Grand Region Security and Reliability Day (GRSRD'16)

March 16, 2016, Nancy, France

Table of Contents

Tracking Personal MicroRNA Expression Profiles over Time 1
Michael Backes, Pascal Berrang, Anne Hecksteden, Mathias Humbert, Andreas Keller and Tim Meyer

The big data deluge in biomedicine: addressing the privacy vs. sharing dilemma 3
Paulo Esteves-Verissimo and Jérémie Decouchant

sElect: A Lightweight Verifiable Remote Voting System 5
Ralf Kuesters, Johannes Mueller, Enrico Scapin and Tomasz Truderung

Two More Efficient Variants of the J-PAKE Protocol 7
Marjan Skrobot, Jean Lancrenon and Qiang Tang

Universal Composition with Responsive Environments 9
Jan Camenisch, Robert R. Enderlein, Stephan Krenn, Ralf Kuesters and Daniel Rausch

A class of precomputation-based distance-bounding protocols 11
Sjouke Mauw, Jorge Toro-Pozo and Rolando Trujillo-Rasua

An Empirical Study on User Access Control in Online Social Networks . . 13
Minyue Ni, Yang Zhang, Weili Han and Jun Pang

Deconstructing MinBFT for Security and Verifiability 15
Vincent Rahli, Francisco Rocha, Marcus Völp and Paulo Esteves-Verissimo

AmpPot: Monitoring and Defending Against Amplification DDoS Attacks 17
Lukas Krämer, Johannes Krupp and Christian Rossow

Micro-Policies for Web Session Security 19
Stefano Calzavara, Riccardo Focardi, Niklas Grimm and Matteo Maffei

A Comprehensive Formal Security Analysis of OAuth 2.0 21
Daniel Fett, Ralf Kuesters and Guido Schmitz

Program Committee

Daniel Fett	University of Trier
Steve Kremer	INRIA Nancy - Grand Est
Ralf Küsters	University of Trier
Stephan Merz	Inria Nancy
Jun Pang	University of Luxembourg
Christian Rossow	Saarland University
Peter Ryan	University of Luxembourg
Dominique Schröder	Saarland University

Tracking Personal MicroRNA Expression Profiles over Time

Michael Backes*†, Pascal Berrang*, Anne Hecksteden§, Mathias Humbert*, Andreas Keller‡ and Tim Meyer¶
*CISPA, Saarland University, lastname@cs.uni-saarland.de, †MPI-SWS
‡Clinical Bioinformatics, Saarland University, andreas.keller@ccs.uni-saarland.de
§Sports Medicine, Saarland University, a.hecksteden@mx.uni-saarland.de
¶Sports Medicine, Saarland University, sportmed@mx.uni-saarland.de

***Abstract*—The decreasing cost of molecular profiling tests, such as DNA sequencing, and the consequent increasing availability of biological data are revolutionizing medicine, but at the same time create novel privacy risks. The research community has already proposed a plethora of methods for protecting *genomic* data against these risks. However, the privacy risks stemming from *epigenetics*, which bridges the gap between the genome and our health characteristics, have been largely overlooked so far, even though epigenetic data such as microRNAs (miRNAs) is no less privacy sensitive. This lack of investigation is attributed to the common belief that the inherent temporal variability of miRNAs shields them from being tracked and linked over time.**

In this work, we show that, contrary to this belief, miRNA expression profiles can be successfully tracked over time, despite their variability. Specifically, we show that two blood-based miRNA expression profiles taken with a time difference of one week from the same person can be matched with a success rate of 90%. We furthermore observe that this success rate stays almost constant when the time difference is increased from one week to one year. In order to mitigate these linkability threats, we propose and thoroughly evaluate two countermeasures: (i) hiding a subset of disease-irrelevant miRNA expressions, and (ii) probabilistically sanitizing the miRNA expression profiles. Our experiments show that the second mechanism provides a better trade-off between privacy and disease-prediction accuracy.

I. Introduction

Since the first sequencing of the human genome in 2001, tens of thousands of genomes and over a million genotypes have been sequenced. The knowledge of our genetic background enables to better predict, and thus anticipate, the risk of developing several diseases, including cancers and cardiovascular and neurodegenerative diseases. Moreover, the genomic research progress enables the development of personalized treatment through pharmacogenomics, studying the effect of the genome on drug response. One of the most important negative counterparts of this genomic revolution is the threat towards genomic privacy [1], [7]. Indeed, genomic data contains very sensitive information about individuals' predisposition to certain severe diseases, about kinship, and about ethnicity, all of which can lead to various sorts of discrimination. Furthermore, genomic data is very stable in time and correlated between family members [5]. Due to these issues, a lot of research has already been carried out to improve the genomic-privacy situation (most of the related literature is surveyed in [3], [9]).

However, our genome is by far not the only element influencing our health. Environmental factors (e.g., pollution, diet, lifestyle,...) often play a crucial role in the development of most common diseases. Epigenetics (or epigenomics), transcriptomics, and proteomics aim to bridge the gap between the genome and our health characteristics. Multi-omics research is a logical complementary step to genome sequencing: the DNA sequence tells us what the cell could possibly do, while the epigenome and transcriptome tell what it is actually doing at a given point in time. Using a computer analogy, if the genome is the hardware, then the epigenome is the software [2].

Despite the growing importance of epigenetics in the biomedical community, privacy concerns stemming from epigenetic data have received little to no attention so far. With the increasing understanding of epigenetics, it becomes clear that epigenetic data contains a vast amount of additional sensitive information, and can thus yield potential privacy risks. For example, major severe diseases (such as cancers, diabetes, or Alzheimer's [4], [6], [11], [12]) are already identified to be affected by epigenetic changes and a recent study stated that epigenetic alterations could even affect sexual orientation [10]. Furthermore, epigenetic data can potentially tell us more about whether someone is carrying a disease at a given point in time, compared to the genome that only informs about the *risk* of getting certain diseases.[1] In this work, we focus on microRNAs, an important element of the epigenome discovered in the early 1990s. MiRNAs are small RNA molecules that regulate the majority of human genes. Studies of miRNA expression profiles have shown that dysregulation of miRNA is linked to neurodegenerative diseases, heart diseases, diabetes and the majority of cancers [4], [6], [8], [11], [12]. Therefore, miRNA expression profiling is a very promising technique that could enable more accurate, earlier and minimally invasive diagnosis of major severe diseases. As a consequence, it will certainly be increasingly used in medical practice. It is widely believed in the biomedical community that the miRNA expression levels are varying sufficiently to invalidate any linkability attempts over time. This work, however, shows the contrary: despite their temporal variability, microRNA expression profiles are still identifiable after time periods of several months.

[1]The only exception to this rule are Mendelian disorders, such as cystic fibrosis, which are largely determined by our genes.

II. Attacks

We study here the temporal linkability of personal miRNA expression profiles by presenting and evaluating two different attacks. We consider a passive adversary who can get access to miRNA expression levels of one or multiple individuals and wants to match them with other miRNA expression levels at some point in time. This epigenetic information could be collected online (publicly shared by the research community, like in the Gene Expression Omnibus), or be leaked through a major security breach, e.g., of a hospital server. We first study an *identification attack*, which pinpoints a specific miRNA expression profile in a database of multiple expression profiles by knowing the targeted profile at another point in time. Second, we study a *matching attack*, which tracks a set of miRNA expression profiles over time.

We rely on principal component analysis to pre-process the miRNA expression levels, and on a maximum weight assignment algorithm for the matching attack. We thoroughly evaluate our linkability attacks by using three different longitudinal datasets: (i) the blood-based miRNA expression levels of 29 athletes at two time points separated by one week, (ii) the plasma-based miRNA expression levels of the same athletes at two time points separated by one week, and (iii) the plasma-based miRNA expression levels of 26 patients with lung cancer over more than 18 months and eight time points. Our experimental results notably show that blood miRNA expression profiles are about twice as easy to track over time than plasma miRNA profiles, and that the matching attack is more successful than the identification attack: We reach a success rate of 90% with blood and a success rate of 48% with plasma miRNAs in the matching attack whereas, in the identification attack, we reach a success rate of 76% with blood and 28% with plasma miRNAs. Moreover, we demonstrate that only 10% of the miRNAs are sufficient to achieve similar success rates as with all miRNAs. With the third dataset containing plasma-based miRNA expression profiles, we observe that the attack achieves a similar success rate from one-week time shifts to 12-month time shifts.

III. Defenses

We present two defense mechanisms to counter the linkability of personal miRNA expression profiles: (i) hiding a subset of the miRNA expressions, e.g., those that are not relevant for medical practice, and (ii) disclosing noisy miRNA expression profiles by adding noise in a differentially private and distributed manner. While the first countermeasure would especially be useful in a clinical setting, in which the disease-relevant miRNAs are already known, the second countermeasure is intended to be better suited for the biomedical research community. In this context, as one of the objective is to discover associations between miRNAs and diseases, it is impossible to restrict the released data to only a few miRNAs.

We evaluate our protection mechanisms with the first aforementioned blood-based miRNA profiles of athletes and a fourth, also blood-based, miRNA dataset of more than 1,000 participants that includes information about 19 diseases (at a single point in time). The former is used to measure how temporal linkability is reduced with our countermeasures, whereas the latter helps us evaluate the evolution of accuracy (i.e., utility) in predicting patients' diseases from their miRNA expressions with a support vector machine (SVM) algorithm. The experiments show that it is possible to decrease linkability by at least 50% for almost no loss of accuracy ($< 1\%$) for the majority of diseases with the noise mechanism. Moreover, our results demonstrate that the noise mechanism provides better privacy-utility trade-offs than the hiding method in 17 out of 19 of diseases, while allowing more flexibility in the data usage for biomedical researchers. This finding is reinforced by the fact that an adversary can use correlations between miRNA expressions to infer more miRNAs than those actually shared with the first countermeasure.

IV. Conclusion

This work demonstrates that personal miRNA expression profiles can be successfully tracked over time, especially when these expressions are measured from blood samples. This study sheds light on a widely overlooked problem, privacy risks stemming from epigenetic data, bringing it to the attention of both the biomedical and computer security research communities. Note also that our linkability attacks can also be used for preventing the mixing of miRNA samples in the clinical setting. Our work also shows that probabilistically sanitizing the expression profiles is a promising technique that could be applied on other types of longitudinal biomedical data to enhance the privacy of their owners.

References

[1] E. Ayday, E. De Cristofaro, J.-P. Hubaux, and G. Tsudik, "Whole genome sequencing: Revolutionary medicine or privacy nightmare?" *Computer*, pp. 58–66, 2015.

[2] J. Cloud, "Why your DNA isn't your destiny," *Time*, January 2010.

[3] Y. Erlich and A. Narayanan, "Routes for breaching and protecting genetic privacy," *Nature Reviews Genetics*, vol. 15, pp. 409–421, 2014.

[4] A. P. Feinberg and M. D. Fallin, "Epigenetics at the crossroads of genes and the environment," *JAMA*, vol. 314, pp. 1129–1130, 2015.

[5] M. Humbert, E. Ayday, J.-P. Hubaux, and A. Telenti, "Addressing the concerns of the Lacks family: quantification of kin genomic privacy," in *Proceedings of the 2013 ACM SIGSAC CCS*, 2013, pp. 1141–1152.

[6] P. A. Jones and S. B. Baylin, "The epigenomics of cancer," *Cell*, vol. 128, pp. 683–692, 2007.

[7] Z. Lin, A. B. Owen, and R. B. Altman, "Genomic research and human subject privacy," *SCIENCE-NEW YORK THEN WASHINGTON-*., pp. 183–183, 2004.

[8] J. Lu, G. Getz, E. A. Miska, E. Alvarez-Saavedra, J. Lamb, D. Peck, A. Sweet-Cordero, B. L. Ebert, R. H. Mak, A. A. Ferrando *et al.*, "Microrna expression profiles classify human cancers," *nature*, vol. 435, no. 7043, pp. 834–838, 2005.

[9] M. Naveed, E. Ayday, E. W. Clayton, J. Fellay, C. A. Gunter, J.-P. Hubaux, B. A. Malin, and X. Wang, "Privacy in the genomic era," *ACM Computing Surveys (CSUR)*, vol. 48, p. 6, 2015.

[10] T. Ngun et al., "Abstract: A novel predictive model of sexual orientation using epigenetic markers," in *American Society of Human Genetics 2015 Annual Meeting*, 2015.

[11] I. A. Qureshi and M. F. Mehler, "Advances in epigenetics and epigenomics for neurodegenerative diseases," *Current neurology and neuroscience reports*, vol. 11, pp. 464–473, 2011.

[12] L. D. Wood, D. W. Parsons, S. Jones, J. Lin, T. Sjöblom, R. J. Leary, D. Shen, S. M. Boca, T. Barber, J. Ptak *et al.*, "The genomic landscapes of human breast and colorectal cancers," *Science*, vol. 318, pp. 1108–1113, 2007.

The big data deluge in biomedicine: addressing the privacy vs. sharing dilemma

Paulo Esteves-Verissimo and Jérémie Decouchant
CritiX Lab - Critical and Extreme Security and Dependability
SnT - Interdisciplinary Centre for Security, Reliability and Trust
University of Luxembourg
Email: ⟨name⟩.⟨surname⟩@uni.lu

***Abstract*—This position paper discusses on-going work on architectures and algorithms for efficient but privacy-preserving storage and analysis of bulk biomedical data.**

I. Introduction

Biomedical information is enduring a revolution: collection and storage of biological material is getting systematic (tissues, fluids, etc.), both for clinical and research purposes; digital representations of these samples are exploding in volume, especially in genomics (DNA) thanks to a large extent to the advent of the so-called Next-Generation-Sequencing (NGS) machines. These machines have lowered the price and increased the speed of sequencing, by several orders of magnitude. Over the past few years, the number of sequenced genomes or exomes (parts thereof) has sky-rocketed, and the trend is there to continue. Bottomline: (i) full-genome sequencing for less than one thousand euros is becoming a reality; and (ii) the life-cycle of the physical biological material stored in biobanks is too expensive to sustain the current growth. In consequence, every day we have quite a few additional dozens of terabytes of stored raw digital data, and the combined scale of biomedical datasets and related data from their stakeholders, has in fact entered the row of big data.

