GESELLSCHAFT
FÜR INFORMATIK

Delphine Reinhardt, Hanno Langweg,
Bernhard C. Witt, Mathias Fischer (Hrsg.)

SICHERHEIT 2020

Sicherheit, Schutz und Zuverlässigkeit

Konferenzband der 10. Jahrestagung des Fachbereichs Sicherheit der Gesellschaft für Informatik e.V. (GI)

17.–20. März 2020
in Göttingen

Gesellschaft für Informatik e.V. (GI)

Lecture Notes in Informatics (LNI) - Proceedings
Series of the Gesellschaft für Informatik (GI)

Volume P-301

ISBN 978-3-88579-695-4
ISSN 1617-5468

Volume Editors
Prof. Dr. Delphine Reinhardt
Universität Göttingen, Goldschmidtstr. 7, 37077 Göttingen, reinhardt@cs.uni-goettingen.de
Prof. Dr. Hanno Langweg
HTWG Konstanz, Alfred-Wachtel-Str. 8, 78462 Konstanz, hanno.langweg@htwg-konstanz.de
Bernhard C. Witt
it.sec GmbH, Einsteinstr. 55, 89077 Ulm, bcwitt@it-sec.de
Prof. Dr. Mathias Fischer
Universität Hamburg, Vogt-Kölln-Str. 30, 22527 Hamburg, mathias.fischer@uni-hamburg.de

printed by Köllen Druck+Verlag GmbH, Bonn

Vorwort

Die zehnte Ausgabe der Fachtagung Sicherheit, Schutz und Zuverlässigkeit (SICHERHEIT) fand im März 2020 in Göttingen statt. Die SICHERHEIT ist die regelmäßig stattfindende Tagung des Fachbereichs Sicherheit – Schutz und Zuverlässigkeit der Gesellschaft für Informatik e.V. (GI). Die Tagung deckt alle Aspekte der Sicherheit informationstechnischer Systeme ab und versucht eine Brücke zwischen den Themen IT Security, Safety und Dependability zu bilden. Die Vortragenden sowie das Publikum kommen aus verschiedenen Bereichen, die von der Forschung bis zu der Entwicklung und Anwendung gehen. Dadurch bietet die Veranstaltung eine Plattform für spannende Diskussionen über die neuesten Herausforderungen, Trends und Techniken in der Wissenschaft sowie in der Industrie.

Zusätzlich zu den traditionellen wissenschaftlichen Beiträgen bietet die SICHERHEIT einen Practitioners Track und ein Doktoranden-Forum mit dem besonderen Ziel der Nachwuchsförderung an. In diesem 2016 eingeführten Format wird der gegenseitige Austausch von wissenschaftlichem Nachwuchs sowie mit fortgeschrittenen Mitgliedern besonders gefördert. Mit dem Finale des CAST/GI-Promotionspreises IT-Sicherheit gibt die SICHERHEIT auch die Bühne für hervorragende Persönlichkeiten frei, die mit einer überdurchschnittlichen Leistung ihre Promotion in dem Bereich abgeschlossen haben.

Dieser Tagungsband umfasst 16 Beiträge, die von den jeweiligen Programmkomitees begutachtet und ausgewählt wurden. Diese Beiträge sind traditionsgemäß sowohl in Deutsch als auch in Englisch verfasst. Von den insgesamt 22 eingereichten wissenschaftlichen Beiträgen wurden 8 angenommen. Darüber hinaus wurden drei Beiträge zum Practitioners Track und fünf Beiträge zum Doktoranden-Forum in den Tagungsband aufgenommen.

Neben den Präsentationen dieser Beiträge haben Ulrich Kelber (Bundesbeauftragter für den Datenschutz und die Informationsfreiheit), Arslan Brömme (National Information Security Officer Germany, Vattenfall GmbH), sowie Prof. Dr. Anja Lehmann (Professorin für Cyber Security – Identity Management an der Universität Potsdam und Hasso-Plattner-Institut) eingeladene Vorträge gehalten und damit das Programm der SICHERHEIT weiter bereichern können.

Wir danken allen für ihren Beitrag zu der Tagung SICHERHEIT 2020. Dazu zählen die Autorinnen und Autoren, die Mitglieder der Programmkomitees, die externen Gutachter sowie unsere Sponsoren. Unser besonderer Dank geht an das gesamte lokale Organisationsteam sowie die Geschäftsstelle der GI für die tatkräftige Unterstützung bei der Organisation der Tagung.

Göttingen, im Januar 2020

Delphine Reinhardt
Hanno Langweg
Bernhard C. Witt
Mathias Fischer

Sponsoren

Wir danken folgenden Gold-Sponsoren für die Unterstützung der Konferenz.

Airbus CyberSecurity GmbH

ERNW Research GmbH

Fachbereich Sicherheit – Schutz und Zuverlässigkeit

Der GI-Fachbereich „Sicherheit - Schutz und Zuverlässigkeit“ wurde 2002 gegründet und vernetzt zwei „Communities“ miteinander: Während die „Safety-Community“ vor allem den Schutz der Umwelt vor IT-Systemen (beispielsweise Sicherheit des Menschen vor schwerwiegenden Systemfehlern in Flugzeugen, Kernreaktoren und Kraftwerken) sowie Fehlertoleranzmaßnahmen (z.B. Systemausfälle als Folge von Ermüdungserscheinungen, Softwarefehlern und Naturereignissen) im Blick hat, beschäftigt sich die „Security-Community“ hauptsächlich mit dem Schutz der IT-Systeme und ihrer Umgebung vor Bedrohungen von außen, insbesondere vor Gefahren, die von bösartigen Angriffen (durch Menschen) ausgehen.
Sicherheit ist ein Querschnittsthema. Für den Fachbereich gilt daher eine hohe Flexibilität hinsichtlich der Möglichkeiten zur Quervernetzung verschiedener Gruppen und Themen innerhalb und außerhalb der GI. Diese Quervernetzung wird sowohl in gemeinsamen Veranstaltungen als auch in der starken Berücksichtigung anderer Themen aus der Informatik deutlich. Sicherheit ist kein Selbstzweck, sondern wichtig zur Erfüllung gesellschaftlicher und wirtschaftlicher Bedürfnisse.

Tagungsleitung der Sicherheit 2020

Delphine Reinhardt, Universität Göttingen (General Chair)
Hanno Langweg, HTWG Konstanz (Programm Co-Chair)

Lokales Organisationteam

Patricia Nitzke, Universität Göttingen
Alexander Richter, Universität Göttingen
Luca Hernández Acosta, Universität Göttingen

Programmkomitee

Chair: Delphine Reinhardt, Universität Göttingen
Programm Co-Chair: Hanno Langweg, HTWG Konstanz

Jens-Peter Akelbein	Hochschule Darmstadt
Florian Alt	Universität der Bundeswehr München
Frederik Armknecht	Universität Mannheim
Nils Aschenbruck	Universität Osnabrück
Andreas Aßmuth	OTH Amberg-Weiden
Harald Baier	Hochschule Darmstadt
Andreas Berl	Deggendorf Institute of Technology
Rainer Boehme	Universität Innsbruck
Jens Braband	Siemens AG
Katharina Braeunlich	Universität Koblenz-Landau
Ruth Breu	Universität Innsbruck
Philipp Brune	Hochschule Neu-Ulm
Erik Buchmann	Hochschule für Telekommunikation Leipzig
Christoph Busch	Hochschule Darmstadt
Bettina Buth	HAW Hamburg
Georg Carle	TU München
Peter Dencker	Hochschule Karlsruhe
Christian Dietrich	CrowdStrike
Jana Dittmann	Universität Magdeburg
Rolf Drechsler	Universität Bremen
Eric Dubuis	Berner Fachhochschule
Paul Duplys	Robert Bosch GmbH
Manuel Duque-Anton	Hochschule Kaiserslautern
Markus Dürmuth	Ruhr-Universität Bochum
André Egners	Rohde & Schwarz Cybersecurity GmbH
Jörn Eichler	Freie Universität Berlin
Thomas Eisenbarth	Universität zu Lübeck
Dominik Engel	Hochschule Salzburg
Evren Eren	Hochschule Bremen
Sascha Fahl	Leibniz Universität Hannover
Bernhard Fechner	FernUniversität in Hagen

