehensive solutions. The selection of problems usually depends on several factors, e.g., background of the researcher, topicality of the subject, etc. Most often, researchers aim for solving very specific problems rather than coming up with more comprehensive solutions. Moreover, the selection is driven by interdisciplinary factors.

For sure, interdisciplinary research does exist already. However, it is mostly restricted to address very few disciplines and has been rather bottom-up by focusing on very specific problems. Instead, the scope of the seminar was to aim for a *broad top-down approach*. To this end, the focus was on the following questions:

- What are the main recurring reasons within disciplines why security solutions fail, i.e., the biggest failures? (Top View)
- How do these failures impact solutions developed in other sub-disciplines? (Broad View)
- What are possible strategies to solve these problems?

Structure

The seminar was structured accordingly. Before the seminar, a survey was conducted where the participants have been asked, what they consider to be biggest failures in security. The list of participants was composed of experts from different, selected sub-fields who were encouraged to explain the main challenges in their field to the audience. Here, ample opportunities for discussions have been provided. That is, instead of having many different talks back-to-back, we had several overview talks from different fields within the first few days. Afterwards, the whole audience commonly identified three topics to be further investigated in separate working groups:

1. The process and role of certifications
2. The human factor in security
3. The education of the society in security

These subgroups met in parallel and worked on specific questions. The remaining days were composed of workgroup meetings and individual talks. At the end of the seminar, the workgroups reported to the whole audience their findings.

This report summarizes the finding of the survey (Section 3), the topics of the individual talks (Section 4), and also the findings of the individual workgroups (Section 5).

2 Table of Contents

3 Survey Results

In order to prepare and to kick-off the seminar, an online survey was distributed to all participants. It mainly contained two questions:

1. What is the one biggest failure in security? Please explain why you selected this one as the biggest one.
2. Which other failures in security should be considered?

The Survey was filled out by 17 participants (3 from industry and 14 from research institutions). Participants have on average 21 years of past experience in security (with min. 13 and max. 36 years).

The open-ended text answers were analysed by two researchers. The answers were clustered and six main and five smaller themes were identified. For the analyses, it was decided not to distinguish between answers of both categories as several participants provided more than one failure in their response to question 1 and some provided more than two failures in their answers to question 2. Though, in the following when we provide quotes, those in italic are those taken from answer to question 2.

In the following the identified main themes are introduced and quotes are provided:

Theme 1: Lack of Holistic Approach for Complex Systems

Several answers were related to various aspects of (not) ideal approaches taken throughout the development of systems which need to be secured against attacks. Example quotes are:

- ... without adequate consideration of the importance of holistic design ...
- ... [systems] are too complex to be well-understood ...
- ... boundaries of a system get more and more fuzzy ...
- ... mechanism provides a solution for a very dedicated security challenge, one can often not exclude the existence of ... other security holes ...
- ... involve multitude of disciplines ...
- ... across disciplinary boundaries ...
- ... quality of risk modeling ... as a whole is ... poor
- ... list of assumptions for the overall system are not clear
- Making tradeoffs that overfocus on providing security to undifferentiated large scale groups rather than numerically smaller demographics

Theme 2: Lack of Usability

Several participants mentioned human related aspects wrt. security mechanisms. Note, the number of answers assigned to this one was higher than for all the other themes. Example quotes are:

- ... Not designing security with the Human Factor in mind – solutions with too much workload, complexity. Users are being set up to fail, ...
- ... Implementing more ideal security features with complicated procedures rather than usability ...
- ... usability is another central issue ... security mechanisms should operate "invisible" ... mechanisms complicated or impacting usability negatively ...
- Overload of IT users, e.g. requesting to memorize > 10 passwords.
- ... failure of organizations to appreciate the interplay between usability and security, driving usability underground, and compromising security ...
- ... which leads to the ... question of usability of security mechanisms ...

- ... why is security sometimes at perceived as trading off usability ... ?
- ... Lack of empirical testing of effectiveness of security measures.
- Lack of user friendly identity management infrastructure.
- The lack of ... unobservable communications usable by normal citizens

Theme 3: Not Learning from Past Mistakes

Several participants provided answers indicating that the community does not learn from past failures. Example quotes are:

- ... how we do not seem to learn from our mistake ...
- ... never seeming to learn from old mistakes. ...
- .. Incapability or inconsequence to learn from failures sustainably ...
- ... We patch it and learn about it on one system ... but when there is a shift to something new, similar ... vulnerability pops up again ...
- ... but many mistakes by programmers are long known and could easily be prevented ...
- ... lack of education where a new generation is doing the old mistakes ...
- ... we continue doing things just because that's the way we've always done them ...

Theme 4: Decision Makers Not Taking (appropriate) Actions

Several participants mentioned various types of decision makers (related to law and politics) in the failures they see. Example quotes are:

- ... we have been slow to update laws to reflect our technology, and slow to appreciate the impact of technology on legal protections ...
- .. Governments take a hands-off approach, and let organizations scale up until it becomes difficult to change ...
- ... lack of attention by decision makers, until sth. major happens ...
- ... Companies are rarely rewarded for building reliable systems ...
- ... [accept] convenient and cheap solutions that lead to major ... problems later.
- ... it seems to widely accepted that companies have outsourced security updates to the users. Users need to spend time and sometimes money ... to fix shortcomings of the systems they are using.
- ... lack of regulations from the onset. Anyone can write, publish/sell an app – other sectors require a clear process ...

Theme 5: Lack of Appropriate Certification Concepts

Answers related to certification and standardisation were assigned to this theme. Example quotes are:

- ... lack of certification concepts for the security and privacy of products and services that scale to the needs of agile development and cloud delivery ...
- ... the question of suitable criteria for cloud based, agile software is not addressed at all in the discussions ...
- ... failure of standards bodies ... to make certificate infrastructure work properly ...
- Not understanding the degree of accuracy required, leading to high failure rates.