II. Problems on the horizon

There are fantastic opportunities in this new world, but there are at least as many threats and challenges. We just enumerate the few ones we consider of most importance, starting by introducing their causes.

First, the need for economically storing and processing these huge amounts of data has put cloud computing on the agenda, inclusively by NGS machine vendors [1]. Not just any cloud, but including public clouds, and not just very restricted access, but including Internet web-based access [14]. Using common (and moderate security) IT techniques to manipulate such critical data, brings about considerable security and privacy risks whose likelihood and impact have perhaps not been correctly evaluated so far, given the recurring failures in the internet/cloud complex [9, 15, 5].

Second, a dramatic increase of the availability of personally identifiable information (PII) has been occurring in parallel, due not only but largely to concurrent effects like: the (sometimes forced, e.g. by governments or companies) digitalization of society activity; the web in general; and social networks activity in particular. Our lifes leave an evergrowing indelible digital trace, and this has had an effect whose consequences we are still starting to comprehend: big data in this case means that there are too many data around, and statistical correlations which were infeasible a few years ago, become trivial, and with astonishing precision and recall. Recent happenings go from real-life episodes as reported in [12], to impressive re-identification of private credit card operations, as in [13]. But research results in the context of biomedical data are much more worrying. In 2000, an alarm was raised [16], by demonstrating the re-identifiability of de-identified patient-specific medical data, and thus showing the ineffectiveness of the de-identification methods used. Thirteen years later, not much had changed in the meantime, since the work in [7] managed to re-identify more than 10% of the de-identified 1000-Genomes project database, generously built from anonymous donors' sequenced DNA, again because useful correlations could be found with the judged anonymous metadata in the genomes database.

Last but not least, there is a great pressure to get hold of biomedical data, by reasons of different nature, and coming from diverse angles, such as researchers, corporations and even governments. This confluence of interests is sometimes detrimental of the investigation and deployment of clear and sound strategies, policies and technologies that help solve the problem at hand in this paper: addressing the privacy vs. sharing dilemma we face with regard to biomedical data.

The remainder of the paper discusses some contributions in that line.

III. A framework for solutions

In a recent paper [17], we advanced an architectural framework to guide possible solutions to a range of problems around the biomedical data scenario. We predicted the advent of the new era of *e-Biobanking*, in terms of the "creation of genuine hybrid ecosystems composed of interplaying physical and computer storage and processing infrastructures, handling physical and computerized samples of biological data, in a symbiotic and seamless way". The vision, depicted in Figure 1, foresaw that systems originally decoupled from each other (Fig. 1a), would progressively evolve toward coalitions of interested stakeholders, organised around hybrid and/or federated cloud computing technologies (Fig. 1b), and where techniques

This work is partially supported by the University of Luxembourg - SnT and by the Fonds National de la Recherche Luxembourg (FNR) through PEARL grant FNR/P14/8149128.

[1] See, e.g., BaseSpace from Illumina http://tinyurl.com/zwjtqts

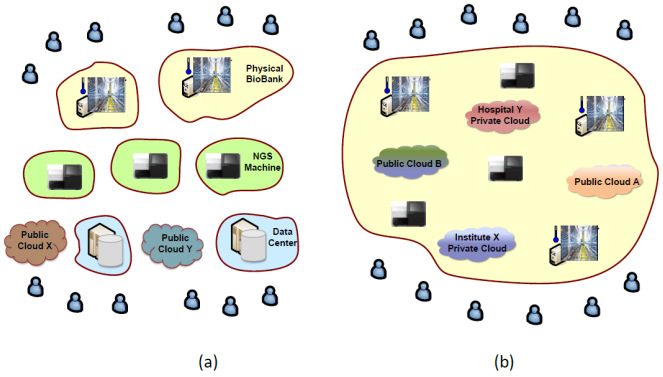

Fig. 1. The *e-Biobanking vision*

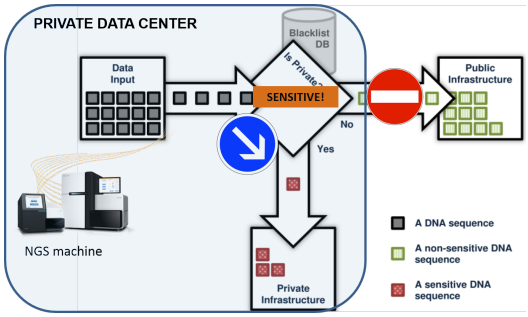

Fig. 2. Privacy filtering: automatic segregation of NGS data

safeguarding security and dependability might be effectively deployed, whilst promoting the desired sharing.

The vision has been slowly coming to fruition. For example, the BioBankCloud project proposes storage architectures which are based on clouds-of-clouds, that is, multiple instances of clouds, both private and public, from several stakeholders and providers, but which are perceived seamlessly as a single cloud, by the e-biobank users and administrators [3].

In [1], researchers propose a federated system enabling the cooperative analysis of NGS sequences across multiple locations. Despite the security mechanisms being at the level of standard IT, this is an interesting step forward.

In another work [2], the authors propose a privacy-preserving method where they encrypt all genomic data before storing it, and manually mask the few genomic variations identified as sensitive. Despite the possible coverage gaps and errors of the manual process, and update problems, this method yields prevention at early stages of the life-cycle.

In some recent work, we and colleagues at the University of Lisboa, have proposed a high-throughput method to automatically segregate genomic information right after it is created, that is, at the exit of NGS machines [6]. The proposed scheme, depicted in Figure 2, fits the e-biobanking vision nicely: very sensitive data is kept within the private premises of the entity generating the data, e.g., a secure data center next to the NGS machines subsystems, segregated from the outside, whereas less sensitive data can be stored in cloud systems of a lower privacy/security category. All this is done automatically, through a rule-based system acting pretty much like an intrusion detection system, and with high-throughput, by resorting to a Bloom filter.

Part of this data, despite being less sensitive, may still have considerable privacy requirements, and can for example be stored in public cloud-of-clouds systems, protected with powerful state-of-the art encryption, coding and dispersion mechanisms, such as those proposed in [4].

IV. Conclusions and Future Work

The classical DNA workflows include, but are not limited to, alignments of reads to a reference DNA, de novo assembly to reconstitute the DNA of a specie from scratch, quality control. We plan to begin with a distributed algorithm to compute the alignment of reads to a reference DNA, which would be protected by mechanisms that would prevent an attacker, or a faulty machine, to obtain useful information.

Second, in the past, alignment algorithms have been optimized for short reads of around 150 bases. These specialized algorithms include BWA [11] and Bowtie [10]. However, very recent sequencers [8] seem to produce again longer reads. We plan to study the impact of this technological shift.

References

[1] Amin Ardeshirdavani et al. “NGS-Logistics: federated analysis of NGS sequence variants across multiple locations”. In: *Genome Medicine* 6.9 (2014), pp. 1–11.

[2] Erman Ayday et al. “Privacy-preserving processing of raw genomic data”. In: *Data Privacy Management and Autonomous Spontaneous Security*. 2014, pp. 133–147.

[3] Alysson Bessani et al. “BiobankCloud: a platform for the secure storage, sharing, and processing of large biomedical data sets”. In: *Workshop on Data Management and Analytics for Medicine and Healthcare*. 2015.

[4] Alysson Bessani et al. “DepSky: Dependable and Secure Storage in a Cloud-of-clouds”. In: *Proceedings of the Sixth Conference on Computer Systems*. 2011.

[5] Chad Brooks. *Cloud Storage Often Results in Data Loss*. 2011.

[6] Vinicius V Cogo et al. “A high-throughput method to detect privacy-sensitive human genomic data”. In: *ACM Workshop on Privacy in the Electronic Society*. 2015.

[7] Melissa Gymrek et al. “Identifying Personal Genomes by Surname Inference”. In: *Science* 339.6117 (2013), pp. 321–324.

[8] CLC Ip et al. “MinION Analysis and Reference Consortium: Phase 1 data release and analysis”. In: *F1000Research* 4 (2015), pp. 1–30.

[9] Peter Judge. *Google has cloud failure, applies quick fix*. 2015.

[10] Ben Langmead et al. “Ultrafast and memory-efficient alignment of short DNA sequences to the human genome”. In: *Genome biol* 10.3 (2009), R25.

[11] Heng Li and Richard Durbin. “Fast and accurate short read alignment with Burrows–Wheeler transform”. In: *Bioinformatics* 25.14 (2009), pp. 1754–1760.

[12] Gus Lubin. *The Incredible Story Of How Target Exposed A Teen Girl’s Pregnancy*. 2012.

[13] Yves-Alexandre de Montjoye et al. “Unique in the shopping mall: On the reidentifiability of credit card metadata”. In: *Science* 347.6221 (2015), pp. 536–539.

[14] Lincoln D. Stein. “The case for cloud computing in genome informatics”. In: *Genome Biology* 11.5 (2010), p. 207.

[15] David Streitfeld. *Google Concedes That Drive-By Prying Violated Privacy*. 2013.

[16] Latanya Sweeney. “Simple demographics often identify people uniquely”. In: *Health (San Francisco)* 671 (2000), pp. 1–34.

[17] Paulo E. Verissimo and Alysson Bessani. “E-biobanking: What Have You Done to My Cell Samples?” In: *Security Privacy* 11.6 (2013), pp. 62–65.

sElect: A Lightweight Verifiable Remote Voting System

Ralf Küsters*, Johannes Müller*, Enrico Scapin*, Tomasz Truderung†
*University of Trier
†Polyas

E-voting systems are used in many countries for national or municipal elections as well as for elections within associations, societies, and companies. There are two main categories of such systems. In the first category, voters vote in polling stations using electronic voting machines, such as direct recording electronic voting systems or scanners. In the second category, called remote electronic voting, voters vote over the Internet using their own devices (e.g., desktop computers or smartphones). In addition, there are hybrid approaches, where voters, via an additional channel, e.g., mail, are provided with codes which they use to vote (code voting).

E-voting systems are complex hardware/software systems and as in all such systems programming errors can hardly be avoided. In addition, these systems might deliberately be tampered with when deployed in elections. This means that voters when using e-voting systems, in general, do not have any guarantee that their votes were actually counted and that the published result is correct, i.e., reflects the actual voters' choices. In fact, many problems have been reported (see, e.g., [1], [13]). Therefore, besides vote privacy, modern e-voting systems strive for what is called *verifiability*. This security property requires that voters are able to check the above, i.e., proper counting of their own votes and integrity of the overall result, even if voting machines/authorities are (partially) untrusted.

Several such e-voting systems have been proposed in the literature, including, for example, such prominent systems as Helios [3], Prêt à Voter [12], and STAR-Vote [5]. Some systems, such as Civitas [7] and Scantegrity [6], are designed to, in addition, even achieve *coercion-resistance*, which requires that vote selling and voter coercion is prevented. Several of these systems have been used in binding elections (see, e.g., [4], [6], [8]). In this work, we are interested in remote electronic voting, which is meant to enable the voter to vote via the Internet.

The design of practical remote e-voting systems is very challenging as many aspects have to be considered. In particular, one has to find a good balance between simplicity, usability and security. This in turn very much depends on various, possibly even conflicting requirements and constraints, for example: What kind of election is targeted? National political elections or elections of much less importance and relevance, e.g., within clubs or associations? Should one expect targeted and sophisticated attacks against voter devices and/or servers, or are accidental programming errors the main threats to the integrity of the election? Is it likely that voters are coerced, and hence, should the system defend against coercion? How heterogeneous are the computing platforms of voters? Can voters be expected to have/use a second (trusted) device and/or install software? Is a simple verification procedure important, e.g., for less technically inclined voters? Should the system be easy to implement and deploy, e.g., depending on the background of the programmers? Should authorities and/or voters be able to understand (to some extent) the inner workings of the system?

Therefore, there does not seem to exist a "one size fits all" remote e-voting system. In this work, we are interested in systems for low-risk elections, such as elections within clubs and associations, rather than national elections, where—besides a reasonable level of security—simplicity and convenience are important.

The goal of this work is to design a particularly lightweight remote system which (still) achieves a good level of security. The system is supposed to be lightweight both from a voter's point of view and a design/complexity point of view. For example, we do not want to require the voter to install software or use a second device. Also, verification should be a very simple procedure for a voter or should even be completely transparent to the voter. More specifically, the main contributions of our work are as follows.

Contributions of our work. We present a new, particularly lightweight remote e-voting system, called *sElect* (secure/simple elections), which we implemented as a platform independent web application and for which we perform a detailed cryptographic security analysis w.r.t. privacy of votes as well as verifiability and accountability. The system combines several concepts, such as verification codes (see, e.g., [9]) and Chaumian mix nets, in a novel way. sElect is not meant to defend against coercion and mostly tries to defend against untrusted or malicious authorities, including inadvertent programming errors or deliberate manipulation of servers, but excluding targeted and sophisticated attacks against voters' devices.

We briefly sketch the main characteristics of sElect, including several novel and unique features and concepts which should be beneficial also for other systems. The full version of this paper contains a more detailed discussion, including the limitations of sElect.