Hannes Federrath	Universität Hamburg
Mathias Fischer	Universität Hamburg
Ulrich Flegel	Infineon Technologies AG
Michael Franz	University of California
Felix Freiling	Friedrich-Alexander-Universität Erlangen-Nürnberg
Bernd Freisleben	Universität Marburg
Lothar Fritsch	Karlstad University
Nils gentschen Felde	Ludwig-Maximilians-Universität München
Laura Georg	Norwegian University of Science and Technology
Dieter Gollmann	TU Hamburg
Ulrich Greveler	Hochschule Rhein-Waal
Rüdiger Grimm	Universität Koblenz
Nils Gruschka	University of Oslo
Tim Güneysu	Ruhr-Universität Bochum & DFKI
Bernhard Haemmerli	ACRIS
Tobias Heer	Hochschule Albstadt-Sigmaringen
Andreas Heinemann	Hochschule Darmstadt
Maritta Heisel	Universität Duisburg-Essen
Jörg Helbach	Rheinische Fachhochschule Köln
Steffen Helke	FH Südwestfalen
Eckehard Hermann	FH Hagenberg
Dominik Herrmann	Universität Bamberg
Martin Hobelsberger	Hochschule München
Dieter Hogrefe	Universität Göttingen
Matthias Hollick	TU Darmstadt
Matthias Hudler	FH Campus Wien
Dieter Hutter	DFKI GmbH
Martin Hübner	HAW Hamburg
Jan Jürjens	Fraunhofer ISST, Universität Koblenz-Landau
Saqib A. Kakvi	Universität Paderborn
Martin Kappes	Frankfurt University of Applied Sciences
Christoph Karg	Hochschule Aalen
Frank Kargl	Universität Ulm
Stefan Karsch	FH Köln
Friedbert Kaspar	Hochschule Furtwangen
Basel Katt	Norwegian University of Science and Technoogy
Hubert B. Keller	Karlsruher Institut für Technologie
Jörg Keller	FernUniversität in Hagen
Thomas Kemmerich	Universität Bremen
Dogan Kesdogan	Universität Regensburg
Konstantin Knorr	Hochschule Trier
Klaus-Michael Koch	TECHNIKON Forschungsgesellschaft mbH
Robert Kolmhofer	FH Oberösterreich Campus Hagenberg
Dirk Koschützki	Hochschule Furtwangen
Klaus-Peter Kossakowski	c/o DFN-CERT Services GmbH
Christian Kraetzer	Otto-von-Guericke Universität Magdeburg
Christoph Krauß	Fraunhofer SIT

Ioannis Krontiris	Huawei Technologies
Ralf Kuesters	Universität Stuttgart
Hanno Langweg	HTWG Konstanz
Van Bang Le	Universität Rostock
Herbert Leitold	A-SIT
Kerstin Lemke-Rust	Hochschule Bonn-Rhein-Sieg
Peter Lipp	TU Graz
Maciej Liskiewicz	Universität zu Lübeck
Luigi Lo Iacono	TH Köln
Volkmar Lotz	SAP AG
Bernardo Magri	Aarhus University
Marian Margraf	Hochschule Darmstadt
Peter Martini	Universität Bonn, Fraunhofer FKIE
Andreas Mayer	Hochschule Heilbronn
Michael Meier	Universität Bonn, Fraunhofer FKIE
Dominik Merli	Hochschule Augsburg
Matthias Meyer	MAMEDO IT-Consulting GmbH
Ulrike Meyer	RWTH Aachen
Holger Morgenstern	Hochschule Albstadt-Sigmaringen
Isabel Münch	Bundesamt für Sicherheit in der Informationstechnik
Georg Neugebauer	oculavis GmbH
Stephan Neuhaus	Zürcher Hochschule für Angewandte Wissenschaften
Andreas Noack	Hochschule Stralsund
Alfred Nordmann	TU Darmstadt
Andriy Panchenko	Brandenburg University of Technology (BTU)
Sebastian Pape	Goethe-Universität Frankfurt
Sachar Paulus	Hochschule Mannheim
Erhard Ploedereder	Universität Stuttgart
Norbert Pohlmann	Westfälische Hochschule
Joachim Posegga	Universität Passau
Kai Rannenberg	Goethe-Universität Frankfurt
Delphine Reinhardt	Universität Göttingen
Rüdiger Reischuk	Universität zu Lübeck
Hans P. Reiser	Universität Passau
Marc Rennhard	Zürcher Hochschule für Angewandte Wissenschaften
Christian Rohner	Uppsala University
Heiko Roßnagel	Fraunhofer IAO
Andy Rupp	Karlsruher Institut für Technologie
Francesca Saglietti	Universität Erlangen-Nürnberg
Peter Schartner	Universität Klagenfurt
Björn Scheuermann	Humboldt-Universität zu Berlin
Ina Schiering	Ostfalia HAW
Rolf Schillinger	Bayerischer Forschungsverbund FORSEC
Sebastian Schinzel	FH Münster
Christian Schlehuber	DB Netz AG
Sebastian Schmerl	AGT Germany
Thomas Schmidt	HAW Hamburg

Henning Schnoor	Universität Kiel
Martin Schramm	TH Deggendorf
Sebastian Schrittwieser	SBA Research
Marko Schuba	FH Aachen
Norbert Schultes	Hochschule Koblenz
Matthias Schunter	Intel
Joerg Schwenk	Ruhr-Universität Bochum
Marcus Schöller	Hochschule Reutlingen
Jürgen Schönwälder	Jacobs University Bremen
Christoph Skornia	OTH Regenburg
Viorica Sofronie-Stokkermans	Universität Koblenz-Landau
Michael Sonntag	Johannes Kepler Universität Linz
Ingo Stengel	Hochschule Karlsruhe
Hermann Strack	Hochschule Harz
Mark Strembeck	WU Vienna
Christoph Striecks	AIT Austria
Thorsten Strufe	TU Dresden
Ramin Tavakoli Kolagari	TU Nürnberg
Paul Tavolato	FH St. Pölten
Jernej Tonejc	Fraunhofer FKIE
Stephan Trahasch	Hochschule Offenburg
Florian Tschorsch	TU Berlin
Johann Uhrmann	HAW Landshut
Johanna Ullrich	SBA Research
Melanie Volkamer	Karlsruher Institut für Technologie
Arno Wacker	Universität der Bundeswehr München
Michael Waidner	Fraunhofer SIT
Kristin Weber	FH Würzburg-Schweinfurt
Edgar Weippl	SBA Research
Steffen Wendzel	Hochschule Worms, Fraunhofer FKIE
Dirk Westhoff	Hochschule Offenburg
Lena Wiese	Leibniz-Universität Hannover
Bernhard C. Witt	it.sec GmbH
Christian Wressnegger	TU Braunschweig
Matthias Wählisch	Freie Universität Berlin
Eberhard Zehendner	Friedrich-Schiller-Universität Jena
Erik Zenner	Hochschule Offenburg
Wilhelm Zugaj	FH Joanneum

Practitioners Track

Chair: Bernhard C. Witt, it.sec GmbH

Bastian Braun	mgm security partners GmbH
Andreas Dewald	ERNW Research GmbH
Ulrich Flegel	Infineon Technologies AG
Michael Heinl	IBM Deutschland GmbH
Klaus Kirst	PTLV
Mathias Kohler	SAP Deutschland SE & Co. KG
Dirk Koschützki	Hochschule Furtwangen
Kirsten Messer-Schmidt	excepture
Isabel Münch	Bundesamt für Sicherheit in der Informationstechnik
Jürgen Neuschwander	Hochschule Konstanz
Hans Pongratz	Technical University of Munich
Manuela Reiss	dokuit
Peer Reymann	ITQS Gesellschaft für Qualitätssicherung in der Informationstechnologie mbH
Lukas Rist	The Honeynet Project
Patrizia Russ	Munich Re F&C
Andreas Schaad	Hochschule Offenburg
Claus Stark	Citigroup Global Markets Europe AG
Karin Schuler	Karin Schuler - Datenschutz und IT-Sicherheit
Jörn Voßbein	UIMC DR. VOSSBEIN GmbH & Co KG