Theme 6: Lack of End User Protection

Several answers focused on the general inability of protection end user / consumers adequately. Example quotes are:

- ... lack of protection of consumers against malware ...
- ... lack of robust online identities ...
- The inability ... to provide consumers with a reasonable &reasonably ICT device for day-to-day tasks –the digital ... Golf to use a car-market analogy
- Protecting humans from bad decisions. Why are systems designed in a way that a user can damage the whole system just by opening a link or an attachment of a mail? ...
- "Solutions" which place the risk at the weak parties ...

Smaller but more specific themes

The following specific failures have been mentioned (note, only those provided by at least two participants are mentioned)

- Unsecure programming language (3)
- Phishing is still among the major causes of breaches (2)
- Passwords are still around (2)
- Issues related to machine learning (2)
- Web browsers becoming an execution environment for everything (2)

Overall, the result of this survey allowed us to make all seminar participants aware of the wide range and level of abstraction of failures one can think of. The result helped us also to group in working groups.

4 Overview of Talks

4.1 DDoS Still Challenging 20 Years Later

Sven Dietrich (City University of New York, US)

We provide an overview of the fundamental flaws that have contributed to allowing the distributed denial-of-service (DDoS) phenomenon [1, 2] to happen over the last 20 years. This includes design flaws for the Internet and its protocols, management decisions, and sometimes faulty defensive stances. We show that the imprecision of the DDoS problem itself contributed (and still contributes) to the difficulty in responding to it. Incremental fixes have only created good albeit partial solutions to subproblems of the DDoS phenomenon. Defense mechanisms have varied from attack source identification, volumetric attack detection, network puzzles, pushback from target-resident detection, and command-and-control detection, and graph-based analysis for botnets [6]. The migration of attack sources over the years from government or university owned computers, to broadband-connected home computer systems and most recently Internet-of-Things devices shows the active continuation of the DDoS phenomenon and our inability to completely suppress the problem [7]. Repeated calls for an overhaul of the Internet, allowing for improvement and better flexibility in addressing the DDoS problem, have been stalled over the years, even though some good starting points for next-generation network infrastructures do exist [4, 3], but many challenges remain to be solved.

References

1 Jelena Mirković, Sven Dietrich, David Dittrich, and Peter Reiher. *Internet Denial of Service: Attack and Defense Mechanisms*. Pearson, USA, 2004.
2 CERT. *Results of the Distributed Systems Intruder Tools Workshop*. Published December 7, 1999.
3 Marc C. Dacier, Sven Dietrich, Frank Kargl, Hartmut König. Dagstuhl Seminar 16361: Network Attack Detection and Defense: Security Challenges and Opportunities of Software-Defined Networking, September 2016.
4 Marc Dacier, Hartmut König, Radoslaw Cwalinski, Frank Kargl, Sven Dietrich. *Security Challenges and Opportunities of Software-Defined Networking*. IEEE Security & Privacy 15(2):96–100 (2017).
5 David Dittrich, Sven Dietrich. *P2P as botnet command and control: a deeper insight*, in Proceedings of the 3rd International Conference on Malicious and Unwanted Software (Malware), pp. 46–63, October 2008.
6 Baris Coskun, Sven Dietrich, Nasir Memon. *Friends of an enemy: identifying local members of peer-to-peer botnets using mutual contacts*. In Proceedings of the 26th Annual Computer Security Applications Conference (ACSAC), pp 131-140. Austin, TX, December 2010.
7 Sven Dietrich. *Cybersecurity and the Future*. IEEE Computer 50(4): 7 (2017)

4.2 Research Directions for a Safer Europe

Fabio di Franco (ENISA – Attica, GR)

Main reference Fabio Di Franco: "Analysis of the European R&D priorities in cybersecurity", ENISA, December 2018.
URL https://www.enisa.europa.eu/publications/analysis-of-the-european-r-d-priorities-in-cybersecurity/at_download/fullReport

Europe should become "a global leader in cybersecurity by 2025, in order to ensure the trust, confidence and protection of our citizens, consumers and enterprises online and to enable a free and law-governed internet", as stated at the Tallinn Digital Summit in September 2017. The focus of the report is to highlight and recommend how focussed R&D can address emerging challenges that might pose a severe risk to our society. A key element is the recognition that the world is moving digitally and fast. The speed of adoption of new technologies has a potentially huge benefit resulting in increasing productivity, but at the same time may also pose risks if the technology where used against the best interests of society. Social norms take dozens of years to develop and the digital transformation is creating an increasingly blurred distinction between the digital and the physical world. In the digital world, a small number of corporations, popularly referred to as Internet giants , are increasingly required to service the societies of the 21st century. However, this requires a barter between the user's data and the internet giants' services: the users allow the digital platforms to track their location, record their interests and monitor their online activities in return for a wide series of services demanded by the users. In almost all cases, there is no direct user interaction in the bartering system, or only to the extent that the user understands the meaning of data collection notices. orted):

References

1 Fabio Di Franco *Analysis of the European R&D priorities in cybersecurity*. ENISA, Dccember 2018

4.3 Attacker Models and Assumption Coverage

Felix Freiling (Friedrich-Alexander-Universität Erlangen-Nürnberg, DE), Frederik Armknecht (Universität Mannheim, DE)

In his seminal paper on "failure mode assumptions and assumption coverage" [1], David Powell defines several central concepts:

1. The notion of *failure mode assertions*, i.e., precise statements about the way in which certain components may fail in the time domain and the value domain.
2. The *failure mode implication graph*, i.e., a lattice induced by the combination of failure modes defining the partial order between different composed failure modes.
3. The notion of *assumption coverage*, i.e., the probability that the assertion defining the assumed behavior of a component proves to be true in practice conditioned by the fact that the component has failed [1, p. 391].