Fully automated verification. One of the important unique fea-

tures of sElect is that it supports fully automated verification. This kind of verification is carried out by the voter's browser. It does not require any voter interaction and is triggered as soon as a voter looks at the election result. This is meant to increase verification rates and ease the user experience. As voters are typically interested in the election results, combining the (fully automated) verification process with the act of looking at the election result in fact appears to be an effective way to increase verification rates as indicated by two small mock elections we performed with sElect. In a user study carried out in [2] for various voting systems, automated verification was pointed out to be lacking in the studied systems, including, for example, Helios. It seems that our approach of automated verification should be applicable and can be very useful for other remote e-voting systems, such as Helios, as well.

Another important aspect of the automated verification procedure of sElect is that it performs certain cryptographic checks and, if a problem is discovered, it singles out a specific misbehaving party and produces binding evidence of the misbehavior. This provides a high level of accountability and deters potentially dishonest voting authorities.

Voter-based verification (human verifiability). Besides fully automated verification, sElect also supports a very easy to understand manual verification procedure: a voter can check whether a verification code she has chosen herself when casting her vote appears in the election result along with her choice. This simple procedure has several obvious benefits. For example, it reduces trust assumptions concerning the voter's computing platform (for fully automated verification the voter's computing platforms needs to be fully trusted). Also voter's can easily grasp the procedure and its purpose, essentially without any understanding of the rest of the system, which should help to increase user satisfaction and verification rates. On the negative side, such codes open the way for voter coercion.

Simple cryptography and design. Unlike other modern remote voting systems, sElect uses only the most basic cryptographic operations, namely, public key encryption and digital signatures. And the overall design and structure of sElect is simple as well. In particular, sElect does *not* rely on any more sophisticated cryptographic operations, such as zero-knowledge proofs, verifiable distributed decryption, universally verifiable mix nets, etc. Our motivation for this design choice is twofold.

Firstly, we wanted to investigate what level of security (privacy, verifiability, and accountability) can be obtained with only the most basic cryptographic primitives (public-key encryption and digital signatures) and a simple and user-friendly design.

Secondly, using only the most basic cryptographic primitives has several advantages (but also some disadvantages), as further discussed in the full version.

Rigorous cryptographic security analysis. We perform rigorous cryptographic analysis of sElect w.r.t. end-to-end verifiability, accountability, and privacy. Since quite rarely implementations of practical e-voting systems come with rigorous cryptographic analysis, this is a valuable feature by itself.

Our cryptographic analysis shows that sElect enjoys a good level of security, given the very basic cryptographic primitives it uses.

Remarkably, the standard technique for achieving (some level of) end-to-end verifiability is to establish both so-called individual and universal verifiability.[1] In contrast, sElect demonstrates that one can achieve (a certain level of) end-to-end verifiability, as well as accountability, without universal verifiability. This is interesting from a conceptual point of view and may lead to further new applications and system designs.

Altogether, sElect is a remote e-voting system for low-risk elections which provides a new balance between simplicity, usability, and security, emphasizing simplicity and usability, and by this, presents a new option for remote e-voting. Also, some of its new features, such as fully automated verification and triggering verification when looking up the election result, could be used to improve other systems, such as Helios, and lead to further developments and system designs.

References

[1] http://www.computerworld.com/s/article/9233058/Election_watchdogs_keep_wary_eye_on_paperless_e_voting_systems, October 30th 2012.

[2] C. Acemyan, P. Kortum, M. Byrne, D. Wallach. Usability of Voter Verifiable, End-to-end Voting Systems: Baseline Data for Helios, Prêt à Voter, and Scantegrity II. *USENIX Journal of Election Technology and Systems (JETS)*, 2014.

[3] Ben Adida. Helios: Web-based Open-Audit Voting. In *USENIX 2008*, pages 335–348. USENIX Association, 2008.

[4] Ben Adida, Olivier de Marneffe, Olivier Pereira, and Jean-Jaques Quisquater. Electing a University President Using Open-Audit Voting: Analysis of Real-World Use of Helios. In *(EVT 2009)*, 2009.

[5] S. Bell, J. Benaloh, M. Byrne, D. DeBeauvoir, B. Eakin, G. Fischer, Ph. Kortum, N. McBurnett, J. Montoya, M. Parker, O. Pereira, Ph. Stark, D. Wallach, and M. Winn. STAR-Vote: A Secure, Transparent, Auditable, and Reliable Voting System. *USENIX Journal of Election Technology and Systems (JETS)*, 1:18–37, August 2013.

[6] R. Carback, D. Chaum, J. Clark, adn J. Conway, E. Essex, P.S. Herrnson, T. Mayberry, S. Popoveniuc, R. L. Rivest, E. Shen, A. T. Sherman, and P.L. Vora. Scantegrity II Municipal Election at Takoma Park: The First E2E Binding governmental Elecion with Ballot Privacy. In *USENIX 2010*. USENIX Association, 2010.

[7] M. R. Clarkson, S. Chong, and A. C. Myers. Civitas: Toward a Secure Voting System. In *S&P 2008*, pages 354–368. IEEE Computer Society, 2008.

[8] Chris Culnane, Peter Y. A. Ryan, Steve Schneider, and Vanessa Teague. vVote: a Verifiable Voting System (DRAFT). *CoRR*, abs/1404.6822, 2014. Available at http://arxiv.org/abs/1404.6822.

[9] Richard A. DeMillo, Nancy A. Lynch, and Michael Merritt. Cryptographic Protocols. In *STOC 1982*, pages 383–400. ACM, 1982.

[10] Ralf Küsters, Tomasz Truderung, and Andreas Vogt. Verifiability, Privacy, and Coercion-Resistance: New Insights from a Case Study. In *S&P 2011*, pages 538–553. IEEE Computer Society, 2011.

[11] Ralf Küsters, Tomasz Truderung, and Andreas Vogt. Clash Attacks on the Verifiability of E-Voting Systems. In *S&P 2012*, pages 395–409. IEEE Computer Society, 2012.

[12] P. Y. A. Ryan, D. Bismark, J. Heather, S. Schneider, and Z. Xia. The Prêt à Voter Verifiable Election System. Technical report, Universities of Luxembourg and Surrey, 2010. http://www.pretavoter.com/publications/PretaVoter2010.pdf.

[13] D. Springall, T. Finkenauer, Z. Durumeric, J. Kitcat, H. Hursti, M. MacAlpine, and J. A. Halderman. Security Analysis of the Estonian Internet Voting System. In *CCS 2014*, pages 703–715. ACM, 2014.

[1] As pointed out in [10], this combination does not guarantee end-to-end verifiability, though.

Two More Efficient Variants of the J-PAKE Protocol

Marjan Skrobot
SnT, University of Luxembourg
Email: marjan.skrobot@uni.lu

Jean Lancrenon
SnT, University of Luxembourg
Email: jean.lancrenon@uni.lu

Qiang Tang
SnT, University of Luxembourg
Email: qiang.tang@uni.lu

Abstract*—Recently, the password-authenticated key exchange protocol J-PAKE of Hao and Ryan (Workshop on Security Protocols 2008) was formally proven secure in the algebraic adversary model by Abdalla et al. (IEEE S&P 2015). In this paper, we propose and examine two variants of J-PAKE - which we call RO-J-PAKE and CRS-J-PAKE - that each makes the use of two less zero-knowledge proofs than the original protocol. We show that they are provably secure following a similar strategy to that of Abdalla et al. We also study their efficiency as compared to J-PAKE's, also taking into account how the groups are chosen. Namely, we treat the cases of subgroups of finite fields and elliptic curves. Our work reveals that, for subgroups of finite fields, CRS-J-PAKE is indeed more efficient than J-PAKE, while RO-J-PAKE is much less efficient. On the other hand, when instantiated with elliptic curves, both RO-J-PAKE and CRS-J-PAKE are more efficient than J-PAKE, with CRS-J-PAKE being the best of the three. We illustrate this experimentally, making use of recent research by Brier et al. (CRYPTO 2010). Regardless of implementation, we note that RO-J-PAKE enjoys a looser security reduction than both J-PAKE and CRS-J-PAKE. CRS-J-PAKE has the tightest security proof, but relies on an additional trust assumption at setup time. We believe our results can be useful to anyone interested in implementing J-PAKE, as perhaps either of these two new protocols may also be options, depending on the deployment context.

I. Introduction

The objective of *Password-Authenticated Key Exchange* (PAKE) is to allow secure authenticated communication over insecure networks between two or more parties who only share a low-entropy password. Many different protocols have been proposed in the literature to accomplish this. Among them, the J-PAKE protocol [1] has been implemented due to its patent-free nature.

J-PAKE is quite unique because it integrates Non-Interactive Zero-Knowledge proofs of knowledge (NIZKs in the rest of the paper) - specifically, Schnorr proofs of knowledge [2] - effectively into its design. However, the presence of these proofs is actually one of the main arguments of J-PAKE's detractors: Indeed, they add more exponentiations to a protocol that already contains many. A question that can be asked therefore is whether variants of J-PAKE using less proofs of knowledge can be found, and how they compare in terms of efficiency to the original protocol.

A. Our Contribution

We answer these questions by exhibiting two new protocols - which we call RO-J-PAKE and CRS-J-PAKE - that are very similar to J-PAKE, but each use two less zero-knowledge proofs. We explicitly prove the security of RO-J-PAKE, following a similar strategy to that of Abdalla et al. in their recent analysis of J-PAKE [3], and show how the proof can be adapted to the case of CRS-J-PAKE. We also provide a more refined analysis of these protocols' efficiency relative to J-PAKE's. We do this by explicitly examining costs depending on which groups are used to deploy the protocol. This is especially important for RO-J-PAKE, since it requires hashing into the group in question. Indeed, while on paper, this appears to have no incidence, in practice it requires some attention. We treat the cases of Elliptic Curve (EC) groups and Subgroups of Finite Fields (SFFs), since all three protocols require the Decisional Diffie-Hellman (DDH) assumption to hold. In more detail, our findings are as follows.

In terms of provable security: RO-J-PAKE and CRS-J-PAKE are asymptotically as secure as J-PAKE, and against the same kind of adversaries, namely, algebraic adversaries. However, RO-J-PAKE enjoys a looser security proof than J-PAKE and CRS-J-PAKE, essentially because of the addition of a random oracle. CRS-J-PAKE has the tightest proof of the three protocols.

In terms of computational and communication efficiency: The apparent computational gain in efficiency that RO-J-PAKE and CRS-J-PAKE enjoy due to their having two less zero-knowledge proofs than J-PAKE can be summarized as follows:

- When all three protocols are instantiated with ECs, CRS-J-PAKE and RO-J-PAKE cost a total of about 8 group-sized exponentiations less than J-PAKE. CRS-J-PAKE has a slight edge over RO-J-PAKE, because the latter requires hashing into an EC group. However, experimental results (see Table I) using recent research by Brier et al. [4] shows that this edge can be practically ignored.
- When all three protocols are instantiated with SFFs, CRS-J-PAKE takes 8 group-sized exponentiations less than J-PAKE, but RO-J-PAKE suffers from two additional exponentiations of size comparable to that of the base field's prime - which is typically way larger than the actual group - thus making it much less efficient than J-PAKE in practice. This is also due to the need to hash into a SFF. Thus, unless an efficient hashing method is devised, this instantiation of RO-J-PAKE may only have theoretical interest.
- Regardless of the group instantiation, both RO-J-PAKE and CRS-J-PAKE are more efficient than J-PAKE in

terms of communication, as they both send four less group elements and two less scalars than J-PAKE does.

RO-J-PAKE and CRS-J-PAKE have a few other (dis)advantages related to their deployability, and that are worth mentioning. See Section III for more details.

II. Protocols.

Here we present the RO-J-PAKE and CRS-J-PAKE protocol informally. In the description below, we will assume that the client and server always check if the received message is well-formed and if the validity of NIZK proof holds under appropriate label.

A. The RO-J-PAKE Protocol

After initialization in which public parameters are fixed and a password different from zero is shared between the client and server, the protocol runs in two phases. In the first phase, each party generates one group element and corresponding NIZK proof and sends them – along with its ID – to the other party. In the second phase, upon receiving the first message, both parties compute a common base D as $H_0(A, B, X_1, X_2)$. Then, each party computes and sends to other party its commit message that consists of $\alpha := (DX_2)^{x_1 pw}$ and corresponding NIZK proof π_α under label l_A in case of client, and $\beta := (DX_1)^{x_2 pw}$ and π_β under label l_B in case of server. Upon receipt of the second message, each party derives a shared secret K, which should be an element of group $\mathbb{G}$, and then a bit-string sk, which will act as a session key.

B. The CRS-J-PAKE Protocol

The observation that RO-J-PAKE's D value can be in a sense replaced by a random group element that neither party has control over can be exploited by adding to the protocol's setup a randomly generated value $U \in \mathbb{G}$ that is fixed once and for all. Hence, in CRS-J-PAKE, the value U plays the role of D from RO-J-PAKE for all protocol executions. Since we no longer need to hash into the underlying group, in contrast to RO-J-PAKE, CRS-J-PAKE has no efficiency issues with respect to a hash implementation. However, the need to generate and trust the hard-coded value U poses its own deployment issues (see Section III).

III. Practical Considerations

In theory, for J-PAKE and the two new variants, the modular exponentiations are the predominant factors in the computation. Hence, the computational cost is estimated based on counting the number of such modular exponentiations. Referring to the protocol specifications in Sec. II-A and II-B, we summarize their complexities in Table I.

In practice however, counting the modular exponentiations is insufficient, in particular for RO-J-PAKE. This is because the true speed depends highly on how H_0 - which lands into the protocol's underlying group - is computed. Thus, we further discuss the computational complexity with respect to two different instantiations.