Doktorandenforum

Chair: Mathias Fischer, Universität Hamburg

Thomas Eisenbarth	Universität zu Lübeck
Simone Fischer-Hübner	Karlstad University
Felix Freiling	Friedrich-Alexander-Universität Erlangen
Dieter Gollmann	TU Hamburg
Stefan Katzenbeißer	Universität Passau
Michael Meier	Universität Bonn
Max Mühlhäuser	TU Darmstadt
Konrad Rieck	TU Braunschweig
Stefanie Roos	TU Delft
Christian Rossow	Universität des Saarlandes
Thorsten Strufe	TU Dresden
Melanie Volkamer	Karlsruher Institut für Technologie
Edgar Weippl	Universität Wien

Zusätzliche Gutachter

Christian Rosenke	Universität Rostock
Jonas Wloka	DFKI GmbH

Inhaltsverzeichnis

Privatheit und Vertraulichkeit

Landwirtschaft und Sicherheit

Recht und Sicherheit

Practitioner Track

Doktorandenforum

Autorenverzeichnis

App-generated digital identities extracted through Android permission-based data access - a survey of app privacy

Nurul Momen[1] Lothar Fritsch [2]

Abstract: Smartphone apps that run on Android devices can access many types of personal information. Such information can be used to identify, profile and track the device users when mapped into digital identity attributes. This article presents a model of identifiability through access to personal data protected by the Android access control mechanism called *permissions*. We populate partial identities with attributes related to permission-protected personal data, and then show how apps accumulate such attributes in a longitudinal study that was carried out over several months. We found that apps' successive access to permissions accumulates such identity attributes, where apps show different interest in such attributes.

Keywords: Privacy; Android; Apps; Identification; Digital Identity; Survey and Permissions

1 Introduction and research question

Smartphones and other small communication devices using the Android operating system are tools inseparable from everyday lives for many human beings. Most aspects of communication, interaction and organization of their lives are channeled through smartphone-based apps. These devices accumulate and concentrate large amounts of personal data in contact lists, digital calendars, phone call logs, text message archives, GPS-labeled photo collections and other data. In addition the smartphone hardware offers sensing and intelligence gathering capabilities that allow data collection such as:

Frequent precise positioning through GPS or the collection of IP addresses and GSM cell information,

Collecting inventories of electronic equipment through scanning of the wireless and Bluetooth environment including own and neighbor home electronics and other person's smartphones,

Tracking phone appearance in contexts such as supermarkets, trains, airports or restaurants by mapping their Wifi hot-spots,

[1] Karlstad university, Dept. of Mathematics and Computer Science, Universitetsgatan 2, Karlstad, Sweden nurul.momen@kau.se
[2] Karlstad university, Dept. of Mathematics and Computer Science, Universitetsgatan 2, Karlstad, Sweden lothar.fritsch@kau.se

doi:10.18420/sicherheit2020_01

Acquisition of biometric and behavioural patterns through location, motion sensors, biometric sensors, cameras and microphones,

Profile media use through audio beacons in music streams and TV programs.

Such intelligence collection accumulates identifying information. The research presented in this article is concerned with how such information informs identity attributes following the model of partial identities proposed by Pfitzmann and Hansen [PH10].

Since digital identifiers can create large profiles of private behaviour when they get used and observed by third parties, the accumulation of information about partial identity attributes can cause privacy problems [PF11]. However, the exact level of how identifiable smartphone users are through such identity profile remains mostly unknown to them, since app-based information extraction and accumulation remains mainly opaque [Mo17]. Recent research on identifiability in anonymized or pseudonymized data collections shows that fifteen or less anonymized attributes are sufficient to re-identify a person in a database [RHDM19].

To investigate these matters, we populate partial identity attributes with statistically aggregated data from long-term monitoring of apps on Android platforms [Mo18a] through a study of app permission access.

Research questions:

1. *Which identity attributes can apps extract through accessing permission-protected smartphone data?* Our research aims at mapping smartphone data into identifiable attributes to show the overall collection of identity attributes extracted per app.
2. *What permission-protected identity attribute data are the apps accessing the most?* Through analysis of a longitudinal sample of app permission use we investigate the aggregated long-term identity attribute profiles gathered about smartphone users.

Outline: The rest of this article is organized as follows: First, we explain our model in Section 2 and the data acquisition and survey method for identity attribute extraction of mobile apps in Section 3. Section 4 describes our results from profiling the identity attribute extraction of 50 apps. Finally, we conclude this paper and point out directions for future research in Section 5.

2 A model for permission-based partial identity

This section will introduce our model of generated, identifiable digital identities extracted from smartphones. As shown in [FM17], there is a correlation between partial identity attributes and access control logs from Android app permission. It defines eight partial identity attributes which comprise the smartphone identity of an individual. Though only

ten permission groups are defined as *dangerous* by Android, there are more than 300 permissions listed in the master branch of its' code base which have the potential to yield identifiable information [Go19]. This section elaborates on the background of our work. Unlike previous work which focused on static analysis of app code[Ha18] or on traffic analysis[Ma12] to measure app behavior, we record permission access of running apps over a period of time.

2.1 Personal identification in digital data

Digital identity is often defined as an identifier with related identity attributes attached [Cl09]. Pfitzmann and Hansen [PH10] defined: *An identity is any subset of attribute values of an individual person which sufficiently identifies this individual person within any set of persons.* Note that seldom there is a single identity for a person, but there exist many combinations and permutations of identity attributes that are used in various sets (relating to distinct social contexts or situations they get applied to). Therefore, they introduce the concept of a partial identity by defining: *A partial identity is a subset of attribute values of a complete identity, where a complete identity is the union of all attribute values of all identities of this person.* A person is identifiable through identity attributes if s/he is easily linked to a digital identity. A person is then supposed to be linkable if he or she can get identified in a data set based on the combination of partial identity attributes used for the transaction. The concept of the partial identity is further used to define relationships between identity and attribute data as well as relationships between sets of attributes. The authors of [PH10] define anonymity, unlinkability and unobservability properties based on the concept of partial identities. One important observation is that a person is unidentifiable (anonymous) in a data set if that person's attributes cannot get identified within that data set. On the other hand, a person may become more identifiable, once more attributes get added to a partial identity. So, identifiability is directly proportional to accumulated partial identity information. Based on the concept of partial identities, we accumulate information gained through app permissions for the partial identity sets that can get retrieved through the permissions. In the next sections, a model is described for building partial identities from information accessible through permissions on Android devices and an empirical study is presented indicating the likelihood of partial identity extraction.

2.2 Partial Identities

For this survey, we use an improved version of the model for partial identities published in [FM17]. The model now leaves out the derived identity attributes. It contains additional edges mapping permission groups into identity attributes. The reduction to direct attributes serves the purpose of computability of accessed identities in the context of our survey. Analyzing the information accessible through those permissions, we mapped identity attribute building information sources to identity attributes that get collected from the

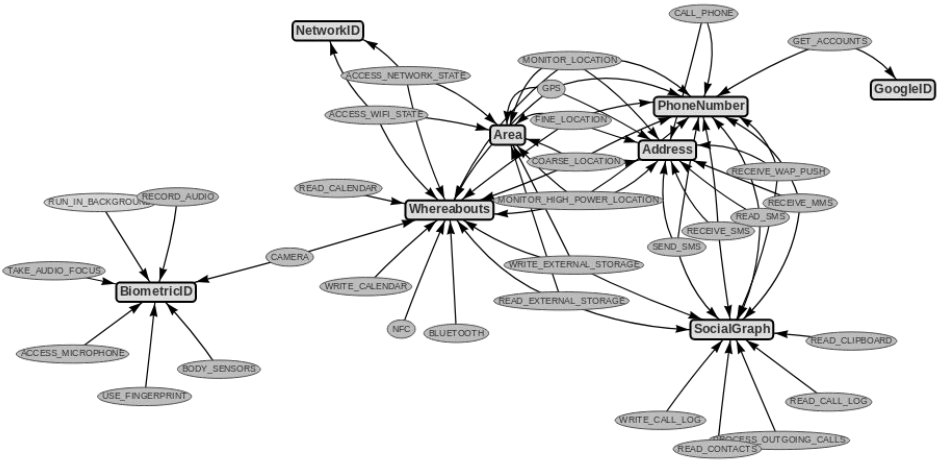


Fig. 1: Permissions (elliptical shapes) contribute to observable partial identities (rectangular shapes), expressed by directed edges.

information sources. From the permissions perspective, the partial identity P of an Android smartphone user is defined as:

$$P = \{Whereabouts, NetworkID, GoogleID, BiometricID, PhoneNumber, Address, Area, SocialGraph\}$$

It should be noted that there are directly accessible identity attributes, such as phone numbers or fine-grained GPS location, as well as data that can be used to derive identity attributes through various techniques. We defined three distinct attributes that are related to location, since the quality of their identity attribute extraction is quite different: *Whereabouts, Area, Address*. The following list defines the directly accessible partial identity attributes, lists the data contributing permissions. Permission groups are sorted into partial identity attributes they inform when extracted from a smartphone.