The goal of this discussion session was to reflect on the similarities and differences between safety and security regarding attacker assumptions and assumption coverage and to ask whether any related work and concepts exist. Safety was understood here as the area of fault-tolerance and dependability, whereas security was understood as the area of cryptography. The connection to the title of this Dagstuhl seminar was the fact, that one of the biggest failures in security appears to be the fact that we do not learn sufficiently from other areas.

Regarding the concept of attacker assumptions, our observation was that in safety attacker assumptions are usually fixed for a specific scenario and in this scenario often empirically measurable. Examples are failure rates of components or maximum frequency of bitflips on communication lines or in memory. The mechanism, with which a component attempts to tolerate these problems, has no influence on the assumption coverage.

In security, the attacker assumption is usually determined by a domain expert and must be regularly checked whether it is still correct. It can even change spontaneously. In circumstances where this is expected to happen, issues of *risk management* arise. Furthermore, security mechanisms can have an effect on attacker behavior:

- either a strong mechanism deters attackers and makes the system uninteresting compared to others,
- or a weak mechanism is circumvented easily with minimal effort.

In safety we have concepts like *graceful degradation* and *stabilization*. On the one hand, graceful degradation means that the level of violation of specification is proportional to the strength of failure behavior. On the other hand, stabilization refers to a temporary violation of a safety property if attacker assumption is violated, and a return to safety property when attacker assumption is satisfied.

In security, the attacker assumption is usually a worst-case attacker assumption. Intermediate levels of attackers are unusual. Also switching between different security mechanisms is unusual and it is unclear on what basis the switch should occur since many violations of confidentiality and integrity are undetectable.

In the discussion, people from security admitted that worst-case assumptions usually are prefered, but often also weaker assumptions are used, so the cryptography community does not really live up to this claim of always choosing worst-case assumptions.

It was also mentioned that *testing* has strong similarities to transient attacks that try to throw a single machine off the tracks, and that stabilization has similarities to the mechanisms used to tolerate denial-of-service attacks.

References

1 David Powell. Failure mode assumptions and assumption coverage. In *Digest of Papers: FTCS-22, The Twenty-Second Annual International Symposium on Fault-Tolerant Computing, Boston, Massachusetts, USA, July 8-10, 1992*, pages 386–395, 1992.

4.4 Values in Computing – a Short Talk

Lucy Hunt (Lancaster University, GB)

Joint work of Emily Winter, Stephen Forshaw, Lucy Hunt, Maria Angela Ferrario
Main reference Emily Winter, Stephen Forshaw, Lucy Hunt, Maria Angela Ferrario: "Towards a systematic study of values in SE: tools for industry and education", in Proc. of the 41st International Conference on Software Engineering: New Ideas and Emerging Results, ICSE (NIER) 2019, Montreal, QC, Canada, May 29-31, 2019, pp. 61–64, IEEE / ACM, 2019.
URL https://doi.org/10.1109/ICSE-NIER.2019.00024

Values in Computing (ViC) is about understanding how human values influence software production and transforming the way values are considered in software industry practices, policy making and education. With the increasing number of high impact technology breaches and failures, we need computing professionals equipped to understand what human values are and what social responsibility means. To this end, we need to help create more resilient, secure and less vulnerable software systems that are mindful of the wider ethical, social and human impact of what their technology does or could do. ViC has a body of research establishing a framework for the systematic investigation of human values in software production and a website to disseminate our work (www.valuesincomputing.org).

How can software (security) incident story-telling be used to improve SE industry and education practices?

4.5 DRM and Security – A Big Failure?

Stefan Katzenbeisser (Universität Passau, DE)

In the talk we discuss the evolution of Digital Rights Managenent techniques, which were proposed to secure online content. The key idea was to encrypt content and transmit the encryption key in a special license. The failure of DRM can be tracked down to technical issues (such as the absence of trusted hardware at that time), changes in the business model (such as the uprising of flatrate streaming media) and usability problems. Media security tried to fill this gap by marking distributed media invisibly. Still, the fundamental different nature of analog signals led to numerous problems (such as robustness issues and conflicts in dispute resolution). Nevertheless, the techniques developed in the area of media security nowadays play a significant role in the construction of covert channels.

4.6 Failures in TLS Implementations

Olivier Levillain (Télécom SudParis – Evry, FR)

In the recent years, we saw a lot of implementation bugs in SSL/TLS stacks, ranging from classical programming errors to parsing bugs, cryptographic issues and state machine flaws. In many cases, similar problems were found in different independent implementations. Maybe the root cause of the problem is not only the developers' lack of skills. On the contrary, it might be time to use better languages and better development methodologies, as well as to improve the specifications we produce. Regarding this last point, we discuss what TLS 1.3 can/will bring to improve the situation.

4.7 Human Involvement in Highly Automated Systems: Human System Integration in Security

Joachim Meyer (Tel Aviv University, IL)

The security of systems depends to a large extent on the actions of the humans who interact with the technology. Human actions can introduce threats, but they can also help to mitigate risks. Humans are often supported by automation that provides them with advice on decisions, guides their actions, blocks alternatives, and may automatically perform acts that are deemed necessary. The talk addresses the question of human-systems integration in the context of automation, presenting four different ways in which humans can be involved in systems (humans receive advice, humans supervise automation and intervene in certain cases, humans supervise automation and set parameters, and maintaining "meaningful human involvement" without specifying its nature). Quantitative models and empirical results for the different types of involvement are shown, and some implications for system design are discussed.