TABLE I
The efficiency comparison between J-PAKE, RO-J-PAKE and CRS-J-PAKE.

Protocol	Complexity	
	Communication	Computation
J-PAKE	$12 \times \mathbb{G} + 6 \times \mathbb{Z}_q$	$28\|q\|$-bit exp
RO-J-PAKE	$8 \times \mathbb{G} + 4 \times \mathbb{Z}_q$	$20\|q\|$-bit exp + 2 H_0
CRS-J-PAKE	$8 \times \mathbb{G} + 4 \times \mathbb{Z}_q$	$20\|q\|$-bit exp

- **SFF instantiation.** Here, we assume that $\mathbb{G}$ is deployed as the q-order subgroup of $GF(p)^*$, where $p = rq + 1$ and p and q are both prime. Thus, we have $|r| = |p| - |q|$. Standard techniques implement H_0 by first hashing into $GF(p)^*$, which is truly cheap, and then *exponentiating the result by* r, which depends on $|r|$. In particular, J-PAKE is more efficient than RO-J-PAKE if and only if $28|q| \leq 20|q| + 2|r|$, i.e. if and only if $4|q| \leq |r|$. In other words, J-PAKE is better than RO-J-PAKE provided that a single $|r|$-bit exponentiation costs more than 4 $|q|$-bit ones. Since in general, $|r|$-bit exponentiations cost *way more* than that, J-PAKE is definitely the better option when using SFFs. Note that CRS-J-PAKE would be better than J-PAKE in this setting.
- **EC instantiation.** In our experiment, we assumed the EC is over prime field $GF(p)$ with $|p| = 256$, and took $\mathbb{G}$ to be an EC group of prime order q with $|q| = 160$. H_0 was implemented using the recently discovered hashing algorithms of Brier et al. [4]. We found that an exponentiation took on average more time than hashing a message into the EC group. Hence, using ECs, both RO-J-PAKE and CRS-J-PAKE are definitely more efficient than J-PAKE.

IV. Conclusion

In our paper, we proposed two new variants of J-PAKE, showed that the security proof from [3] can be adapted to cover our proposals, and compared the overall efficiency of all three protocols when instantiated with ECs or SFFs. Since RO-J-PAKE using SFFs is the least efficient because of the implementation of the hash function H_0, it would be interesting to see if it can be proven secure using a large SFF (and therefore, a "small r"), all while using a short-exponent-type complexity assumption (e.g. as in [5]).

References

[1] F. Hao and P. Ryan, "J-PAKE: Authenticated Key Exchange without PKI," *Transactions on Computational Science*, vol. 11, pp. 192–206, 2010.

[2] C. Schnorr, "Efficient Identification and Signatures for Smart Cards," in *Advances in Cryptology - CRYPTO 1989*, ser. LNCS, G. Brassard, Ed., vol. 435. Springer, 1989, pp. 239–252.

[3] M. Abdalla, F. Benhamouda, and P. MacKenzie, "Security of the J-PAKE Password-Authenticated Key Exchange Protocol," in *2015 IEEE Symposium on Security and Privacy, SP 2015*. IEEE Computer Society, 2015, pp. 571–587.

[4] E. Brier, J. Coron, T. Icart, D. Madore, H. Randriam, and M. Tibouchi, "Efficient Indifferentiable Hashing into Ordinary Elliptic Curves," in *Advances in Cryptology - CRYPTO 2010*, ser. LNCS, T. Rabin, Ed., vol. 6223. Springer, 2010, pp. 237–254.

[5] P. D. MacKenzie and S. Patel, "Hard Bits of the Discrete Log with Applications to Password Authentication," in *Topics in Cryptology - CT-RSA 2005*, ser. LNCS, vol. 3376. Springer, 2005, pp. 209–226.

Universal Composition with Responsive Environments

Jan Camenisch*, Robert R. Enderlein*†, Stephan Krenn‡, Ralf Küsters§, Daniel Rausch§
*IBM Research – Zurich, Rüschlikon, Switzerland
Email: {jca,enr}@zurich.ibm.com
†Department of Computer Science, ETH Zürich, Zürich, Switzerland
‡AIT Austrian Institute of Technology GmbH, Vienna, Austria
Email: stephan.krenn@ait.ac.at
§University of Trier, Trier, Germany
Email: {kuesters,rauschd}@uni-trier.de

One of the most demanding tasks when designing a cryptographic protocol is to define its intended security guarantees, and to then prove that it indeed satisfies them. In the best case, these proofs should guarantee the security of the protocol in arbitrary contexts, i.e., also when composed with other, potentially insecure, protocols. This would allow one to split complex protocols into smaller components, which can then be separately analyzed one by one and once and for all, thus allowing for a modular security analysis. Over the last two decades, many models to achieve this goal have been proposed [1], [2], [3], [4], [5], [6], [7], [8], [9], [10], with Canetti's UC model being one of the first and most prominent ones.

All these models have in common that the designer first needs to specify an *ideal functionality* $\mathcal{F}$ defining the intended security and functional properties of the protocol. Informally, a *real protocol* realizes $\mathcal{F}$ if no efficient distinguisher (the *environment*) can decide whether it is interacting with the ideal functionality and a *simulator*, or with the real world protocol and an *adversary*.

Urgent requests/messages. In the specifications of such real protocols and ideal functionalities, it is often required for the adversary (and the environment) to provide to the protocol or the functionality some meta-information via the network interface, such as cryptographic algorithms, cryptographic values of signatures, ciphertexts, and keys, or corruption-related messages, and conversely, protocols/functionalities often have to provide the adversary with meta-information, for example, signaling information (e.g., the existence of machines) or again corruption-related messages. Such meta-information does not correspond to any real network messages, but is merely used for modeling purposes. Typically, giving the adversary/environment the option to not respond immediately to such modeling-specific messages does not translate to any real attack scenario. Hence, often it is natural for protocol designers to expect that the adversary/environment (answers and) returns control back to the protocol/functionality immediately when the adversary is requested to provide meta-information or when the adversary receives meta-information from the protocol/functionality. In the following, we call such messages from protocols/ideal functionalities on the network interface *urgent messages* or *urgent requests*.

Urgent requests occur in many functionalities and protocols from the literature, see, e.g., [1], [11], [12], [13], [14], [15], [16], [17], [18]. This is not surprising since the exchange of meta-information between the adversary/environment and the protocols/functionalities is an important mechanism for protocol designs in any UC-like model. In general, it seems impossible to dispense with urgent requests altogether, and certainly, such requests are very convenient and widely used in the literature.

The non-responsiveness problem. In the existing universal composability models it currently is not guaranteed that urgent requests are answered immediately by the adversary: when receiving an urgent request on the network interface, adversaries and environments can freely activate protocols and ideal functionalities in between, on network and I/O interfaces, without answering the request. In what follows, we refer to this problem as the *problem of non-responsive adversaries/environments* or the *non-responsiveness problem*.

This problem formally does not invalidate any of the UC-style models. It, however, often makes the specification of protocols and functionalities much harder and the models less expressive. Most disturbingly, as mentioned, the non-responsiveness problem is really an artificial problem: urgent requests do not correspond to any real messages and the adversary not responding promptly to such requests does not reflect any real attack scenario. Hence, non-responsiveness forces protocol designers to take care of artificial adversarial behavior that was unintended in the first place and is merely a modeling artifact.

In particular, protocol designers currently have to deal with various delicate problems: i) While waiting for a response to an urgent request, a protocol/ideal functionality might receive other requests, and hence, protocol designers have to take care of interleaving and dangling requests. ii)

While a protocol/ideal functionality is waiting for an answer from the adversary to an urgent request other parties and parts of the protocol/ideal functionality can be activated in the meantime (via the network or the I/O interface), which might change their state, even their corruption status, and which in turn might lead to race conditions.

This makes it difficult to deal with the non-responsiveness problem and results in unnecessarily complex and artificial specifications of protocols and ideal functionalities, which, in addition, are then hard to re-use. In some cases, one might not even be able to express certain desired properties. In current models, there is no generic and generally applicable way to deal with the non-responsiveness problem, and hence, one has to resort to solutions specifically tailored to the protocols at hand.

Looking at the literature, urgent requests and the non-responsiveness problem occur in many protocols and functionalities. Nevertheless, protocol designers frequently ignore this problem (see, e.g., [11], [12], [13], [14], [16], [18]), i.e., they seem to implicitly assume that urgent request *are* answered immediately, probably, at least as far as ideal functionalities are concerned, because their simulators promptly respond to these kinds of requests. As a result, protocols and ideal functionalities are underspecified and/or expose unexpected behavior, and thus, are not usable in other (hybrid) protocols or security proofs of hybrid protocols are flawed.

Our contribution. In this paper, we propose a universal composability framework with the new concept of *responsive environments* and *adversaries*, which should be applicable to all existing UC-style models (see below). This framework completely avoids and, by this, solves the non-responsiveness problem as it guarantees that urgent requests *are* answered immediately. This really is the most obvious and most natural solution to the problem: there is no reason that protocol designers should have to take care of the non-responsiveness problem and its many negative consequences.

More specifically, the main idea behind our framework is as follows. When a protocol/ideal functionality sends what we call a *restricting* message to the adversary/environment on the network interface, then the adversary/environment is forced to be responsive, i.e., reply with a valid response before sending any other message to the protocol. This requires careful definitions and non-trivial proofs in order to make sure that all properties and features that are expected in models for universal composition are lifted to the setting with responsive environments and adversaries.

By using our framework and concepts, protocols and ideal functionalities can be modeled in a very natural way: protocol designers can simply declare urgent requests to be restricting messages, which hence, have to be answered immediately. Furthermore, with our concepts we can easily fix existing specifications from the literature where the non-responsiveness problem has not properly been dealt with or has simply been ignored as protocol designers often implicitly assumed responsiveness for urgent messages. In some cases, we can now even express certain functionalities in a natural and elegant way which could not be expressed before. Of course, with simplified and more natural functionalities and protocols, security proofs become easier as well because the protocol designer does not have to consider irrelevant and unrealistic adversarial behavior and execution orders.

Our framework and concepts apply to existing models for universal composability. This is exemplified for three prominent models: UC [1], GNUC [4], and IITM [3], [5]. For the IITM model we provide detailed definitions and full proofs.

References

[1] R. Canetti, "Universally composable security: A new paradigm for cryptographic protocols," 2001, pp. 136–145, see also https://eprint.iacr.org/2000/067.pdf for full and previous versions.

[2] B. Pfitzmann and M. Waidner, "Composition and integrity preservation of secure reactive systems," 2000, pp. 245–254.

[3] R. Küsters, "Simulation-Based Security with Inexhaustible Interactive Turing Machines," in *CSFW '06*. IEEE, 2006, pp. 309–320.

[4] D. Hofheinz and V. Shoup, "GNUC: A new universal composability framework," Cryptology ePrint Archive, Report 2011/303, 2011.

[5] R. Küsters and M. Tuengerthal, "The IITM model: a simple and expressive model for universal composability," Cryptology ePrint Archive, Report 2013/025, 2013.

[6] U. Maurer, "Constructive Cryptography - A New Paradigm for Security Definitions and Proofs," in *TOSCA 2011*, vol. 6993, 2011, pp. 33–56.

[7] U. Maurer and R. Renner, "Abstract cryptography," 2011, pp. 1–21.

[8] R. Canetti, Y. Dodis, R. Pass, and S. Walfish, "Universally composable security with global setup," 2007, pp. 61–85.

[9] M. Backes, B. Pfitzmann, and M. Waidner, "The reactive simulatability (RSIM) framework for asynchronous systems," *Information and Computation*, vol. 205, no. 12, pp. 1685–1720, 2007.

[10] R. Canetti, L. Cheung, D. K. Kaynar, M. Liskov, N. A. Lynch, O. Pereira, and R. Segala, "Time-Bounded Task-PIOAs: A Framework for Analyzing Security Protocols," in *DISC 2006*, ser. LNCS, vol. 4167. Springer, 2006, pp. 238–253.

[11] S. Zhao, Q. Zhang, Y. Qin, and D. Feng, "Universally composable secure TNC protocol based on IF-T binding to TLS," in *NSS 2014*, vol. 8792, 2014, pp. 110–123.

[12] R. Canetti, D. Shahaf, and M. Vald, "Universally composable authentication and key-exchange with global PKI," Cryptology ePrint Archive, Report 2014/432, 2014.

[13] J. Camenisch, M. Dubovitskaya, K. Haralambiev, and M. Kohlweiss, "Composable & modular anonymous credentials: Definitions and practical constructions," *ASIACRYPT 2015*, 2015.

[14] M. Abe and M. Ohkubo, "A framework for universally composable non-committing blind signatures," 2009, pp. 435–450.

[15] R. Dowsley, J. Müller-Quade, A. Otsuka, G. Hanaoka, H. Imai, and A. C. A. Nascimento, "Universally composable and statistically secure verifiable secret sharing scheme based on pre-distributed data," *IEICE Transactions*, vol. 94-A, no. 2, pp. 725–734, 2011.

[16] T. Matsuo and S. Matsuo, "On universal composable security of time-stamping protocols," in *IWAP 2005*, 2005, pp. 169–181.

[17] R. Canetti, S. Halevi, and J. Katz, "Adaptively-secure, non-interactive public-key encryption," 2005, pp. 150–168.

[18] M. Backes and D. Hofheinz, "How to break and repair a universally composable signature functionality," 2004, pp. 61–72.