SocialGraph : *Gathers social graph elements from* READ_CONTACTS, READ_CLIPBOARD, CALL_LOG, EXTERNAL_STORAGE

Address : *Retrieves address from* SMS, CALL_PHONE, LOCATION

PhoneNumber : *Retrieves phone number from* CALL_PHONE, GET_ACCOUNTS, LOCATION, SMS

GoogleID : *Get Google ID from* GET_ACCOUNTS

Whereabouts : *Precise location from* LOCATION, EXTERNAL_STORAGE, NFC, CAMERA, BLUETOOTH, CALENDAR, ACCESS_WIFI_STATE, ACCESS_NETWORK_STATE

NetworkID : *Network access ID from* ACCESS_WIFI_STATE, ACCESS_NETWORK_STATE

Area : *Retrieves geographic area from* LOCATION, EXTERNAL_STORAGE, ACCESS_WIFI_STATE, ACCESS_NETWORK_STATE

BiometricID : *Collects biometric information from* CAMERA, USE_FINGERPRINT, RECORD_AUDIO, RUN_IN_BACKGROUND, BODY_SENSOR

As shown in Fig. 1, the relationship between accessed permissions and partial identity attributes can be visualized as a directed graph where the partial identities and permissions are nodes that are connected with unidirectional edges [FM17].

3 Methodology and data collection

This section describes our method and the implementation of the data collection for survey. Figure 2 shows the overall procedure. Based on popularity, we installed a sample of the 50 most popular apps from five categories: *Communication*, *Social*, *Fitness*, *Weather*, and *Music*. These apps were installed on Nokia 5 smartphones with Android 7.1.1 (Nougat). They had the pre-configured monitoring tool *Aware* installed that collected apps' permission access logs [Mo18b]. Then the partial identity model was applied to the collected data.

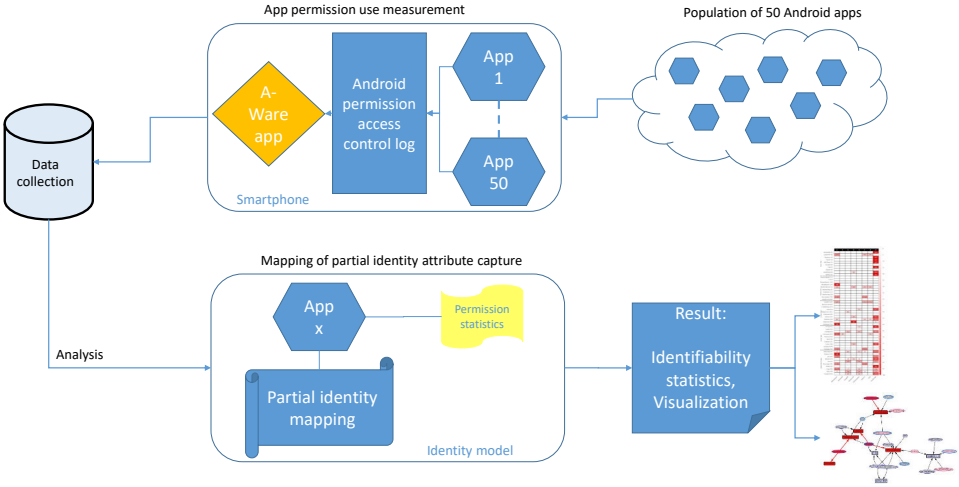


Fig. 2: Data collection: Permission logging using *Aware*, followed by mapping into partial identity model.

Capturing permission-based access control data: Accessibility to user data through permissions gives full discretionary access for the app without any constraints. We measure apps' permission access patterns using the *Aware*-client based on the method described in [Mo18b]. In brief, each permission request delivered to the Android system will be logged with date, time and location, together with the app package name. The experimental setup is described in detail in [Ha19]. *Aware* was run 24/7 on seven smartphones over the data collection period. Apps and categories considered in this article are shown in Figure 4.

Data collection: The app permission access patterns were gathered from seven Android smartphones in the time period 11 December 2018 to 11 March 2019. 50 popular Play store apps in 5 different app categories were installed. All installed apps were started, personalized and then left alone while the phones were resting or were taken to various locations. Apps were running in background at their own discretion. The smartphones had mobile 4G data access as well as configured Wifi connections to ensure permanent connectivity. In total, 664339 lines of log were accumulated in the survey period. The accumulated log data is structured as shown in Table 1.

package_name	permission_name	timestamp
com.whatsapp	READ_EXTERNAL_STORAGE	Mon Mar 11 10:36:50 GMT+01:00 2019
com.joelapenna.foursquared	MONITOR_LOCATION	Sun Mar 10 19:28:20 GMT+01:00 2019
com.google.android.apps.fitness	READ_EXTERNAL_STORAGE	Tue Feb 19 19:25:39 GMT+01:00 2019
com.yahoo.mobile.client.android.weather	MONITOR_HIGH_POWER_LOCATION	Tue Jan 15 10:02:02 GMT+01:00 2019
com.spotify.music	TAKE_AUDIO_FOCUS	Fri Mar 08 19:13:17 GMT+01:00 2019

Tab. 1: Sample of data captured from Android permission access control. Shown: App package name, permission accessed, timestamp.

Mapping the permission access data into the model of partial identities: Data selection and pre-processing were performed with the *KAUDroid* data visualization tool [SBB19] that was implemented by students at Karlstad University. The tool supports import and selection of *Aware* log files from the KAUDroid database [Ca18], the selection of apps and date ranges as well as permission groups for visualization. It implements various views on the selected data, one of which is the graph visualization of partial identity extraction based on permissions. Figure 3 shows the case of *Whatsapp*: partial identity model is applied to the permission access logs which contribute to formulate five partial identity attributes out of eight. From the collected log for *Whatsapp*, 1834 entries contributed to attribute formulation out of 3770 entries.

4 Results

Our accumulated data in Figure 4 shows large differences between apps. The figure shows in each row the percentage of accessed identity attributes per app. Numbers and coloring

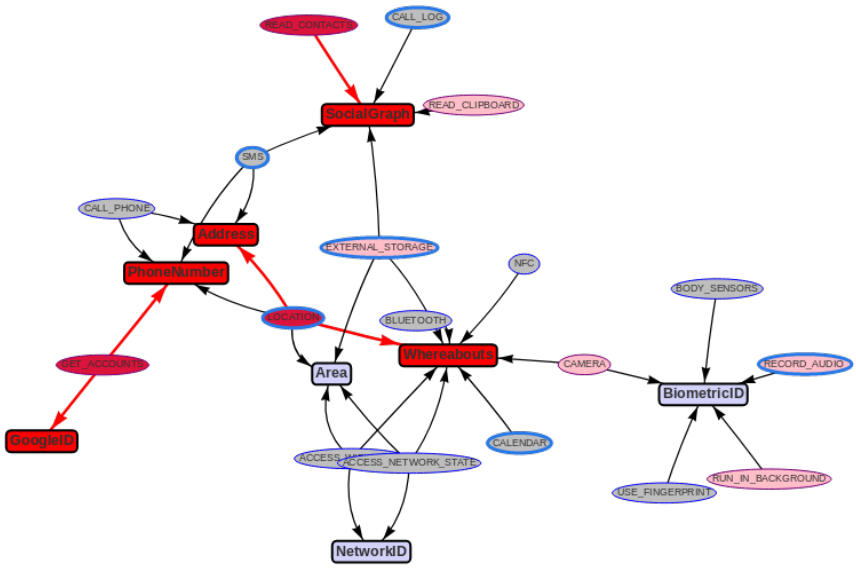


Fig. 3: Partial identity model applied on *Whatsapp*'s permission use. The extracted identity attributes are colored in intense red color, while the informing permissions linked to them that were measured are colored with light red color.*Whatsapp* extracts five identity attributes.

indicate the weighting of access. The numbers show each attributes percentage of the total attribute accesses. Each row has a total of 100 per cent. App *Pedometer* - a training tracking app - for example extracts mainly the partial identity attributes *BiometricID* (95%) and *SocialGraph* (5%), while App *Viber* - a communication and messaging app - extracts five identity attributes with a focus on *SocialGraph* (33%), *PhoneNumber* (29%) and *GoogleID* (29%).