4.8 The Biggest Failures to "Protect" You in the Internet

Vasily Mikhalev (Universität Mannheim, DE)

Today, personal data is among the most important resources which is being collected by some governments and big organizations. This data can be used for many different purposes including targeted advertising and even targeted propaganda. The existing technologies which are based on the combination of data science methods together with better understanding of human brain allow for "hacking" human beings using their personal data collected from the internet and for manipulating people's emotions. In this talk, we discuss the "protection" measures that Russian government has implemented in order to increase the security of citizens and why most of these measures appeared to be the biggest failures.

4.9 Relation of Business Models to Security (Failures)

Sebastian Pape (Goethe-Universität Frankfurt am Main, DE)

URL https://pape.science/files/talks/1911_Pape_Dagstuhl.pdf

When looking at the usability of current systems, we can note systems often leave the users in (potentially) dangerous situations. In theory, it should not be possible to brick a system or get infected by malware when reading mails or working on office documents. Many of the features are used by a small number of users or not appropriate for the tool leaving users in a state with lots of rules what they should do (do not click on embedded links, do not open attachments, do not activate macros, ...). As a consequence, users are used to do 'strange things' for the sake of security. This can be exploited by dark patterns and companies make use of it by their business models. For example when companies blame hackers for outage or simply security failures, outsource consequences of bad security (e.g. malwertising, insecure IoT devices) and effort (correction of false positives, e.g. in malware detection), and obfuscate business goals with security (e.g. when asking for phone numbers for two factor authentication, but inadvertently used them for advertising).

The result of that is a downward spiral where users have the feeling that they need to do 'strange things' for the sake of security which can be exploited by companies to ask them to obey 'strange orders' pretending to improve the users security. Which again increases the users feeling that they need to do 'strange things' for security reasons.

4.10 Memory Corruption Vulnerability Exploitation and Mitigations

Michalis Polychronakis (Stony Brook University, US)

In this talk I will present our work on generating self-specializing software that i) reduces its attack surface by removing unneeded code and logic according to mission-specific or end-point-specific configurations and dependencies, and ii) shields itself against exploitation by retrofitting specialized protection mechanisms, such as code randomization and data isolation. Endpoint-specific specialization is facilitated by a novel binary code transformation framework that relies on compiler-rewriter cooperation to enable fast and robust fine-grained code transformation on endpoints, while achieving transparent deployment by maintaining compatibility with existing software distribution models.

4.11 Trusted Computing: The Biggest Failure or Opportunity?

Ahmad-Reza Sadeghi (TU Darmstadt, DE)

After years of research in hardware security, we are still missing adequate solutions to protect modern computing platforms. Deployed hardware solutions like PUFs, TPMs, and Trusted Execution Environments (TEEs) are lacking widespread usage, or have been attacked

through various side-channels. Additionally, we are witnessing a shift towards cross-layer attacks, exploiting hardware vulnerabilities from software, also remotely, as demonstrated recently by attacks like CLKScrew, Meltdown, and Spectre, which affect even systems with advanced defenses such as (Control Flow Integrity (CFI). Moreover, the Hack@DAC 2018 hardware security competition revealed a protection gap for current chip designs, since existing verification approaches may fail to detect certain classes of vulnerabilities in RTL code. In this talk will provide an overview of hardware-assisted security. We will discuss the impact of deployed solutions, their strengths and shortcomings, as well as new research directions.

4.12 Challenges of Regulating Security

Christoph Sorge (Universität des Saarlandes, DE)

Can legislation help mitigate the "Biggest Failures in Security"? Laws can obviously influence behaviour, and provide incentives to prioritize security. However, IT security legislation is hard due to conflicting goals.

Unspecific laws are not very useful. They lead to uncertainty, and even companies trying to abide by the laws risk fines or civil liability. Too specific regulation is quickly outdated, and can only cover individual sectors. Instead of detailed ex-ante regulation, liability rules could be considered as an alternative. Liability, however, also requires an understanding of obligations (and does not replace this understanding). As a consequence, IT security regulation usually has a limited scope. The focus can be on a specific sector, or on specific aspects such as security management and processes.

The German communication platform used for communication between lawyers and courts (beA) may serve as an example for a failure in security. Its security issues were, in part, caused by a problematic regulation approach and a resulting lack of requirements engineering.

To conclude, security legislation may work, as long as its scope is limited, and there are ways to adapt the legal requirements due to technical innovation. The technical community, however, should not wish for a detailed and overarching security regulation.

4.13 Fantastic Embedded Security Failures and Where to Find Them.

Lennert Wouters (KU Leuven, BE)

Main reference Lennert Wouters, Eduard Marin, Tomer Ashur, Benedikt Gierlichs, Bart Preneel: "Fast, Furious and Insecure: Passive Keyless Entry and Start Systems in Modern Supercars", IACR Trans. Cryptogr. Hardw. Embed. Syst., Vol. 2019(3), pp. 66–85, 2019.
URL https://doi.org/10.13154/tches.v2019.i3.66-85

During this talk we discuss common security issues encountered in embedded devices.

We take a look at issues ranging from the use of broken cryptographic primitives to insecure firmware updates and backend API issues. All of these issues are discussed using several real world examples ranging from vacuum cleaners to high-end cars. The goal of this talk is to spark discussion on how these issues came to be and how we can prevent them in the future.