A class of precomputation-based distance-bounding protocols

Sjouke Mauw
CSC/SnT, University of Luxembourg
sjouke.mauw@uni.lu

Jorge Toro-Pozo
CSC, University of Luxembourg
jorge.toro@uni.lu

Rolando Trujillo-Rasua
SnT, University of Luxembourg
rolando.trujillo@uni.lu

***Abstract*—Distance-bounding protocols serve to thwart various types of proximity-based attacks, such as relay attacks. A particular class of distance-bounding protocols measures round trip times of a series of one-bit challenge-response cycles, during which the proving party must have minimal computational overhead. This can be achieved by precomputing the responses to the various possible challenges. We formalize this class of precomputation-based distance-bounding protocols. By designing an abstract model for these protocols, we can study their generic properties, such as security lower bounds in relation to space complexity. Further, we present a novel family of protocols in this class that resists well to mafia fraud attacks.**

I. Introduction

Contactless technologies such as RFID, have become the *de facto* solution for many identification/authentication applications, e.g. access control, ticketing, e-passports. Some access control mechanisms have been designed in such a way that physical proximity is enforced easily, e.g., mechanical locks or biometric identification. However, due to the open nature of wireless channels, providing the same kind of guarantee in wireless systems is far from trivial.

Simple proximity enforcing techniques, such as setting up small communication timeouts or short-range communication channels, can be easily circumvented in practice by a variety of attacks [1]. Perhaps, the most popular and devastating of such attacks is *mafia fraud* [2], also known as *relay attack* [3].

The most reliable countermeasure against these type of attacks is *distance-bounding* (DB) which typically consists in measuring the Round Trip Time (RTT) of a message exchange. Amongst more than 30 DB protocols proposed[1] so far, we can find a large class (e.g., [4], [5], [6], [7], [8]) whose members follow two core principles raised by Hancke and Kuhn in [3]:

- RTT measurements should exchange single-bit messages.
- Each RTT measurement ought to be based on a challenge-response authentication scheme so that, even if the protocol stops after a few RTT measurements, some guarantees of proximity can be provided.

Distance-bounding protocols adhering to these principles normally consist of two phases. The first phase is called the *slow phase*, where the verifier and the prover exchange nonces and use a shared key to secretly *precompute* a lookup table with potential responses for the next phase. The second phase, known as *fast phase*, consists of n RTT measurements (often called *rounds*). At the ith round, the verifier sends a random bit-challenge c_i to the prover and starts a clock. The prover replies instantly to the challenge c_i by using the precomputed lookup table. Upon reception of the prover's reply, the verifier stops the clock and computes the RTT. The protocol finishes correctly if all responses are correct and all RTTs are lower than a predefined threshold.

Because all protocols based on these principles have a similar shape, it would be natural and useful to consider them as instantiations of the same protocol scheme, with slight variations. That would provide us with a mathematical model, allowing us to study theoretical properties that hold for a large class of protocols. We have called them *precomputation-based distance-bounding protocols*.

II. Precomputation-based distance-bounding protocols

In [3] the authors explain the advantage of avoiding a final slow phase, even at the cost of an apparent decrease in the resistance to mafia fraud. In terms of execution time and computational complexity, the cost of executing a couple of additional rounds during the fast phase is significantly lower than the cost of performing expensive cryptographic operations and message exchanges over a traditional communication channel.

Precomputation-based protocols are distance-bounding protocols without a final slow phase where all possible responses to the verifier's challenges are precomputed in the initial slow phase. These responses are stored in memory such that query time should be minimum. The authentication is carried on during the fast phase. Because of the time measurement, the operations have to be as low cost as possible, e.g., accessing to random access memory, simple bit operations.

We sketch a simple model that captures a prominent class of DB protocols based on precomputation. The approach uses a particular class of Deterministic Finite Automata (DFA) with labels attached to states. We consider a protocol as a set of DFA, where each automaton describes the protocol's behaviour in the fast phase. The structure and labeling of such automaton follows from the calculations in the slow phase, in which, e.g., the nonces are chosen. Consequently, every possible outcome of the slow phase results in an automaton.

The execution of a protocol therefore consists of the (random) selection of one automaton (in the slow phase) and

[1] http://www.avoine.net/rfid/index.php

a run (fast phase) consisting of a n-steps walk through the automaton, i.e. an alternation of input and output symbols, that represent challenges and responses, respectively. We remind that n stands for the number of rounds.

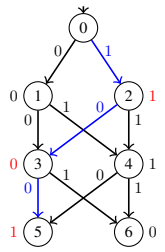

Fig. 1. An automaton example

In Figure 1 we depict an automaton example for the HK protocol [3] for 3 rounds. The states $\{0, 1, \ldots, 6\}$ are linked by transitions represented by the arrows. The states have binary labels attached, shown to their left and right. We have shown also an execution with the challenges 100 –in blue– whose corresponding responses –in red– are 101.

The proposed model captures several state-of-the-art DB protocols, such as [3], [4], [5], [8]. The virtue of this model is that it supports generic analysis of members of this protocol class. For instance, we can analyze the security limits of a protocol in relation to the number of rounds.

III. Properties and security analysis

In 2009, Avoine and Tchamakerten proposed the tree-based protocol [5], whose security against mafia fraud is $\frac{1}{2^n}(1+\frac{n}{2})$, where n is the number time measurements. Since then, that value has become a *de facto* lower bound on the resistance to this type of attacks.

Supported in our model, we have proved that, for any precomputation-based distance-bounding protocol, there exists an attack that succeeds with such a probability. This demonstrates that $\frac{1}{2^n}(1+\frac{n}{2})$ is a tight lower bound on the security of this type of protocols against mafia fraud. Based on this result, we introduce the concept of *optimality* that stands for a precomputation-based distance-bounding protocol for which, there does not exist a mafia fraud attack whose probability of success is higher than the above-mentioned value.

In that sense, the model allows to analyze the relation memory-optimality given that the required memory is *measurable* according to the number of states of the largest possible automaton in the protocol. It is worth remarking tree-based approach is the only optimal protocol proposed so far. However, it requires an exponential amount of states and consequently the memory complexity becomes exponential.

We also describe a subclass within the precomputation-based distance-bounding protocols, whose members are *layered* and *random-labeled*. A brief description of these two properties is as follows:

- A protocol is layered if for every automaton, two input sequences of different length reach different states.
- A protocol is random-labeled if for a random automaton and a random state in it, the probability of being labeled with any input symbol is the same.

To evaluate the resistance of a DB protocol against mafia fraud usually two strategies are considered: *pre-ask* and *post-ask* [9], although the latter is not relevant in protocols without a final slow phase [9]. A pre-ask strategy can be summarized as follows: the adversary relays the first slow phase between the verifier and the prover. Then –before the verifier starts the fast phase– it executes the fast phase with the prover, retrieving some information on possible responses (often called *pre-ask session*). Afterward, the attacker performs the fast phase with the legitimate verifier.

In the context of layered and random-labeled protocols, we proved that the best pre-ask strategy for the adversary is simply to reply to the verifier with the responses obtained from the prover in the pre-ask session. We state that HK and tree-based protocols are both within this subclass whereas the Poulidor protocol [4] is not.

Finally, we propose a family of layered and random-labeled protocols, called *uniform protocols*, whose members have a security level arbitrarily close to the optimal value $\frac{1}{2^n}(1+\frac{n}{2})$. The use of memory of these protocols are considerable lower than the exponential one required by the tree-based approach.

IV. Future work

As future work, we will study security of precomputation-based distance-bounding protocols against other types of attack, e.g. *distance fraud* and *terrorist fraud*. We will also analyze deeply the relation optimality-memory. Our hypothesis is that optimality implies exponential amount of states. Further, for a given upper-bound on the number of states (i.e. available memory), we want to find the best precomputation-based distance-bounding protocol. In addition, we will study the security in non-layered precomputation-based distance-bounding protocols, e.g. Poulidor protocol [4].

References

[1] G. P. Hancke, K. Mayes, and K. Markantonakis, "Confidence in smart token proximity: Relay attacks revisited," *Computers & Security*, vol. 28, no. 7, pp. 615–627, 2009.

[2] Y. Desmedt, C. Goutier, and S. Bengio, "Special uses and abuses of the Fiat-Shamir passport protocol," in *Proc. Advances in Cryptology (CRYPTO'87)*, ser. LNCS, vol. 293. Springer, 1988, pp. 21–39.

[3] G. Hancke and M. Kuhn, "An RFID distance bounding protocol," in *Proc. First International Conference on Security and Privacy for Emerging Areas in Communications Networks (SecureComm'05)*. IEEE, 2005, pp. 67–73.

[4] R. Trujillo-Rasua, B. Martin, and G. Avoine, "The Poulidor distance-bounding protocol," in *Proc. 6th International Conference on Radio Frequency Identification: Security and Privacy Issues (RFIDSec'10)*, ser. LNCS, vol. 6370. Springer, 2010, pp. 239–257.

[5] G. Avoine and A. Tchamkerten, "An efficient distance bounding RFID authentication protocol: Balancing false-acceptance rate and memory requirement," in *Proc. 12th International Conference on Information Security (ISC'09)*, ser. LNCS, vol. 5735. Springer, 2009, pp. 250–261.

[6] R. Trujillo-Rasua, B. Martin, and G. Avoine, "Distance bounding facing both mafia and distance frauds," *IEEE Transactions on Wireless Communications*, vol. 13, no. 10, pp. 5690–5698, 2014.

[7] C. Kim and G. Avoine, "RFID distance bounding protocols with mixed challenges," *IEEE Transactions on Wireless Communications*, vol. 10, no. 5, pp. 1618–1626, 2011.

[8] S. Kardas, M. S. Kiraz, M. A. Bingöl, and H. Demirci, "A novel RFID distance bounding protocol based on physically unclonable functions," in *Proc. 7th International Conference on Radio Frequency Identification: Security and Privacy Issues (RFIDSec'11)*, ser. LNCS, vol. 7055. Springer, 2012, pp. 78–93.

[9] G. Avoine, M. A. Bingöl, S. Kardaş, C. Lauradoux, and B. Martin, "A framework for analyzing RFID distance bounding protocols," *J. Comput. Secur.*, vol. 19, no. 2, pp. 289–317, Apr. 2011.

An Empirical Study on User Access Control in Online Social Networks

Minyue Ni
Software School, Fudan University
myni14@fudan.edu.cn

Yang Zhang
FSTC, University of Luxembourg
yang.zhang@uni.lu

Weili Han
Software School, Fudan University
wlhan@fudan.edu.cn

Jun Pang
FSTC&SnT, University of Luxembourg
jun.pang@uni.lu

ABSTRACT

This paper presents the first large-scale empirical study on users' access control usage on Twitter and Instagram. Based on the data of 150k users on Twitter and 280k users on Instagram collected consecutively during three months in New York, we have conducted both static and dynamic analysis on users' access control usage. Our findings include: female users and young users are more concerned about their privacy; users who enable their access control setting are less active and have smaller online social circles; global events and important festivals can influence users to change their access control settings. Furthermore, we exploit machine learning classifiers to perform an access control setting prediction. Through experiments, the predictor achieves a fair performance with the AUC equals to 0.70, indicating whether a user enables his access control setting or not can be predicted to a certain extent.

1. INTRODUCTION

Online social networks (OSNs) have attracted a huge number of users during the past decade, nowadays, they have become a primary way for people to connect, communicate and share life moments. Although OSNs have brought a lot of convenience to our life, users' privacy, on the other hand, has become a major concern due to the large amount of personal data shared online. To mitigate users' privacy concern, major OSNs have deployed access control schemes to delegate the power to users themselves to control who can view their private information. To further improve access control in OSNs, academia have conducted many research, most of which take either formal or logical approaches [2, 4, 1]. On the other hand, understanding how users exploit access control in their daily life is essential to improve access control in OSNs. Much to our surprise, this is left largely unexplored.

In this paper, we perform a large-scale empirical study on access control usage of Twitter and Instagram users in New York. To the best of our knowledge, this is the first work on analyzing users' access control on Twitter and Instagram. We collect the data of 150k Twitter users and 280k Instagram users continuously within three months and study their access control usage from both static and dynamic point of view. Especially, the dynamic analysis is conducted on a daily base, instead of a yearly base as done in the previous works [3, 5]. This allows us to understand in depth how users exploit access control in their daily OSN life.

2. BACKGROUND AND DATASET

We exploit Twitter and Instagram's APIs to query New York users' access control settings on a daily base for nearly three months, from October 15th, 2015 until January 12th, 2016. In total, we collect the data of 150k Twitter users and 280k Instagram users.

Users' demographic information is an important aspect of our analysis. We exploit Face++[1] to recognize a user's profile photo and get his gender, race (Asian, White, African American) and age information.

3. STATIC ANALYSIS

Our static analysis focus the relation between users' access control settings and their demographics, online behaviors and offline behaviors. Experimental results show that:

- female users, young users and Asian users are more concerned about their privacy than others;
- private users publish more posts than public users, but have smaller online social circles (see Figure 1);
- in the offline world, private users are more socially active than public users represented by check-ins.

4. DYNAMIC ANALYSIS

We study dynamic usage of users' access control in OSNs and have observed the following.

- Many users change their access control settings from time to time. Instagram users change more often than Twitter users. More users change from public to private, showing that users become more concerned about their privacy day by day.

[1] http://www.faceplusplus.com/

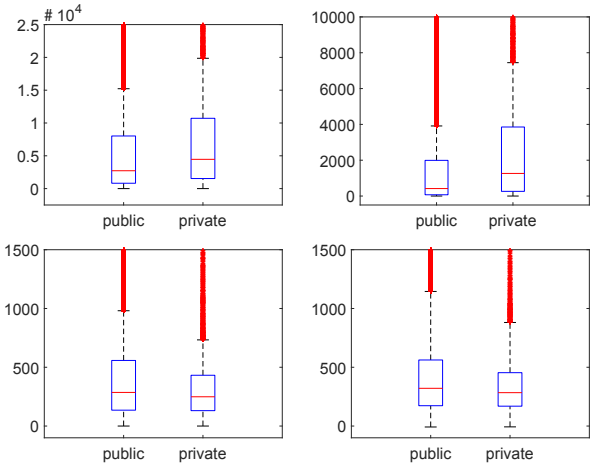

Figure 1: Users' distributions of posted tweets (top-left), favored tweets (top-right), followers (bottom-left) and followees (bottom-right).