Social relationships and location profiling: Majority of the apps extract social graph attribute related information and whereabouts (location information with high degree of detail). This is visible in columns A and H of Figure 4. We observe therefore that detailed geographic information and the social graph of a smartphone user is the most sought-after combination of identity attributes.

***Communication* and *social* apps extract most attributes:** The two app groups that show the widest extraction of identity attributes are the communication apps and the Fitness apps, with the latter group extracting half of the attributes for the firstly mentioned group. Table 2 shows the average attributes collected per group - and the number of apps with broad attribute access (4 or more).

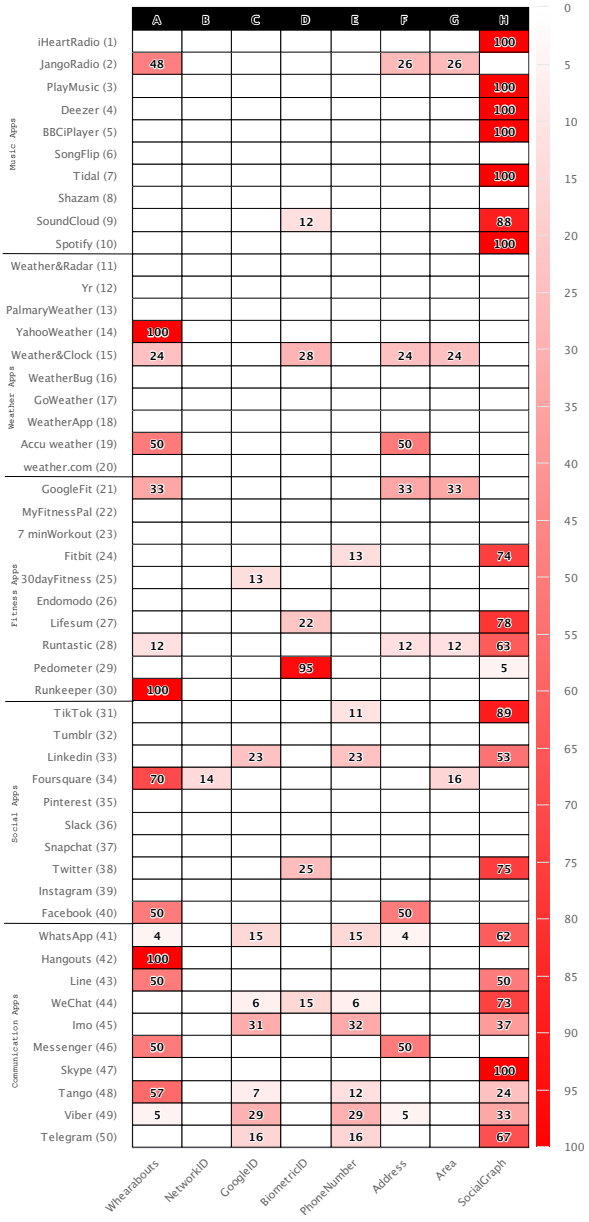


Fig. 4: Identifiability per app, as of partial identity collected. Numbers shown are percentage of access to identity attribute compared to all accesses.

	Music	**Weather**	**Fitness**	**Social**	**Communication**
Average count of attrib.	1.1	0.7	1.5	1.2	3.0
Apps using 4+ attributes	0	1	1	0	4

Tab. 2: Average number of ID attributes extracted per group and number of apps that access four or more attributes.

Upon examination of individual apps, we noticed that there are several apps that access identity attributes that may not be necessary to fulfil the advertised purpose of the respective app. *Weather&Clock* is accessing all location-related attributes, even the high-resolution ones that are much too detailed for weather forecast apps. In addition the app is interested in biometric identity attributes. A second example is *Runtastic* in the *Fitness* category which extracts *SocialGraph* attributes besides all location attributes.

Comparing Figure 4 to a former analysis of app permission showing we notice consistency of our results. In Figure 3b in [MHF19] we show that app permission access is broadest and most frequent in the communication app group. The Fitness apps and social apps are shown to be very active, too. The potential privacy risk of *Runtastic* has been noticed in a previous study as well [Ha19].

A human-friendly visual presentation: To facilitate visual analysis, we show a graph view of user identifiability through apps. The diagram in Figure 3 shows access to five partial identity attributes for *Whatsapp*. Further visualisations are provided in the appendix. The graphs are created using the *KAUDroid* tool [SBB19]. The graph view may facilitate various intuitive views. By coloring extracted attributes, it provides an overview of how identifiable a user is as seen by an app. We plan to add the magnitude of identification to the nodes such that it will be intuitive to distinguish between apps that access attributes more often and those that access attributes less.

We have to point out that our sample of 50 apps may not be representative for the larger population of apps. However we see that our analysis through partial identity profiling provides an additional perspective on app privacy issues.

5 Discussion and conclusion

Our survey shows major differences in the number of identity attributes extracted by apps. The group extracting the highest number of identity attributes are the *Communication apps*, followed by the *Fitness apps*. Concerning research question 1, we provided and tested a mapping of access to groups of Android permissions into a model of identity attributes. By collecting permission showing data, and by mapping it into our model, we showed in Figure 4 that each of our attributes is accessed by apps. We therefore constitute that we found a mapping of permission-protected data on identity attributes that works. Research question 2 provided an overview of apps that are overly concerned with collecting identity attributes

that relate to a smartphone user's social network and that provide the phone's detailed whereabouts. The other attributes in our model are extracted at much smaller numbers. We therefore answered this question by finding that social relations and location are the major identifiable attributes apps are interested in. Our findings are puzzling at first, however the social graph and detailed location profiles over time will identify a person sufficiently. Home and work locations as well as communication patterns with social contacts will not change quickly for most human beings. We must therefore expect that the re-identification potential based on a partial identity $P = \{Whereabouts, SocialGraph\}$ is sufficient for most tracking applications when used with forensic tools (see [MHW12] and [MHS11] for insights in the forensic potential of social networks and smartphone location tracks, respectively).

Utility of results and future work: Our model shows that we can rank apps by how identifiable their users are through partial identities. We see three major applications for our results. First, such rankings may help consumers make decisions about which app they will use to solve a task. Then, app rankings may constitute useful information for regulatory authorities who oversee data protection legislation and who need data about actual app behavior when data is sourced e.g. through citizen science crowdsourcing [BZNT11]. Finally, through graphical visualization, our model can provide insight for users when used on their individual data. As one path for future work we plan to combine the *Aware* data capture app with the partial identity model and the graph visualizations into a personal transparency-enhancing tool (TET) [MFH17] for end users. The tool will be useful to evaluate own exposure and identifiability against app providers. Further improvement of the data aggregation that populates the model will be necessary. As of now we equally weigh all permissions and all identity attributes independent of their contribution to identification risk and independent of their potential differences in privacy impact. We consider personalized weights as a mechanism that will allow the selection of privacy personas as default calibration of the data analysis.

Some challenges of our approach: Apps are not a static infrastructure. They constantly update themselves. Their actual behavior is triggered by their user, by their service provider pushing messages to apps, and in the case of social media the behavior is based on the social network member's activities, too. Such dynamics create difficulties for reproducible survey results. Many of those difficulties are described in Section 1.3 in [MHF19]. One major challenge is the reproduction of user behaviour including the separation of experimental use and private activities on smartphones [Go17]. To avoid major difficulties, we decided to install the apps, to personalize them by creating profiles or accounts, and then leave the devices undisturbed on our desks. The resulting data shows app behavior based on idle-time app activity. Our results are therefore presumably a subset of potential app activity with regular interaction. Additional challenges are the frequent updates of Android and especially of the permission system.