4.14 Layers of Abstraction and Layers of Obstruction

Moti Yung (Columbia University – New York, US)

In this work we argue that what has made the field of "computer science" and its realization in real life as the "Information Technology Industry" successful, in fact, makes security hard! The success of computer science evolves around its evolution as a field where "complexity is controlled", namely, the ability to abstract sub-problems and sub-fields, solve problems in the abstracted domain and then apply it to the entire system in the right layer. The ability to solve concrete specific problems within a layer extends to sub-area, which enables the splitting of computer science into well defined courses: one can study hardware, computer organization, architecture, software, operating systems, databases, computer languages, algorithms, etc. in separate courses, yet in reality computation as a field employs all areas. Well defined API's and other mechanisms to connect layers enable also separate companies to deal with a subarea: hardware, database management system, cloud infrastructure, application package, which again, in reality work together.

When it comes to the area of security, we have to deal with an external threat, typically described as a threat model or an adversary. The adversary is an external entity to the system, hence it does not obey the layering assumption and design methodology: it is going to attack across layers! Thus, to defend systems practically, the notion of ethical hacking (white hat and red hat teams) that mimc attacks and itself performs attacks and observations across layers is typically employed.

We examine how the layers of abstraction, most often obstruct design of security. We ask: Is practical white hat monitoring and examination the only way to remedy the situation, or can design be updated to include some cross layers security considerations? We attempt to examine by example the latter.

The example we use is the development of the "Universal Second Factor" (U2F) by looking at the small example of servers, additional servers, user, device, and second factor device. By showing that considering attacks of different elements in the system, and further measures that are taken to cope with it, a design which is more robust and foils more attacks can be achieved. It demonstrates a possible refinement methodology, which adds attacks on other layers, as part of refining a design of a layer, thus being much more robust than merely considering each layer by itself.

5 Working groups

5.1 Certification Working Group

Felix Freiling (Universität Erlangen-Nürnberg, DE) and Begül Bilgin (Rambus – Rotterdam, NL & KU Leuven, BE)

The topic of this working group started out rather fuzzy as a discussion involving "certification, quantification, liability, responsibility, etc." in the context of avoiding security failures in the future, and so the working group started by collecting and sorting out the issues that

had motivated the participants to join the group. Participants were asked to provide specific questions which were subsequently grouped into three main categories:

1. Technical aspects of certification, e.g., how to integrate different perspectives and needs into the certification process,
2. understanding certificates, e.g., how to formulate the essence of the certified security properties so that relevant stakeholders can understand them, and
3. the big picture, on how responsibility, liability and regulation work together, e.g., the usefulness of certification in the absence of quantified risk models, possible civil or criminal liability for bad security products, or economic incentives for certification.

We aimed to go top down from category 1 to 3, but since a lot of questions from category 1 had been discussed in the talk by Volkmar Lotz on "Security Product Certification" the discussion started from category 2 with frequent side steps into other categories.

5.1.1 Understanding Certificates

Certification is often confused with penetration testing, a common technique to exhibit the security of a system in practice. While both topics are related, certification usually consists of a fixed set of tests that are performed more in the direction of a a checklist, while in penetration testing a skilled attacker tries to find vulnerabilities with defined resources. Security certification in terms of Common Criteria, however, is very close to penetration testing since it required independent analysis, repeatability, and a definition of attackers' resources.

What is also often confused is that a certificate for some part of a system does not necessarily imply the security of that part of the system. It always depends on the scope of the certification and the commitment of the involved parties. For example, if a specific security parameter (e.g., the ECC curve choice) was not included in the certificate, then plugging in the wrong security mechanism (the wrong curve) makes the system vulnerable. In the ideal process of *committed certification* all stakeholders try to honestly and with true interest try to raise the security level of a system or product through certification. But in practice, it is often not clear whether certification is applied in this way. This is exhibited by the often fierce battle of stakeholders about the scope of certification. A trait often seen in practice and termed *creative certification* is to formulate the certification goals in such a way that they sound good and appear to capture the essence of what is to be proven, but at second sight fail to follow the spirit of certification. Certifying a product will therefore often follow the letter of law but lead to no clear increase in security. Even worse, *fraudulent certification* tries to misuse the certification process to make certain stakeholders like the public believe in a security property which was never actually intended to hold.

In this context it is important to understand the concept of a *protection profile* as defined in the Common Criteria, which is a carefully crafted statement of the security targets and the associated resource bounds (cost, etc.) for attackers tailored to a specific class of systems. The discussions frequently referred to the example of a protection profile for electronic voting systems developed in Germany by the Federal Information Security Agency (BSI) which took about 4 years to develop. This is also a general problem in certification: certification documents must be precise, but they still should be understandable. Today, many certification documents are dominated by rather mechanical language that is hard to understand by people who are not from the regulation field. For researchers, for example, it is often easier to read and understand an evaluation report from an independent evaluator or white hat hacker that is written more like a research paper.

Looking at certification in terms of Common Criteria, it was mentioned that certification appears to work better for hardware than for software. The reasons for this were conjectured to be (1) the higher complexity of big software systems in contrast to big hardware systems and (2) the need (or maybe trend) of commercial software for frequent functionality updates. It was also mentioned that in the context of safety systems, systems are only allowed to operate in a certified state. The discovery of a security vulnerability puts system operators in a conflict between safety and security: they may either keep the safety certification of the system and risk successful attacks, or violate safety considerations due to security updates. This is a fundamental and still unsolved goal conflict.

5.1.2 Technical Aspects of Certification

As discussed above, the scope of certification is important and is usually described in the protection profile. In a certain sense, it defines what is "sufficient" to call a system secure. Perfect security, i.e., the ability to withstand all attacks, if often not the aim. For certain attacks, other security behaviors can be acceptable. With respect to data protection issues it was asked whether we can get inspiration from other application areas about what happens when a software component does not function as it is supposed to or when there are usability problems, e.g., for a customer to claim the money used to purchase the product.