- Female users and young users change their access control settings more frequently and their changing trend from public to private is faster than others. Asian and African American users behaves differently on Twitter and Instagram, while Wither users' changing behaviors are the least active in both OSNs.

- Constantly-private users are less active than constantly-public users in terms of published posts and new followers/followees. When users change from public to private, they publish less tweets than users changing from private to public, and delete their followers, their posts' topics are more privacy sensitive than before.

- Global events and festivals cause more users to change their access control setting from private to public (see Figure 2).

5. ACCESS CONTROL PREDICTION

We further investigate whether it is possible to predict a user's access control setting. Being able to predict a user's access control setting opens up opportunities for appealing applications. For instance, OSNs can automatically assign access control setting to their users for better privacy protection; government can develop a privacy advisor to remind users of their privacy leaks.

We model access control prediction as a binary classification problem, and intend to solve the problem with machine learning classifiers. We label private users as positive cases while public users as negative cases. For features used in classification, we consider users' number of follower/followees, number of tweets and number of favorites together with their demographics.

Our best classifier achieves a fair prediction (AUC=0.70), this indicates that a user's access control setting can be predicted to a certain extent.

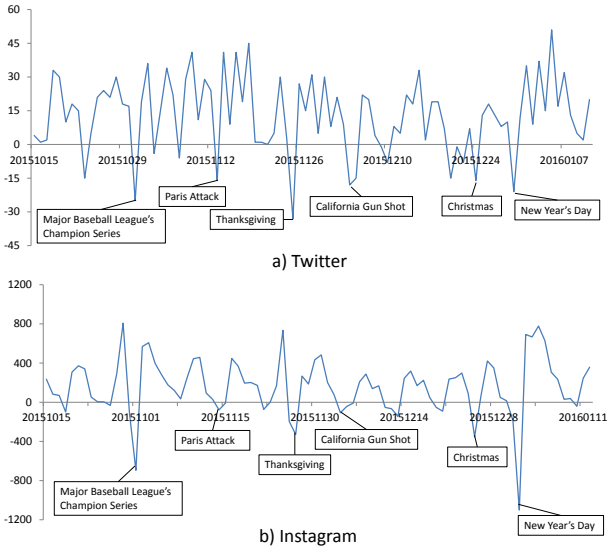

Figure 2: Differences between daily new private users and daily new public users.

References

[1] G. Bruns, P. W. L. Fong, I. Siahaan, and M. Huth. Relationship-based access control: its expression and enforcement through hybrid logic. In *Proc. 2nd ACM Conference on Data and Application Security and Privacy (CODASPY)*, pages 117–124. ACM, 2012.

[2] B. Carminati, E. Ferrari, R. Heatherly, M. Kantarcioglu, and B. Thuraisingham. A semantic web based framework for social network access control. In *Proc. 14th ACM Symposium on Access Control Models and Technologies (SACMAT)*, pages 177–186. ACM, 2009.

[3] R. Dey, Z. Jelveh, and K. Ross. Facebook users have become much more private: A large-scale study. In *Proc. 2012 IEEE International Conference on Pervasive Computing and Communications Workshops*, pages 346–352. IEEE, 2012.

[4] P. W. L. Fong. Preventing sybil attacks by privilege attenuation: a design principle for social network systems. In *Proc. 32nd IEEE Symposium on Security and Privacy (S&P)*, pages 263–278. IEEE CS, 2011.

[5] F. Stutzman, R. Gross, and A. Acquisti. Silent listeners: The evolution of privacy and disclosure on Facebook. *Journal of Privacy and Confidentiality*, 4(2):2, 2013.

Deconstructing MinBFT for Security and Verifiability

Vincent Rahli, Francisco Rocha, Marcus Völp and Paulo Esteves-Verissimo
CritiX Lab - Critical and Extreme Security and Dependability
SnT - Interdisciplinary Centre for Security, Reliability and Trust
University of Luxembourg
Email: ⟨name⟩.⟨surname⟩@uni.lu

***Abstract*—MinBFT is an efficient asynchronous Byzantine fault-tolerant state machine replication protocol and an element of the standard toolbox for hardening critical systems and infrastructures against faults and intrusions. The presence of a trusted component—the Unique Sequential Identifier Generator (USIG)—simplifies the normal case operation of MinBFT, which requires fewer communication steps than its predecessors, including PBFT. However, vulnerabilities in USIG itself might jeopardize these objectives. In this extended abstract, we elaborate on our work in progress in deconstructing MinBFT to reduce its trusted computing base and verify its critical components using the Verified Software Toolchain (VST) framework.**

I. Introduction

Our modern society came to rely so much on computer systems that vulnerabilities in critical systems and infrastructures can be disastrous. Practically all such systems and infrastructures rely on distributed systems technology, known to be quite difficult to implement, test, verify, and maintain. These characteristics make them prone to faults and attacks by increasingly well-educated adversaries. In general, these faults and attacks may result in components that can produce arbitrary, possibly malicious, outputs.

One standard technique for tolerating Byzantine behavior is to use state machine replication protocols such as PBFT [6, 7], BFT-SMaRt [5], MinBFT [16], CheapBFT [9], and COP [3]. Since these protocols are themselves distributed programs, checking all their corner cases by hand is error prone. Other approaches such as testing, static analysis, and model checking, although undeniably useful, are likely to suffer from severe limitations, such as state space explosion. Therefore, in this work we use proof assistants.

Among the above mentioned BFT systems, MinBFT [16] achieves high efficiency through the use of a trusted component called USIG, that has to be tamperproof. The solution adopted in CheapBFT [9] is to implement USIGs on FPGAs. However, many contemporary server systems lack support for reconfigurable hardware and the software stack required to upload and interface with this hardware is not trivial and a further source of errors. In this extended abstract, we describe our ongoing work to verify a C implementation of USIG, using the Coq proof assistant [8] and the Verified Software Toolchain (VST) [2, 1]. After verifying the sourcecode, one can generate verified machine code using the CompCert verified C compiler [12, 13], which is a compiler from Clight (a large subset of the C programming language) to assembly code. Such trustworthy components may then be run on a formally proven microhypervisor, such as seL4 [10] or XMHF [15].

This work is partially supported by the University of Luxembourg - SnT and by the Fonds National de la Recherche Luxembourg (FNR) through PEARL grant FNR/P14/8149128.

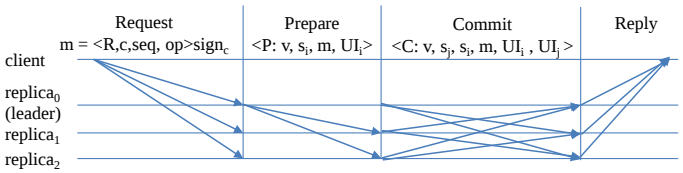

Fig. 1. Normal case operation.

Two of our overarching goals in this project are (1) to formally verify MinBFT and (2) to reduce the trusted computing base an attacker can explore to compromise the system by decomposing the MinBFT protocol into small, independent and trustworthy components such as the USIG. We can then leverage the execution environment offered by microkernels to execute these components in isolation from each other. This clearly reduces the attack surface of these components and permits the quick rejuvenation of compromised components when these are stateless.

We chose to base our approach on microhypervisors because they are more secure than other alternatives such as monolithic kernels. Another advantage is that they support legacy operating-systems and virtual machines (e.g., to execute the replicas). They also provide a more application-oriented execution environment for directly executing the compiled C code of our protocol components. Our approach extends to hypervisor-only setups, where we have to rely on Unikernels [14] to construct this execution environment.

Sec. II briefly introduces MinBFT. Sec. III describes its decomposition. Sec. IV describes the implementation of USIG on top of L4 Fiasco.OC [11]. Sec. V summarizes our planned verification effort. We conclude in Sec. VI.

II. MinBFT in a Nutshell

Fig. 1 shows the normal case operation of MinBFT. After receiving a signed request from the client, the leader selects a pending request, attaches a USIG identifier to the client message and forwards the request to the remaining replicas.

The USIG thereby ensures uniqueness, monotonicity and sequentiality of requests by never assigning the same identifier to different messages and by assigning subsequent values of an

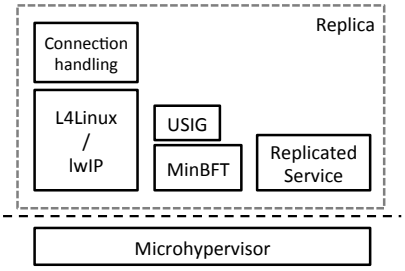

Fig. 2. Decomposition of MinBFT components

internal counter. Therefore, USIGs prevent equivocation, i.e., leaders cannot propose different messages to different replicas.

The multicast of commit messages ensures that a replica only accepts a client request when it receives a quorum of $f+1$ valid commit messages from different replicas for the proposed request. Finally, each replica replies to a client, which waits until it receives $f+1$ such replies, where f is the number of tolerable faults/attacks.

Although the USIG component prevents a faulty leader from performing replay attacks or assign random sequence values to messages, it is not capable of preventing a faulty leader from denying service by not assigning sequence numbers. In these scenarios, MinBFT executes a view change protocol which, briefly, involves correct replicas suspecting the faulty leader and reaching consensus in the election of a new leader.

III. Decomposing MinBFT

Our decision to decompose MinBFT is motivated by the observation that certain components (such as the network stack, the communication handling part and the virtual machine monitor (VMM)) are stateless and can be easily restarted, providing that the MinBFT instance feeds the replica with its desired state. This way, the attack surface on MinBFT is reduced as it essentially operates only on in-memory messages or on the replica checkpoints during recovery. The separation of USIG further improves the reliability of this component.

IV. Implementing USIG

We implemented USIG to read-share the MinBFT message memory as a 1:1 mapping to prevent message size restrictions and to avoid unnecessary copies, as well as to defend against page-faults in the MinBFT memory region (see Fig. 3). Upon receiving a pointer p and size s to a contiguously stored message, USIG computes the HMAC of the message using: (1) its internally stored secret key; (2) its current UI counter value; and (3) the message in the region $[p, p+s)$. Because of the 1:1 mapping, USIG can safely de-reference all message words after validating that $[p, p+s)$ is in the message memory region. After successfully reading the message, USIG increments the UI counter value and returns the hash.

To prevent tampering with USIG, we use L4's page-fault reflection feature to abort HMAC computation in case the computation page faults in the message region.

V. Verification

We are using the VST framework to verify the correctness of MinBFT's USIG component. VST is a separation logic-based verification framework for C, implemented in Coq. It

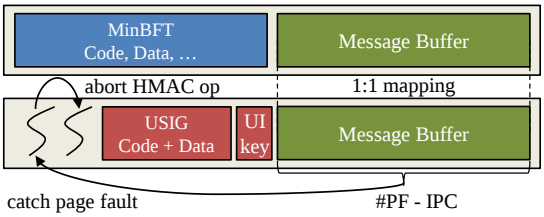

Fig. 3. USIG implementation

allows one to verify that C programs satisfy post-conditions assuming some pre-conditions. Because it is based on separation logic, these pre- and post-conditions can mention shared and mutable data structures. In order to verify the correctness of USIG, we are taking advantage of the fact that an OpenSSL implementation of HMAC with SHA-256 has recently been verified using VST [4]. Based on this result, the main result we will be proving is that the USIG generates certificates that satisfy the HMAC specification (in addition to the uniqueness, sequentiality and monoticity properties mentioned in Sec. II).

VI. Conclusions and Future Work

The original idea behind MinBFT was to derive an efficient BFT algorithm by isolating some critical functionality, here performed by the trustworthy USIG component mentioned above. Along those lines, we want to decompose the MinBFT even further into isolated components that would be easy to verify, and would be resilient because of their reduced trusted computing base.

References

[1] Andrew W. Appel. *Program Logics - for Certified Compilers*. Cambridge University Press, 2014.
[2] Andrew W. Appel. "Verified Software Toolchain - (Invited Talk)". In: *ESOP 2011*. Vol. 6602. LNCS. Springer, 2011, pp. 1–17.
[3] Johannes Behl, Tobias Distler, and Rüdiger Kapitza. "Consensus-Oriented Parallelization: How to Earn Your First Million". In: *16th Annual Middleware Conference*. ACM, 2015, pp. 173–184.
[4] Lennart Beringer et al. "Verified Correctness and Security of OpenSSL HMAC". In: *USENIX Security 15*. USENIX Association, 2015, pp. 207–221.
[5] Alysson Neves Bessani, João Sousa, and Eduardo Adílio Pelinson Alchieri. "State Machine Replication for the Masses with BFT-SMART". In: *DSN 2014*. IEEE, 2014, pp. 355–362.
[6] Miguel Castro and Barbara Liskov. "Practical Byzantine Fault Tolerance". In: *OSDI 1999*. USENIX Association, 1999, pp. 173–186.
[7] Miguel Castro and Barbara Liskov. "Practical byzantine fault tolerance and proactive recovery". In: *ACM Trans. Comput. Syst.* 20.4 (2002), pp. 398–461.
[8] *The Coq Proof Assistant*. URL: http://coq.inria.fr/.
[9] Rüdiger Kapitza et al. "CheapBFT: resource-efficient byzantine fault tolerance". In: *EuroSys '12*. ACM, 2012, pp. 295–308.
[10] Gerwin Klein et al. "seL4: Formal Verification of an OS Kernel". In: *SOSP 2009*. ACM, 2009, pp. 207–220.
[11] *Fiasco*. URL: https://os.inf.tu-dresden.de/fiasco/.
[12] Xavier Leroy. "Formal certification of a compiler back-end or: programming a compiler with a proof assistant". In: *POPL 2006*. ACM, 2006, pp. 42–54.
[13] Xavier Leroy. "Formal verification of a realistic compiler". In: *Commun. ACM* 52.7 (2009), pp. 107–115.
[14] Anil Madhavapeddy et al. "Unikernels: library operating systems for the cloud". In: *ASPLOS '13*. ACM, 2013, pp. 461–472.
[15] Amit Vasudevan et al. "Design, Implementation and Verification of an eXtensible and Modular Hypervisor Framework". In: *SP 2013*. IEEE Computer Society, 2013, pp. 430–444.
[16] Giuliana Santos Veronese et al. "Efficient Byzantine Fault-Tolerance". In: *IEEE Trans. Computers* 62.1 (2013), pp. 16–30.