Summarizing our findings, we successfully mapped app's access to smartphone data to a model of digital partial identity which we then used to show the differences and specific

behaviors in a sample of 50 popular consumer apps. In particular, the graphical visualization can be used to inform about identifiability risk through such apps.

Acknowledgements

The research leading to this publication was partially funded by the Norwegian Research Council ArsForensica project nr.248094/O70.

References

[BZNT11] Burguera, Iker; Zurutuza, Urko; Nadjm-Tehrani, Simin: Crowdroid: behavior-based malware detection system for android. In: Proceedings of the 1st ACM workshop on Security and privacy in smartphones and mobile devices. ACM, pp. 15–26, 2011.

[Ca18] Carlsson, Adrian; Pedersen, Christian; Persson, Fredrik; Söderlund, Gustaf: KAUDroid : A tool that will spy on applications and how they spy on their users. report, Karlstad University, 2018. Project report.

[Cl09] Clarke, Roger: A sufficiently rich model of (id) entity, authentication and authorisation. In: The 2nd Multidisciplinary Workshop on Identity in the Information Society, LSE. volume 5, 2009.

[FM17] Fritsch, Lothar; Momen, Nurul: Derived Partial Identities Generated from App Permissions. In: Proceedings of the Open Identity Summit (OID) 2017, Lecture Notes in Informatics LNI, 277. Gesellschaft für Informatik, 2017.

[Go17] Goodyear, Victoria A.: Social media, apps and wearable technologies: navigating ethical dilemmas and procedures. Qualitative Research in Sport, Exercise and Health, 9(3):285–302, 2017.

[Go19] Google Git: Android Master Branch: , Number of permissions. `https://android.googlesource.com/platform/frameworks/base/+/master/core/res/AndroidManifest.xml`, 2019. Accessed on 14-Oct-2019.

[Ha18] Habib, Sheikh Mahbub; Alexopoulos, Nikolaos; Islam, Md Monirul; Heider, Jens; Marsh, Stephen; Müehlhäeuser, Max: Trust4App: automating trustworthiness assessment of mobile applications. In: 2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE). IEEE, pp. 124–135, 2018.

[Ha19] Hatamian, Majid; Momen, Nurul; Fritsch, Lothar; Rannenberg, Kai: A Multilateral Privacy Impact Analysis Method for Android Apps. In (Naldi, Maurizio; Italiano, Giuseppe F.; Rannenberg, Kai; Medina, Manel; Bourka, Athena, eds): Privacy Technologies and Policy. Springer International Publishing, Cham, pp. 87–106, 2019.

[Ma12] Marforio, Claudio; Ritzdorf, Hubert; Francillon, Aurélien; Capkun, Srdjan: Analysis of the communication between colluding applications on modern smartphones. In: Proceedings of the 28th Annual Computer Security Applications Conference. ACM, pp. 51–60, 2012.

[MFH17] Murmann, Patrick; Fischer-Hübner, Simone: Tools for Achieving Usable Ex Post Transparency: A Survey. IEEE Access, 5:22965–22991, 2017.

[MHF19] Momen, Nurul; Hatamian, Majid; Fritsch, Lothar: Did App Privacy Improve After the GDPR? IEEE Security & Privacy, 17(6):10–20, November-December 2019.

[MHS11] Maus, Stefan; Höfken, Hans; Schuba, Marko: Forensic analysis of geodata in android smartphones. In: International Conference on Cybercrime, Security and Digital Forensics, http://www. schuba. fh-aachen. de/papers/11-cyberforensics. pdf. 2011.

[MHW12] Mulazzani, Martin; Huber, Markus; Weippl, Edgar: Social network forensics: Tapping the data pool of social networks. In: Eighth Annual IFIP WG. volume 11. Citeseer, pp. 1–20, 2012.

[Mo17] Momen, Nurul; Pulls, Tobias; Fritsch, Lothar; Lindskog, Stefan: How Much Privilege Does an App Need? Investigating Resource Usage of Android Apps (Short Paper). In: 2017 15th Annual Conference on Privacy, Security and Trust (PST). pp. 268–2685, Aug 2017.

[Mo18a] Momen, Nurul: Towards Measuring Apps' Privacy-Friendliness (licentiate dissertation). Technical Report 2018:31, Karlstad University, Department of Mathematics and Computer Science, 2018.

[Mo18b] Momen, Nurul: Towards Measuring Apps' Privacy-Friendliness (licentiate thesis). PhD thesis, Karlstads universitet, 2018.

[PF11] Paintsil, Ebenezer; Fritsch, Lothar: A Taxonomy of Privacy and Security Risks Contributing Factors. In (Fischer-Hübner, Simone; Duquenoy, Penny; Hansen, Marit; Leenes, Ronald; Zhang, Ge, eds): Privacy and Identity Management for Life. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 52–63, 2011.

[PH10] Pfitzmann, Andreas; Hansen, Marit: Anonymity, unlinkability, unobservability, pseudonymity, and identity management-a consolidated proposal for terminology. In: Designing privacy enhancing technologies. Technische Universität Dresden, pp. 1–9, 10-Aug-2010.

[RHDM19] Rocher, Luc; Hendrickx, Julien M; De Montjoye, Yves-Alexandre: Estimating the success of re-identifications in incomplete datasets using generative models. Nature communications, 10(1):1–9, 2019.

[SBB19] Sundberg, Simon; Blomqvist, Alexander; Bromander, Anton: KAUdroid-Project Report: Visualizing how Android apps utilize permissions. report, Karlstad University, 2019.

Appendix A - Partial identity graphs

We show for further illustration the partial identity graphs for five selected app groups. The graphs show the app that extracts most identity attributes and the app that extracts the least attributes in the app groups. The graphs were created using the KAUDroid tool by Sundberg, Blomqvist, Bromander [SBB19].

The extracted identity attributes are colored in intense red color (dark), while the informing permissions linked to them that were measured are colored with light red color.

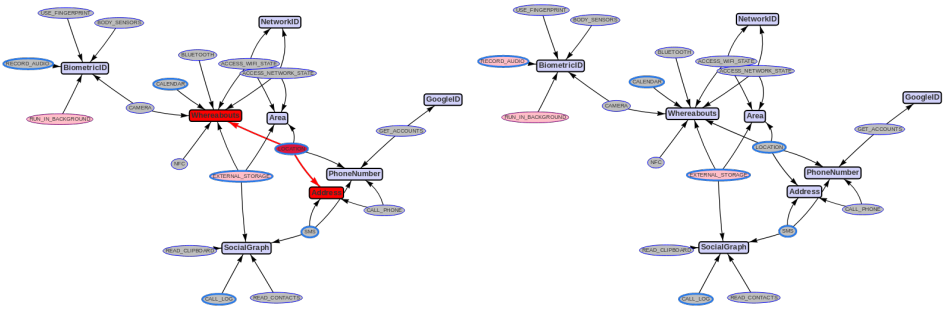


Fig. 5: JangoRadio and Shazam in *Music Apps* are the most and least user identifying apps.

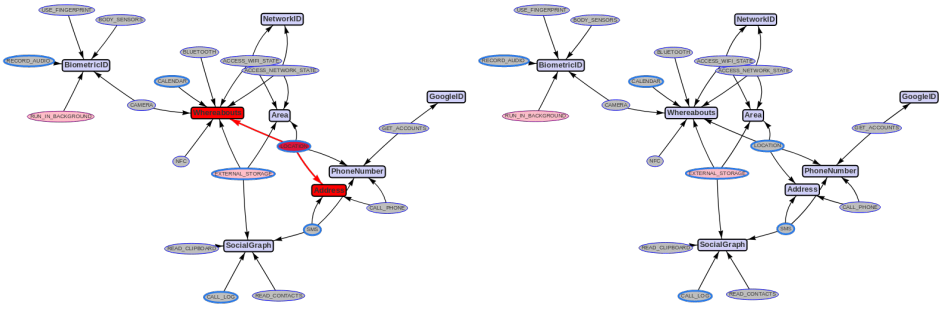


Fig. 6: Weather&Clock and PalmaryWeather in *Weather Apps* are the most and least user identifying apps.