The newly introduced European data protection legislation GDPR states explicitly that "state-of-the-art" security evaluation has to be perfomed, but it also mentions the cost factor. It is often not so clear and debated what this means in practice, and this is also an issue where the research community needs to take a stand.

Another example is the legislation involving critical infrastructures where also state-of-the-art certification is often referred to. Such infrastructures are geeting increasingly large. As an example the infrastrucure to manage millions of autonomous vehicles in the future was mentioned. There are, however, already examples from this area that were discussed. For example, trains and the railway system have a long tradition of safety and (partly also) security evaluation. There, a large system (a complete train system) is divided into pieces (physical or logical) which should have the same security level and which should follow standardized functional requirements almost to the point of having a checklist. Problems arise in the interconnections of these systems because composibility of security properties and checking for composed requirements are known hard problems. It was discussed in what way the division into parts could help in the case of updates for already certified devices. One could try to use isolation properties to update certain parts of the system without affecting others.

On a technical side, it was asked whether formal verification could not be used to a larger extent in certification. It was stated that to a certain extent, formal verification is already part of many certification processes, but in the end the input and the output of a certification is a document in human language and not in a formal language which could be used as a basis for formal verification. So generally, a first step in using formal verification in certification is to formalize the set of security targets appropriately, a hard task in its own.

5.1.3 Liability, Resposibility, and Regulation

We finally turned to the third aspect of the discussion, the big picture involving liability and regulation. The interplay between these issues and certification was a frequent initial question since not every product needs security but it appears that everything connected to the Internet could need some minimal form of security. In this context issues of *negligence*

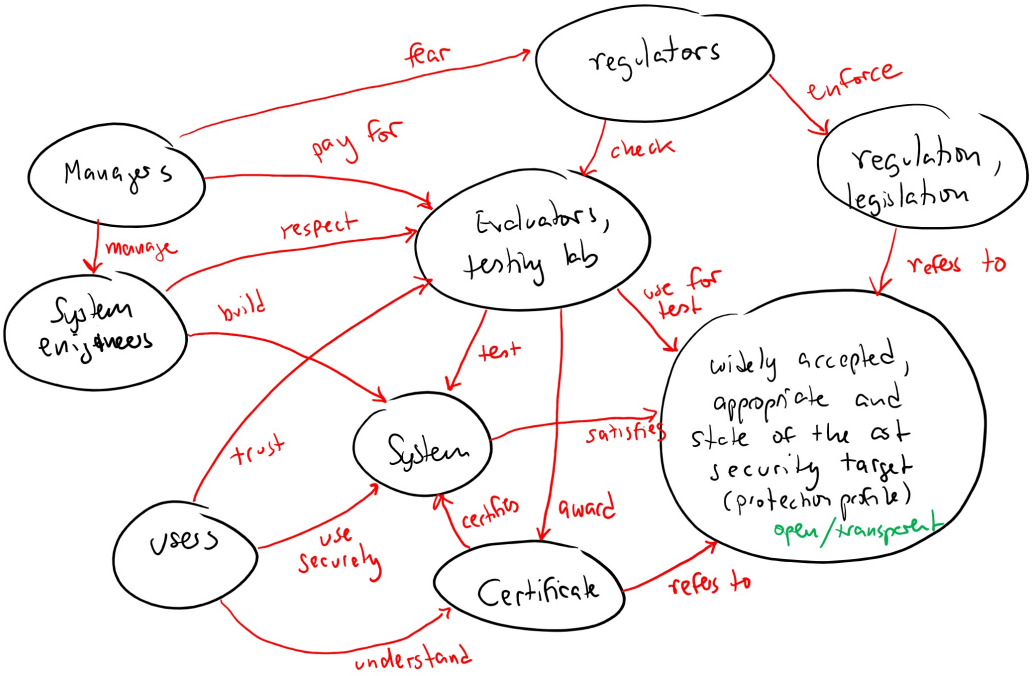

Figure 1 The "ideal world" of certification for security.

were discussed, a term coming from regulation but often introduced into discussions by people from the research community. However, the question is where does negligence start? The problem is that there is no common consensus from the security community. It is important that the security community attempts to interact with the law/regulation community to have more concrete orientation points.

In practice the incentives for certification are various, some involve regulation necessities (like critical infrastructures or GDPR) and risks of law suits agains a company, others involve the risks of bad press and the general problem of naming and shaming that appears to work sometimes at least.

The certification process as a whole was also questioned: Would it be better in certain circumstances to not certify a system at all so as to not create any false expectations? Should we have something more lightweight in-between certification and no certification that is a bit faster but still understandable? It is not so clear what this could be, alltough it would surely be better than performing no security evaluation at all. It is however important to raise no false expectations, as with regular certification. It boils down to knowledge of different levels of assurance, the target of evaluation and the assumptions that come with the evaluation, and an understandable message to the end user and other stakeholders.

5.1.4 Summary

When preparing the results of the discussions during the workshop, the authors of this summary felt that it was easier to summarize the discussed issued based on an understanding of the ideal world in certification for security (see Figure 1, taken from the presentation). In this ideal world, there exist widely accepted, appropriate and state-of-the-art security targets for the system in question that are also openly accessible. Certificates, that are issued by

trusted and independent testing labs or evaluators, can then refer to these targets to test the system. Regulators in turn check the practices of the evaluators to avoid fraudulent certification. In the end, users can then use the system securely.

Obviously, there are many open issues in which the real world differs from the ideal world. Most critically, a set of "widely accepted, appropriate and state-of-the-art security targets" does not exist for most systems, especially security targets that contain measures based on empirical evidence. Furthermore, such security targets are necessary for regulators to define negligence and enforce liability (and motivate managers to pay for certification). In turn, certification standardizes such security targets and can be used for "branding" security, but they still can be misused in the sense of creative and/or fraudulent compliance and misunderstood. Lastly, independent, professional evaluators with high work ethics are needed for trustworthy evaluation. This statement is true even independently of certification.