AMPPOT: Monitoring and Defending Against Amplification DDoS Attacks

Lukas Krämer, Johannes Krupp, and Christian Rossow

CISPA, Saarland University, Germany

Abstract

The recent amplification DDoS attacks have swamped victims with huge loads of undesired traffic, sometimes even exceeding hundreds of Gbps attack bandwidth. We analyze these amplification attacks in more detail. First, we inspect the reconnaissance step, i.e., how both researchers and attackers scan for amplifiers that are open for abuse. Second, we design AMPPOT, a novel honeypot that tracks amplification attacks. We deploy 12 honeypots to reveal previously-undocumented insights about the attacks. We find that the vast majority of attacks are short-lived and most victims are attacked only once. Furthermore, 96% of the attacks stem from single sources, which is also confirmed by our detailed analysis of four popular Linux-based DDoS botnets.

1 Introduction

Distributed denial-of-service (DDoS) attacks have threatened critical Internet infrastructures for many years [1–3]. Recently, in particular amplification DDoS attacks [4] have gained increasing popularity. In such amplification attacks, an attacker abuses so called *amplifiers* (or *reflectors*) to exhaust the bandwidth of a victim. Instead of directing the attack traffic to the victim directly, the adversary sends requests to reflectors and spoofs the source IP address, so that the reflectors' responses are directed to the victim. An attacker may abuse any public server that is vulnerable to reflection attacks, such as open DNS resolvers or NTP servers. Worse, these protocols are known to amplify the bandwidth significantly, easily allowing an attacker to launch Gbps-scale attacks with a much smaller uplink. In fact, amplification attacks have caused the largest DDoS attack volume ever observed, e.g., against Spamhaus in 03/2013 ($\approx$ 300 Gpbs) and OVH in 02/2014 ($\approx$ 400 Gbps).

The rise of amplification attacks raises many research questions. How frequent are such attacks, and whom do they target? Are individual sources spoofing traffic to trigger attack traffic, or do distributed botnets cause the DDoS attacks? Which software do adversaries use to launch the attacks, and how do they identify amplifiers? Can network-based filtering methods be used to detect amplification attacks? All these questions help to improve our understanding of the threat, to learn attack motivations, and to devise effective countermeasures.

In this paper, we will close this gap by studying in-the-wild activities of attackers preparing and launching amplification DDoS attacks. We first leverage a `/16` IPv4 darknet to identify scans for amplifiers, revealing that over 5,000 hosts scanned for DDoS-related services. We observe the scans over time, and monitor a sudden increase of scans caused by whitehats in early 2014. Further analyses reveal that scans are widely distributed, and large parts of the scans rely on Zmap [5] for their reconnaissance.

We then perform a longitudinal study of amplification attacks. To this end, we introduce AMPPOT, a novel open-source honeypot specifically designed to monitor amplification attacks. AMPPOT can mimic services that are known to be vulnerable to amplification attacks, such as DNS and NTP. To make them attractive to attackers, our honeypots send back legitimate responses. Attackers, in turn, will abuse these honeypots as amplifiers, which allows us to observe ongoing attacks, their victims, and the DDoS techniques. To prevent damage caused by our honeypots, we limit the response rate. This way, while attackers can still find these rate-limited honeypots, the honeypots stop replying in the face of attacks.

We deployed 12 globally-distributed AMPPOT instances, which observed more than 3 million attacks in 2015. Analyzing the attacks more closely, we find that more than 96% of the attacks stem from single sources, such as booter services. We show that most attacks are relatively short-lived, and victims are rarely attacked multiple times—giving interesting insights into the motivation behind the attacks. We conclude that amplification

DDoS attacks are a global problem, with most victims being located in the US (32%) and China (14%).

To foster attack mitigation, we further devise reactive countermeasures against amplification attacks. First, we provide a live feed of amplification attacks. Second, we derive and present a list of domains that are abused in DNS-based amplification attacks. Finally, to study the root cause of amplification attacks, we analyze the new trend of Linux-based DDoS botnets. We inspect over 200 DDoS malware samples and classify most of them into four families. We manually reverse-engineer these samples to analyze their attack techniques, revealing amplification capabilities in all families. In an attempt to map attacks to DDoS botnets, we fingerprint the traffic of these families and link it to the attacks observed at the honeypots. This analysis reveals little overlap, showing that DDoS botnets are not the main source of amplification attacks.

To summarize, the contributions of this paper are as follows:

1. We design AMPPOT, a novel honeypot to capture amplification DDoS attacks. We evaluate various response modes and, based on our collected attacks, devise best practices for deploying such honeypots.

2. We leverage a `/16` darknet and the data collected by 12 AMPPOT instances to shed light on the current state of in-the-wild amplification attacks. We use these results to derive honeypot-assisted defense mechanisms.

3. We analyze the recent threat of Linux-based DDoS bots. We show that these bots offer amplification DDoS capabilities, but using traffic fingerprinting, we also reveal that their overall share in the amplification attacks is negligible.

Our full paper has been accepted at the *International Symposium on Research in Attacks, Intrusions and Defenses* (RAID) 2015, and is available at

`http://christian-rossow.de/publications/amppot-raid2015.pdf`

References

[1] Mirkovic, J., Reiher, P.: A Taxonomy of DDoS Attack and DDoS Defense Mechanisms. In: ACM SIGCOMM Computer Communication Review. Volume 34. (2004) 39–53

[2] Specht, S.M., Lee, R.B.: Distributed Denial of Service: Taxonomies of Attacks, Tools, and Countermeasures. In: Proceedings of the International Conference on Parallel and Distributed Computing (and Communications) Systems (ISCA PDCS), San Francisco, CA (2004)

[3] Paxson, V.: An Analysis of Using Reflectors for Distributed Denial-of-Service Attacks. In: Computer Communication Review. (2001)

[4] Rossow, C.: Amplification Hell: Revisiting Network Protocols for DDoS Abuse. In: Proceedings of the 2014 Network and Distributed System Security (NDSS) Symposium. (2014)

[5] Durumeric, Z., Wustrow, E., Halderman, J.A.: ZMap: Fast Internet-wide Scanning and Its Security Applications. In: Proceedings of the 22nd USENIX Security Symposium, Washington, D.C., USA (2013)

Micro-Policies for Web Session Security

Stefano Calzavara
Università Ca' Foscari Venezia
calzavara@dais.unive.it

Riccardo Focardi
Università Ca' Foscari Venezia
focardi@dais.unive.it

Niklas Grimm
Saarland University
grimm@cs.uni-saarland.de

Matteo Maffei
Saarland University
maffei@cs.uni-saarland.de

***Abstract*—Micro-policies, originally proposed to implement hardware-level security monitors, constitute a flexible and general enforcement technique, based on assigning security tags to system components and taking security actions based on dynamic checks over these tags. In this paper, we present the first application of micro-policies to web security, by proposing a core browser model supporting them and studying its effectiveness at securing web sessions. In our view, web session security requirements are expressed in terms of a simple, purely declarative information flow policy, which is then automatically translated into a micro-policy implementing it. This leads to a browser-side enforcement mechanism which is elegant, sound and flexible, while being accessible to web developers. We show how a large class of attacks against web sessions can be uniformly and effectively prevented by the adoption of this approach. Since we carefully designed micro-policies with ease of deployment in mind, we are also able to implement our proposal as a Google Chrome extension, Michrome: our experiments show that Michrome can be easily configured to enforce strong security policies without breaking the websites functionality.**

I. Introduction

The Web is nowadays the primary means of access to a plethora of online services with strict security requirements. Electronic health records and online statements of income are a well-established reality as of now, and more and more security-sensitive services are going to be supplied online in the next few years. Despite the critical importance of securing these online services, web applications and, more specifically, *web sessions* are notoriously hard to protect, since they can be attacked at many different layers.

At the network layer, man-in-the-middle attacks can break both the confidentiality and the integrity of web sessions running (at least partially) over HTTP. The standard solution against these attacks is deploying the entire web application over HTTPS with trusted certificates and, possibly, making use of HSTS [9] to prevent subtle attacks like SSL stripping. At the session implementation layer, code injection attacks (or again network attacks) can be exploited to steal authentication cookies and hijack a web session, or to compromise the integrity of the cookie jar and mount dangerous attacks like session fixation [11]. This is particularly problematic because, though the standard HttpOnly and Secure cookie attributes [2] are effective at protecting cookie confidentiality, no effective countermeasure exists as of now to ensure cookie integrity on the Web [16]. Finally, web sessions can also be attacked at the application layer: for instance, since browsers automatically attach cookies set by a website to all the requests sent to it, cross-site request forgery (CSRF) attacks can be mounted by a malicious web page to harm the integrity of the user session with a trusted web application and inject attacker-controlled messages inside it. Standard solutions against this problem include the usage of secret tokens and the validation of the Origin header attached by the browser to filter out malicious web requests [3].

In principle, it is possible to achieve a reasonable degree of security for web sessions using the current technologies, but the overall picture still exhibits several important shortcomings and it is far from being satisfactory. First, there are mechanisms like the HttpOnly cookie attribute which are easy to use, popular and effective, but lack flexibility: a cookie may either be HttpOnly or not, hence JavaScript may either be able to access it or be prevented from doing any kind of computation over the cookie value. There is no way, for instance, to let JavaScript access a cookie for legitimate computations, at the cost of disciplining its communication behaviour to prevent the cookie leakage. Then, there are defenses which are sub-optimal and not always easy to implement: this is the case for token-based protection against CSRF. Not only this approach must be directly implemented into the APIs of a web development framework to ensure that it is convenient to use, but also it is not very robust, since it fails in presence of code injection vulnerabilities which disclose the token value to the attacker. Finally, we observe that some attacks and attack vectors against web sessions are underestimated by existing standards and no effective solution against them can be deployed as of now: this is the case for many threats to cookie integrity [16]. These issues will likely be rectified with ad-hoc solutions in future standards, whenever browser vendors and web application developers become more concerned about their importance, and find a proper way to patch them while preserving the compatibility with existing websites.

In this paper, we advocate that a large class of attacks harming the security of web sessions can be provably, uniformly, and effectively prevented by the adoption of *browser-enforced security policies*, reminiscent of a dynamic typing discipline for the browser. In particular, we argue for the adoption of *micro-policies* [8] as a convenient tool to improve the security of web sessions, by disciplining the browser behaviour when interacting with security-sensitive web applications. Roughly, the specification of a micro-policy involves: (1) the definition of a set of *tags*, used to label selected elements of the web ecosystem, like URLs, cookies, network connections, etc., and (2) the definition of a *transfer* function, defining which operations are permitted by the browser based on the tags and how tags are assigned to browser elements after a successful operation. This kind of security policies has already proved

helpful for deploying hardware-level security monitors and nicely fits existing web security solutions, like cookie security attributes [2] and whitelist-based defenses in the spirit of the Content Security Policy [15].

Though previous work has already proposed browser-side security policies as a viable approach for protecting the Web [10], [12], [13], [14], [7], we are the first to carry out a foundational study on a possible extension of a web browser with support for micro-policies and discuss web session security as an important application for this framework. There are many different ways to deploy micro-policies in web browsers, but our proposal is driven by two main design goals aimed at simplifying a large-scale adoption. First, it is *light-weight* and it does not need to change existing web browsers too much, as testified by a prototype implementation of our approach as an extension for Google Chrome. Second, it is *practical* to use: although our technical development is based on a non-trivial theory, we strive for supporting declarative policies for web session security, which do not significantly deviate from the tools and the abstractions which web developers already appreciate and use today. We thus propose to express web session security requirements in terms of a simple, purely declarative information flow policy, which can be automatically translated into a micro-policy implementing it.

Our contributions can be summarized as follows:

1) we design FF^τ, a core model of a web browser extended with support for micro-policies. We define the operational behaviour of FF^τ using a small-step reactive semantics in the spirit of previous formal work on browser security [5], [4], [6]. The semantics of FF^τ is parametric with respect to an arbitrary set of tags and the definition of a transfer function operating on these tags;
2) we instantiate the set of tags of FF^τ to intuitive information flow labels and we characterize standard attackers from the web security literature in terms of these labels. We then discuss how to translate simple information flow policies for web session security into micro-policies which enforce them: this is crucial to ensure that most web developers can benefit from our proposal;
3) we discuss example applications of our theory by revisiting known attacks against web sessions and discussing limitations of existing solutions. We then show how these issues are naturally and more effectively solved by our enforcement technique;
4) we develop a prototype implementation of our proposal as a standard Google Chrome extension, Michrome, and we run a set of experiments testing its practicality. Our experiments show that Michrome can be easily configured to enforce strong security policies without breaking the functionality of existing websites.