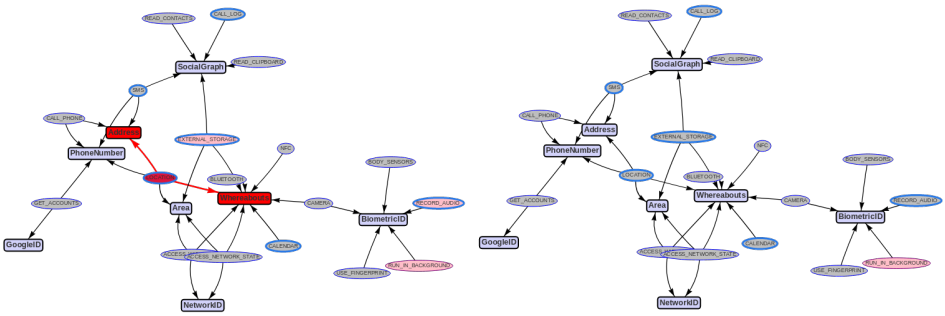


Fig. 7: Runkeeper and Endomondo in *Fitness Apps* are the most and least user identifying apps.

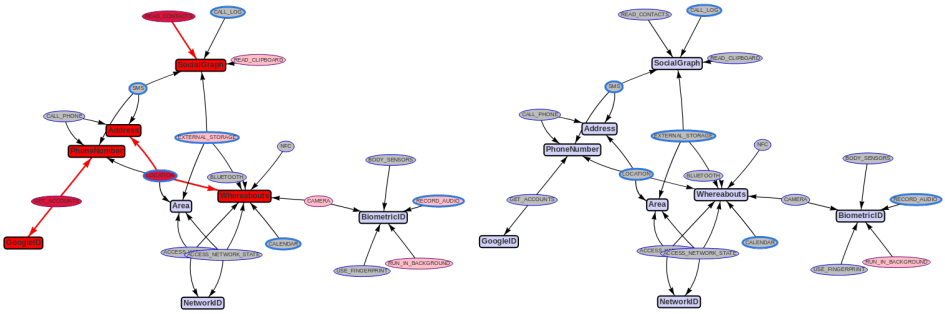


Fig. 8: Whatsapp and Instagram in *Social Apps* are the most and least user identifying apps.

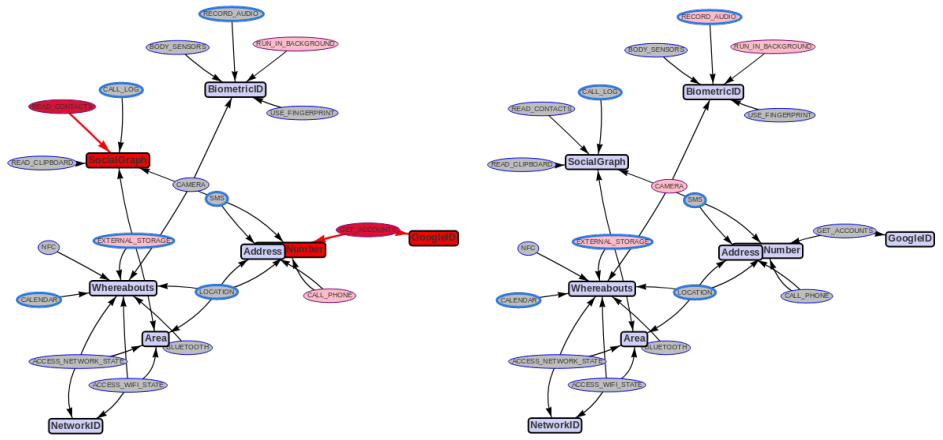


Fig. 9: Telegram and Skype in *Communication Apps* are the most and least user identifying apps.

D. Reinhardt, H. Langweg, B.C. Witt, M. Fischer et al. (Hrsg.): Sicherheit 2020,
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Göttingen 2020 29

Analyzing PeerFlow – A Bandwidth Estimation System for Untrustworthy Environments

Asya Mitseva[1], Thomas Engel[2], Andriy Panchenko[3]

Abstract: Tor is the most popular low-latency anonymization network comprising over 7,000 nodes, mostly run by volunteers. To balance user traffic over the diverse resource capabilities of those nodes, Tor users choose nodes in proportion to their available bandwidth. However, since self-reported bandwidth values are not trustworthy and active measurements put undesired load on the network, an amount of research looks for alternatives. Recently, a new bandwidth measurement system, PeerFlow, has been inroduced aiming to solve the Tor bandwidth estimation problem. In this work, we perform a practical analysis of PeerFlow. We propose a set of strategies for the practical realization of probation periods in PeerFlow and show that many Tor nodes fail to recover to their normal state after one measuring period. We also demonstrate that low-bandwidth adversaries gain significantly higher bandwidth estimates, exceeding the theoretically defined security boundaries of PeerFlow.

Keywords: Privacy; Anonymous Communication; Onion Routing; Tor; Security; Performance

1 Introduction

In the age of mass surveillance and censorship, users rely on anonymization techniques to exercise their right to freedom of expression and to freely access information. Currently, Tor [DMS04, DM18b] is the most popular anonymization network designed for low-latency applications, e.g., web browsing. To hide the identity (i.e., IP address) of a user, the user traffic in Tor is sent through a virtual tunnel, i.e., *circuit*, over three nodes, called *onion relays* (ORs). Depending on their position in the circuit, the ORs are known as *entry*, *middle*, and *exit*. A set of trusted ORs, called *directory authorities* (DAs), periodically distribute a list of available ORs together with their identities and a status description in the form of a *consensus* document [bib18]. The Tor users select ORs for circuits probabilistically according to the ORs' consensus bandwidth, availability, and exit policy to a given target [DM18a]. The users negotiate a separate symmetric key with each OR in a circuit and utilize these keys to encrypt user data in multiple layers [DM18b]. While forwarding the user data, each OR removes (or adds, depending on the direction) a layer of encryption. Thus, no OR in the circuit knows both the user and its destination at the same time.

[1] University of Luxembourg, Computer Science and Communications Research Unit, 6 Avenue de la Fonte, L-4364 Esch-sur-Alzette, Luxembourg asya.mitseva@uni.lu
[2] University of Luxembourg, Computer Science and Communications Research Unit, 6 Avenue de la Fonte, L-4364 Esch-sur-Alzette, Luxembourg thomas.engel@uni.lu
[3] Brandenburg University of Technology, Chair of IT Security, Konrad-Wachsmann-Allee 5, 03046 Cottbus, Germany andriy.panchenko@b-tu.de

doi:10.18420/sicherheit2020_02

Nowadays, Tor comprises over 7,000 ORs run by volunteers and carries terabytes of traffic for more than two million daily users [Met19]. To balance the large user traffic load over the diverse resource capabilities of the ORs, Tor entirely relied on self-reported bandwidth values in its original specification. However, malicious ORs could easily lie about their resources by advertising significantly higher bandwidth than they actually have and, thus, attract a larger fraction of traffic to be routed through them [BMG 07]. In response, Tor hardened its bandwidth estimation method by combining both self-reported values and external measurements. Nevertheless, this method remains an attractive target for adversaries aiming to compromise user connections. If an attacker can strategically run ORs and give the impression that these ORs have a large amount of available bandwidth, he can dramatically increase the likelihood of these ORs to be selected for Tor connections more often than they would have been chosen when advertising their actual capacity. As a result, these malicious ORs may either frequently appear as entry ORs in user connections, which allows them to conduct website fingerprinting [PLZ 16, SIJW18], or be chosen for both entry and exit ORs by users and, thus, execute traffic correlation [BMG 07, JWJ 13]. Some Tor users may even create Tor connections entirely comprising malicious ORs, which leads to a trivial deanonymization of these communications.

To limit the threat of attackers trying to cheat the bandwidth estimation method, Tor currently conducts periodic bandwidth measurements of all ORs executed by the DAs [LPG11, Per09]. However, previous studies [BPW13, JJH 17] showed that these measurements are easily identifiable and prone to attacks (i.e., detected measurement probes can be prioritized). The authors demonstrated that malicious ORs can make the bandwidth estimation method to believe that they have up to 177 times higher bandwidth than their actual one. Until recently, EigenSpeed, an opportunistic bandwidth estimation system proposed by Snader and Borisov [SB09], was considered as the main alternative of the current Tor bandwidth measurement system. EigenSpeed relies on passive measurements between ORs collected while communicating with each other that are reported to the DAs. The DAs employ Principal Component Analysis to compute bandwidth estimates for each OR. However, Johnson et al. [JJH 17] have recently shown fundamental flaws in this approach, which allow malicious ORs to either get a large number of honest ORs kicked out of Tor or obtain up to 28 times higher bandwidth than they actually have. In response, the authors presented the most recent proposal, PeerFlow, aiming to solve the Tor bandwidth estimation problem.