At the end of the discussions we collected a final round of statements on what participants had taken from the discussions. Here is an unsorted list, that still gives a good insight into the final mental state of the group:

- We need certification but it is unclear how to do this for complex systems.
- We need to define meaningful certifications.
- Currently, certifications are a marketing thing.
- Hardware certification is different from software certification.
- I am skeptical about certification.
- We need a Dagstuhl seminar on certification for security.
- Expectations on certification are too high.
- Certification is better than doing nothing.
- Certification has limited scope, but what is the scope?
- Certificates are necessary.
- Certificates shouldn't lead to blame shifting.

It seems that discussions must continue.

5.2 Education Working Group

Lucy Hunt (Lancaster University, GB), Magnus Almgren (Chalmers University of Technology – Göteborg, SE), Hervé Debar (Télécom SudParis, FR), Fabio di Franco (ENISA – Attica, GR), Sven Dietrich (City University of New York, US), Daisuke Fujimoto (Nara Institute of Science and Technology, JP), Youngwoo Kim (Nara Institute of Science and Technology, JP), Gabriele Lenzini (University of Luxembourg, LU), Olivier Levillain (Télécom SudParis – Evry, FR), Lennert Wouters (KU Leuven, BE), and Moti Yung (Columbia University – New York, US)

5.2.1 Introduction and Approach

Despite enormous efforts, securing IT systems remain an open challenge for both community and industry. Prior to the Dagstuhl seminar, participants identified key security failures and challenges through a "Biggest Failures in Security" survey. From the presented survey results, the group decided on three strategy areas to explore in smaller working groups:

- Certification
- Education
- Human Factors

Working group 2, made up of 11 people, explored education as a strategy to the identified challenges. We reflected that IT security is a multi-disciplinary field, raising questions about our understanding of the population and diversity of engineers – who are the key stakeholders that need educating about security? To make an impact through education we have to understand the audience and effectiveness of channels for sharing and maintaining usable system security mechanisms, knowledge and best practices. We identified educational goals for three stakeholder groups:

- Formally educated engineers
- Non-formally educated engineers
- Industry and the general public

We need to better educate and communicate security knowledge to engineers (software, firmware, hardware, electrical, networking, Internet of Things etc.) that study at university or take formal training. We also need to find ways to identify and reach non-formally educated (e.g. self-trained) engineers – there are many more people than before coding (or such), whose code may have an effect on security. The overarching goal is to make security a good (and easy) thing to do – from the usability for end users to the security design decisions engineers make while building systems. Alongside this, we identified the need for societal change where there is a better understanding, demand and willingness to pay for secure devices, products and services.

5.2.2 Engineers

For engineers, we discussed the need to:

1. Implement incentives and support for educators to demonstrate secure practices and behaviors. This means pointing to secure coding standards and verified (or at least vetted) best practices. In terms of education best practices, successful channels allowed interaction with incident response teams (source: CERT).
2. Identify and train (retrain) IT professionals in security best practices – in particular people that haven't recently or ever been through formal software engineering education, are self-taught software engineers or come in from different fields. All have a need for security resources, training and support. We identified examples of local security education initiatives (CyberEdu in France, Seccap in Japan), the challenge is how to scale globally and so reach larger audiences. We reflected on certification schemes and organisational responsibility for security – as working group 1 were looking at this we parked that discussion.
3. Find ways to attract and train new people into security roles. Security practitioners are sometimes seen as "getting in the way" of the software development life cycle, if security has not been properly integrated into the process previously.
4. Develop better code re-use opportunities, to take advantage of engineer laziness (cut and paste of code samples). To make good practices more accessible, while trying to eradicate bad examples from the publicly available online resources.

We identified a number of solutions, focusing on helping engineers to identify best practices (rather than common practices):

1. Creating tools (e.g. compilers) and methods to make it easier to do the good/right thing. We need software development tools that make security an integral part. Both the creation of more secure code as well as providing feedback to the programmer (software engineer) to transparently move forward with enhancing security aspects are needed.

2. Designing better security mechanisms for engineers, that improve usability for the users. Engineers need to work closely with usability experts to allow better interaction of the users with the hardware devices and their associated operating systems and application software.
3. Teaching engineers how to design user interfaces that help get security concepts across to users. Early studies going back to 1999 showed challenges and confusion when it comes to security concepts. An interdisciplinary approach is needed in training engineers to convey the right ideas. Sample best practices and positive feedback would help reinforce this approach.
4. Enhancing code sharing platforms (such as the web site Stack Overflow): new voting for "best practice secure" answers and code samples so that less experienced programmers understand the choices they make when copying and pasting shared code. By providing accessible and vetted code samples, designs, or approaches, best secure practices would be promoted, while making sure the engineers understand why that choice was made.
5. Industry incentives: "follow these practices – we will rank your app higher". A reward mechanism with software repositories, such as Apple App Store or Google Play Store for mobile and desktop software, could issue a higher ranking for developers that adhere to best practices.

5.2.3 Industry and Public

For industry and the wider public, we need to:

- capture and share IT failures and consequences – to exploit failures and raise awareness
- find approaches to better demonstrate security – to experts, industry and wider society
- motivate people to value and prioritize security requirements

We discussed initiatives for the wider engagement and awareness raising within society including better publicity of vulnerabilities and associated real life failure and success stories – how do we capture and learn from our mistakes? Can the need for security be compared to the climate change movement – can we use society to drive changes?