Michrome and a technical report including full proofs are available online [1].

REFERENCES

[1] Anonymus, "Micro-policies for web session security," 2016, available at https://sites.google.com/site/micropolwebsese.

[2] A. Barth, "Http state management mechanism," 2011, available at https://tools.ietf.org/html/rfc6265.

[3] A. Barth, C. Jackson, and J. C. Mitchell, "Robust defenses for cross-site request forgery," in *Proceedings of the 2008 ACM Conference on Computer and Communications Security, CCS 2008, Alexandria, Virginia, USA, October 27-31, 2008*, 2008, pp. 75–88.

[4] A. Bohannon and B. C. Pierce, "Featherweight firefox: Formalizing the core of a web browser," in *USENIX Conference on Web Application Development, WebApps'10, Boston, Massachusetts, USA, June 23-24, 2010*, 2010.

[5] A. Bohannon, B. C. Pierce, V. Sjöberg, S. Weirich, and S. Zdancewic, "Reactive noninterference," in *Proceedings of the 2009 ACM Conference on Computer and Communications Security, CCS 2009, Chicago, Illinois, USA, November 9-13, 2009*, 2009, pp. 79–90.

[6] M. Bugliesi, S. Calzavara, R. Focardi, W. Khan, and M. Tempesta, "Provably sound browser-based enforcement of web session integrity," in *IEEE 27th Computer Security Foundations Symposium, CSF 2014, Vienna, Austria, 19-22 July, 2014*, 2014, pp. 366–380.

[7] A. Czeskis, A. Moshchuk, T. Kohno, and H. J. Wang, "Lightweight server support for browser-based CSRF protection," in *22nd International World Wide Web Conference, WWW '13, Rio de Janeiro, Brazil, May 13-17, 2013*, 2013, pp. 273–284.

[8] A. A. de Amorim, M. Dénès, N. Giannarakis, C. Hritcu, B. C. Pierce, A. Spector-Zabusky, and A. Tolmach, "Micro-policies: Formally verified, tag-based security monitors," in *2015 IEEE Symposium on Security and Privacy, SP 2015, San Jose, CA, USA, May 17-21, 2015*, 2015, pp. 813–830.

[9] J. Hodges, C. Jackson, and A. Barth, "Http strict transport security (hsts)," 2012, available at https://tools.ietf.org/html/rfc6797.

[10] T. Jim, N. Swamy, and M. Hicks, "Defeating script injection attacks with browser-enforced embedded policies," in *Proceedings of the 16th International Conference on World Wide Web, WWW 2007, Banff, Alberta, Canada, May 8-12, 2007*, 2007, pp. 601–610.

[11] M. Johns, B. Braun, M. Schrank, and J. Posegga, "Reliable protection against session fixation attacks," in *Proceedings of the 2011 ACM Symposium on Applied Computing (SAC), TaiChung, Taiwan, March 21 - 24, 2011*, 2011, pp. 1531–1537.

[12] M. T. Louw and V. N. Venkatakrishnan, "Blueprint: Robust prevention of cross-site scripting attacks for existing browsers," in *30th IEEE Symposium on Security and Privacy (S&P 2009), 17-20 May 2009, Oakland, California, USA*, 2009, pp. 331–346.

[13] S. Stamm, B. Sterne, and G. Markham, "Reining in the web with content security policy," in *Proceedings of the 19th International Conference on World Wide Web, WWW 2010, Raleigh, North Carolina, USA, April 26-30, 2010*, 2010, pp. 921–930.

[14] J. Weinberger, A. Barth, and D. Song, "Towards client-side HTML security policies," in *6th USENIX Workshop on Hot Topics in Security, HotSec'11, San Francisco, CA, USA, August 9, 2011*, 2011.

[15] M. West, A. Barth, and D. Veditz, "Content security policy (csp)," 2015, available at http://www.w3.org/TR/CSP/.

[16] X. Zheng, J. Jiang, J. Liang, H. Duan, S. Chen, T. Wan, and N. Weaver, "Cookies lack integrity: Real-world implications," in *24th USENIX Security Symposium, USENIX Security 15, Washington, D.C., USA, August 12-14, 2015.*, 2015, pp. 707–721.

A Comprehensive Formal Security Analysis of OAuth 2.0

Daniel Fett, Ralf Küsters, and Guido Schmitz
University of Trier, Germany
Email: {fett,kuesters,schmitzg}@uni-trier.de

The OAuth 2.0 authorization framework [7] defines a web-based protocol that allows a user to grant web sites access to her resources (data or services) at other web sites (*authorization*). The former web sites are called relying parties (RP) and the latter are called identity providers (IdP).[1] In practice, OAuth 2.0 is often used for *authentication* as well. That is, a user can log in at an RP using her identity managed by an IdP (single sign-on, SSO).

Authorization and SSO solutions have found widespread adoption in the web over the last years, with OAuth 2.0 being one of the most popular frameworks. OAuth 2.0, in the following often simply called *OAuth*,[2] is used by identity providers such as Facebook, Google, Microsoft, Yahoo, GitHub, and Dropbox. This enables billions of users to log in at millions of RPs or share their data with these [12], making OAuth one of the most used single sign-on systems on the web.

OAuth is also the foundation for the new single sign-on protocol OpenID Connect, which is already in use and actively supported by PayPal ("Log In with PayPal"), Google, and Microsoft, among others. Considering the broad industry support for OpenID Connect, a widespread adoption of OpenID Connect in the next years seems likely. OpenID Connect builds upon OAuth and provides clearly defined interfaces for user authentication and additional (optional) features, such as dynamic identity provider discovery and relying party registration, signing and encryption of messages, and user logout.

OAuth defines a complex protocol. The interactions between the user and her browser, the RP, and the IdP can be performed in four different flows, or grant types: authorization code grant, implicit grant, resource owner password credentials grant, and the client credentials grant (we refer to these as *modes* in the following). In addition, in most of these modes, depending on the configuration and prior setup of the RP and the IdP, further options within the different modes are provided.

Therefore, analyzing the security of OAuth is a complex task. So far, most analysis efforts were targeted towards finding errors in specific implementations [1], [2], [9], [11], [13], rather than the comprehensive analysis of the standard itself. Probably the most detailed formal analysis carried out on OAuth so far is the one in [1]. However, none of the existing analysis efforts of OAuth account for all modes of OAuth running simultaneously, which may potentially introduce new security risks. In fact, many existing approaches analyze only the authorization code mode and the implicit mode of OAuth. Also, importantly, there are no analysis efforts that are based on a comprehensive formal web model (see below), which, however, is essential to rule out security risks that arise when running the protocol in the context of common web technologies.

Contributions of this Paper. We perform the first extensive formal analysis of the OAuth 2.0 standard for all four modes, which can even run simultaneously within the same and different RPs and IdPs, based on a comprehensive web model which covers large parts of how browsers and servers interact in real-world setups. Our analysis also covers the case of malicious IdPs and RPs.

Formal model of OAuth. Our formal analysis of OAuth uses an expressive Dolev-Yao style model of the web infrastructure [3] proposed by Fett, Küsters, and Schmitz. This model has already been used to analyze the security of the BrowserID single sign-on system [3], [4] as well as the security and privacy of the SPRESSO single sign-on system [5]. This web model is designed independently of a specific web application and closely mimics published (de-facto) standards and specifications for the web, for instance, the HTTP/1.1 and HTML5 standards and associated (proposed) standards. It is the most comprehensive web model to date. Among others, HTTP(S) requests and responses, including several headers, such as cookie, location, strict transport security (STS), and origin headers, are modeled. The model of web browsers captures the concepts of windows, documents, and iframes, including the complex navigation rules, as well as new technologies, such as web storage and cross-document messaging (postMessages). JavaScript is modeled in an abstract way by so-called scripting processes which can be sent around and, among others, can create iframes and initiate XMLHTTPRequests (XHRs). Browsers may be corrupted dynamically by the adversary.

Using this generic web model, we build a formal model of OAuth, closely following the OAuth 2.0 standard [7]. Since this standard does not fix all aspects of the protocol, we use the current OAuth 2.0 security recommendations (RFC6819 [10]) and current web best practices (e.g., regarding sesssion handling) to obtain a model of OAuth 2.0 with state-of-the-art security

[1] Following the OAuth 2.0 terminology, IdPs are called *authorization servers* and *resource servers*, RPs are called *clients*, and users are called *resource owners*. Here, however, we stick to the more common terms mentioned above.

[2] Note that in this document, we consider only OAuth 2.0, which is very different to its predecessor, OAuth 1.0.

features in place, in order to avoid known implementation attacks. (Note that the security recommendations in RFC6819 cover many of the bugs found in earlier analysis efforts on implementations of OAuth.) As mentioned above, our model includes RPs and IdPs that (simultaneously) support all four modes and can be dynamically corrupted by the adversary. Also, we model all configuration options of OAuth.

Formalization of security properties. Based on this model of OAuth, we provide formal definitions of the security properties of OAuth. In particular, we state two separate properties: authorization and authentication.

New attacks on OAuth 2.0 and fixes. While trying to prove these properties, we discovered two previously unknown attacks on OAuth, which both break authorization as well as authentication. In the first attack, IdPs inadvertently forward user credentials (i.e., username and password) to the RP or the attacker. In the second attack, a network attacker can impersonate any victim. This severe attack is caused by a logical flaw in the OAuth 2.0 protocol and depends on the presence of malicious IdP. In practice, OAuth setups often allow for selected (and thus hopefully trustworthy) IdPs only. In these setups the attack would not apply. The attack, however, can be exploited in OpenID Connect, which, as mentioned, builds directly on OAuth. We have verified the attacks on an implementation of OAuth and OpenID Connect.

We also propose fixes for OAuth and OpenID Connect that are easy to implement in new and existing deployments.

We have notified the respective working groups, who confirmed the attacks and are currently discussing changes to the standards and recommendations based on our proposed fixes [8].

Formal analysis of OAuth 2.0. We then prove that OAuth satisfies the authorization and authentication properties in a model of OAuth with the fixes in place. This is the first proof which establishes the security of OAuth in a comprehensive and expressive web model.

We note that while these results provide strong security guarantees for OAuth they do not directly imply security of OpenID Connect because OpenID Connect adds specific details on top of OAuth. We leave a formal analysis of OpenID Connect to future work. The results obtained here can serve as a good foundation for such an analysis.

See [6] for the full version of this paper.

References

[1] C. Bansal, K. Bhargavan, A. Delignat-Lavaud, and S. Maffeis. Discovering Concrete Attacks on Website Authorization by Formal Analysis. *Journal of Computer Security*, 22(4):601–657, 2014.

[2] E. Y. Chen, Y. Pei, S. Chen, Y. Tian, R. Kotcher, and P. Tague. OAuth Demystified for Mobile Application Developers. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security - CCS '14*, pages 892–903, 2014.

[3] D. Fett, R. Küsters, and G. Schmitz. An Expressive Model for the Web Infrastructure: Definition and Application to the BrowserID SSO System. In *35th IEEE Symposium on Security and Privacy (S&P 2014)*, pages 673–688. IEEE Computer Society, 2014.

[4] D. Fett, R. Küsters, and G. Schmitz. Analyzing the BrowserID SSO System with Primary Identity Providers Using an Expressive Model of the Web. In *Computer Security - ESORICS 2015 - 20th European Symposium on Research in Computer Security, Vienna, Austria, September 21-25, 2015, Proceedings, Part I*, Lecture Notes in Computer Science, pages 43–65. Springer, 2015.

[5] D. Fett, R. Küsters, and G. Schmitz. SPRESSO: A Secure, Privacy-Respecting Single Sign-On System for the Web. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, Denver, CO, USA, October 12-6, 2015*, pages 1358–1369. ACM, 2015.

[6] D. Fett, R. Küsters, and G. Schmitz. A Comprehensive Formal Security Analysis of OAuth 2.0. Technical Report arXiv:1601.01229, arXiv, 2016. Available at http://arxiv.org/abs/1601.01229.

[7] D. Hardt (ed.). RFC6749 – The OAuth 2.0 Authorization Framework. IETF. Oct. 2012. https://tools.ietf.org/html/rfc6749.

[8] M. Jones and J. Bradley. OAuth 2.0 Mix-Up Mitigation – draft-jones-oauth-mix-up-mitigation-01. IETF. Jan. 2016. https://tools.ietf.org/html/draft-jones-oauth-mix-up-mitigation-01.

[9] W. Li and C. J. Mitchell. Security issues in OAuth 2.0 SSO implementations. In *Information Security - 17th International Conference, ISC 2014, Hong Kong, China, October 12-14, 2014. Proceedings*, pages 529–541, 2014.

[10] T. Lodderstedt (ed.), M. McGloin, and P. Hunt. RFC6819 – OAuth 2.0 Threat Model and Security Considerations. IETF. Jan. 2013. https://tools.ietf.org/html/rfc6819.

[11] M. Shehab and F. Mohsen. Towards Enhancing the Security of OAuth Implementations in Smart Phones. In *2014 IEEE International Conference on Mobile Services*. Institute of Electrical & Electronics Engineers (IEEE), jun 2014.

[12] SimilarTech. Facebook Connect Market Share and Web Usage Statistics. Last visited Nov. 7, 2015.

[13] S.-T. Sun and K. Beznosov. The Devil is in the (Implementation) Details: An Empirical Analysis of OAuth SSO Systems. In T. Yu, G. Danezis, and V. D. Gligor, editors, *ACM Conference on Computer and Communications Security, CCS'12*, pages 378–390. ACM, 2012.