5.2.4 Further Questions

Other questions raised for further discussion:

1. What has been the impact of GDPR (General Data Protection Regulation) on secure coding practices?
2. Regarding workplace incentives and measures – what metrics are there for secure practices?
3. How has the population and diversity of (software) engineers changed?
4. Is education really failing us in security – how to measure the success and impact of security education?

5.3 Human Factors Working Group

Joachim Meyer (Tel Aviv University, IL), Robert Biddle (Carleton University – Ottawa, CA), Sebastian Pape (Goethe-Universität Frankfurt am Main, DE), Kazue Sako (NEC – Kawasaki, JP), Martina Angela Sasse (Ruhr-Universität Bochum, DE), Stephan Somogyi (Google Inc. – Mountain View, US), Borce Stojkovski (University of Luxembourg, LU), Ingrid Verbauwhede (KU Leuven, BE), and Yuval Yarom (University of Adelaide, AU)

The seminar provided a group of participants with different academic and employment backgrounds with the opportunity to learn and to reflect about what might be the greatest failures and threats in security today. Our specific group dealt with the roles humans and human behavior have in security. The work is based on the premise that the introduction of threats into systems often results from human actions, which may be inadvertent (e.g., the opening of an infected email attachment) or may be deliberate risk taking (e.g., the override of a certificate warning about a site). The group spent several hours discussing ways to address the issue of human involvement in threats. It became clear that this is a complex, multilayered problem, that still warrants a comprehensive conceptual analysis. The group started to discuss the possibility of writing a "cybersecurity harm-reduction manifesto" that would be a synthesis of the different positions brought by the members of the group. In particular, the idea would be to apply ideas from public health by making efforts at a broad level to reduce real harm, rather than offloading the responsibility onto individual users, stigmatizing human behavior and blaming users for any failures.

Major points that arose in the discussions include:

1. Humans are involved in systems in various, often very different capacities (developers, system administrators, security experts, end users, etc.). The knowledge, preferences, and attitudes towards security issues may differ greatly between these groups.
2. The dealing with threats can take various forms, and it is not clear under which conditions, which approach might be best. For instance, one can aim to design out the possibility of threats materializing, one can lower the harm that may be done if a threat materialized, one can train people to detect and cope intelligently with threats, etc. It is not clear how realistic the adoption of different approaches will be to deal with different threats.
3. We still lack well-substantiated knowledge about the effectiveness of different risk-reduction methods. Intuitive approaches (e.g., force users to have very long passwords, which need to be changed every few weeks) often fail to provide the desired results.
4. Security-related behavior is part of a person's interaction with the system. The person's perceptions of risks and the adequacy of different behaviors, the estimates of costs and benefits of different outcomes, and the user's mental model of the system and its security all affect the user's actions and choices. The design of secure systems will also require the design of the interactions that support secure behavior.
5. We still lack theoretical tools to predict the effects, changes in the system, the threats, the environment or the user may have on risk-related behaviors. A major challenge for the scientific work in this field will be to develop and validate such tools.

These points demonstrate the wealth of topics that were discussed and that need to be considered when dealing with the human aspects of security threats and failures. The Dagstuhl seminar can serve as a starting point for discussions and the development of joint research on this broad topic.

Participants

- Tigist Abera
TU Darmstadt, DE
- Magnus Almgren
Chalmers University of Technology – Göteborg, SE
- Frederik Armknecht
Universität Mannheim, DE
- Daniel J. Bernstein
University of Illinois – Chicago, US
- Sarani Bhattacharya
KU Leuven, BE
- Robert Biddle
Carleton University – Ottawa, CA
- Begül Bilgin
Rambus – Rotterdam, NL & KU Leuven, BE
- Dominik Brodowski
Universität des Saarlandes, DE
- Marc C. Dacier
EURECOM – Sophia Antipolis, FR
- Hervé Debar
Télécom SudParis, FR
- Fabio di Franco
ENISA – Attica, GR
- Sven Dietrich
City University of New York, US
- Felix Freiling
Universität Erlangen-Nürnberg, DE
- Daisuke Fujimoto
Nara Institute of Science and Technology, JP
- Lucy Hunt
Lancaster University, GB
- Ghassan Karame
NEC Laboratories Europe – Heidelberg, DE
- Stefan Katzenbeisser
Universität Passau, DE
- Florian Kerschbaum
University of Waterloo, CA
- Youngwoo Kim
Nara Institute of Science and Technology, JP
- Tanja Lange
TU Eindhoven, NL
- Gabriele Lenzini
University of Luxembourg, LU
- Olivier Levillain
Télécom SudParis – Evry, FR
- Volkmar Lotz
SAP Labs France – Mougins, FR
- Michael Meier
Universität Bonn, DE
- Joachim Meyer
Tel Aviv University, IL
- Vasily Mikhalev
Universität Mannheim, DE
- Christian Müller
Universität Mannheim, DE
- Sebastian Pape
Goethe-Universität Frankfurt am Main, DE
- Michalis Polychronakis
Stony Brook University, US
- Kai Rannenberg
Goethe-Universität Frankfurt am Main, DE
- Ahmad-Reza Sadeghi
TU Darmstadt, DE
- Kazue Sako
NEC – Kawasaki, JP
- Martina Angela Sasse
Ruhr-Universität Bochum, DE
- Stephan Somogyi
Google Inc. – Mountain View, US
- Christoph Sorge
Universität des Saarlandes, DE
- Borce Stojkovski
University of Luxembourg, LU
- Ingrid Verbauwhede
KU Leuven, BE
- Melanie Volkamer
KIT – Karlsruher Institut für Technologie, DE
- Edgar Weippl
SBA Research – Wien, AT
- Lennert Wouters
KU Leuven, BE
- Yuval Yarom
University of Adelaide, AU
- Moti Yung
Columbia University – New York, US