In this paper, we introduce the analysis of PeerFlow. In particular, *our contributions* are as follows: (*i*) We implement and practically evaluate the impact of probation phases (methods applied when the observed bandwidth is lower than the expected/believed one to give ORs a chance to prove that they are able to transmit traffic at a higher rate) on PeerFlow's performance. For this, we consider several strategies for the practical realization of probation periods. (*ii*) We propose a set of novel attacks against PeerFlow, which allow a low-bandwidth attacker to gain significantly higher bandwidth estimates than the theoretical security boundaries defined by the authors of PeerFlow. We show that the variable duration of measurement periods in PeerFlow introduces fundamental flaws in the approach.

2 State-of-the-art Bandwidth Estimation Systems for Tor

To limit the use of self-reported bandwidth values, Tor currently applies an active bandwidth-scanning system, TorFlow [LPG11, Per09], managed by the DAs. Each DA willing to measure ORs' bandwidth operates four bandwidth scanners. Each of these scanners is responsible for a quarter of all currently available ORs ordered by their capacity from the most recent consensus. Each bandwidth scanner further divides its list of ORs into groups, called *slices*, of ORs of similar bandwidth and uses a Tor client to create two-hop circuits via different ORs in a slice. Based on the total bandwidth of a slice, the Tor client downloads a file of a predefined size from the same, publicly-known server. This process is repeated several times as long as a sufficient quantity of measurements is collected. Each bandwidth scanner computes the average bandwidth of an OR, measured on circuits in a slice containing this OR. Once per hour, each measuring DA aggregates these values and estimates the amount of available bandwidth of each OR. This is achieved by multiplying the self-reported bandwidth of the given OR by the ratio of the measured bandwidth of that OR to the average network bandwidth. This data is then used to produce the consensus bandwidth values of ORs.

Despite its main goal to make Tor resistant to bandwidth manipulation, previous research [BPW13, JJH 17] showed that TorFlow remains vulnerable to common attacks. As the self-reported bandwidth is a multiplier in the TorFlow's bandwidth estimation formula, malicious ORs can still announce significantly higher bogus bandwidth than their actual one and influence on their consensus bandwidth values (as shown in [BMG 07]). Even worse, an adversary's OR can easily identify measurement circuits due to their specific length and the publicly-known IP addresses of the measuring nodes. Consequently, it can provide more bandwidth to those circuits while suppressing all others. This, in turn, gives the impression that this OR has high capacity by using a meager amount of resources.

Bauer et al. [BMG 07] proposed the use of a proactive distributed probing of nodes, e.g., by applying anonymous auditing [SNDW06]. Snader et al. [SB08] suggested that ORs collect bandwidth statistics for other ORs during an interaction with them and announce these statistics to the DAs. Instead of measuring node's bandwidth capacity, several authors [SBL09, PR09] proposed that Tor users create paths based on link characteristics, e.g., latency, jitter, loss, to reduce the likelihood of manipulation.

To address the limitations of TorFlow, Snader et al. [SB09] suggested a bandwidth estimation system, EigenSpeed, based on passive measurements. In EigenSpeed, each OR records the speed of each connection with another OR. The collected data is weighted with the current exponentially-weighted moving average of the connections and reported to the DAs. EigenSpeed was considered to provide secure bandwidth estimation [SB11] and was suggested for deployment in Tor before Johnson et al. [JJH 17] revealed major flaws of the system. To overcome the discovered drawbacks, the authors proposed another bandwidth

Bandwidth scanners belonging to a single DA use the same Tor client running on the DA itself [LPG11].
https://trac.torproject.org/projects/tor/ticket/5464

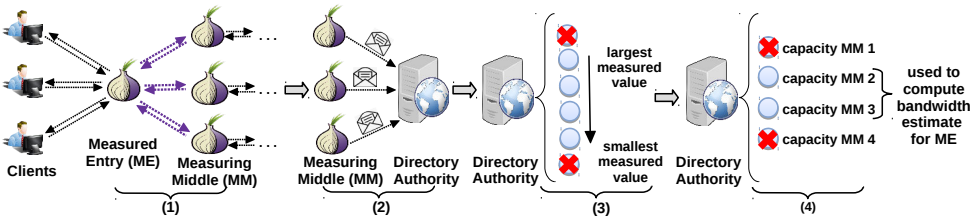


Fig. 1: A brief overview of the PeerFlow operation.

measurement system, PeerFlow, relying on a bandwidth-weight voting mechanism to compute consensus bandwidth of ORs. Contrary to previous work that mainly focus on developing alternative bandwidth estimation systems for Tor, in our work we introduce the first analysis of PeerFlow.

3 Operation of PeerFlow

PeerFlow [JJH 17] relies on a bandwidth-weight voting mechanism to calculate the consensus bandwidth of ORs. Figure 1 illustrates the main steps from the operation of PeerFlow. Based on the idea of EigenSpeed, PeerFlow utilizes passive measurements collected by the ORs while communicating between each other. To this end, each OR counts the number of bytes sent to or received from another OR and adds a random noise to each measurement to prevent information leakage (step 1). This data is periodically collected by each OR and reported to the DAs (step 2). The DAs, in turn, weigh each measurement with the probability of choosing the corresponding measuring OR in a given circuit position (i.e., whether the measuring OR can be selected for an entry, middle or exit node in that circuit). Then, the DAs sort the data gathered for each measured OR according to its possible circuit position (step 3). The largest and smallest values are discarded. Next, the DAs assign to each measurement a voting weight proportional to the *current* capacity of the corresponding measuring OR (i.e., the OR that reported this measurement) and take into account only those statistics whose voting weights are positive and fall into a predefined interval (step 4). The current OR's capacity is estimated based on the traffic that the OR was able to transfer in its last measurement period. Finally, the DAs aggregate the selected statistics to compute the total traffic r_t relayed by an OR r. If r_t is significantly lower than the amount of traffic that the OR r is *expected* to relay (based on history) and its self-reported bandwidth is significantly larger than r_t, the relay enters a *probation* state whereas its consensus bandwidth is not decreased yet. The expected amount of traffic transmitted by an OR is calculated based on the median of inferred bytes transferred by the set of ORs that can be

According to the Tor specification, only ORs which are able to fulfill a set of predefined requirements (e.g., availability, exit policy, etc.) can be chosen for an entry or exit position in a circuit. Based on OR's available bandwidth, the DAs further calculate *consensus weights* used to balance traffic load across different circuit positions that a single OR can be selected for. We omit further details about the selection of an OR for a given circuit position due to space constraints and refer the reader to [DM18b] for more information.

selected for the same position in the last measurement period. If in the next measurement period the OR r does not manage to prove that it is able to transmit traffic at a given rate (by exchanging dummy traffic with a set of measuring ORs), the DAs reduce the consensus bandwidth of this OR. As the authors of [JJH 17] do not specify how ORs are informed when they enter a probation state, in Section 5 we discuss possible methods used by the ORs to detect their probation state. The duration of measurement periods is variable for each OR and depends on how quickly the OR can exchange an amount of user traffic with each measuring OR. The goal of this variability of the measurement periods is to guarantee that the overall random noise added to the reported measurements is relatively small and, thus, ensure accurate bandwidth estimates. Moreover, new ORs joining Tor are selected for middle position only and their measurements are not considered for a given period of time to prevent manipulation.

4 Experimental Setup

To allow for verifiable results of our analysis, we next present our experimental setup. For our evaluation, we deployed the same Tor network configuration used in [JJH 17] consisting of four DAs, 498 ORs, 7,500 Tor users, and 1,000 web servers as traffic destinations. We simulated the operation of the private Tor network by utilizing Shadow [JH12] – a parallel discrete-event network simulator particularly designed for Tor network experimentation. Like [JJH 17], each experiment was repeated three times to measure variability across runs.

Simulation of dummy traffic: For the evaluation of each strategy used to distribute dummy traffic by ORs in probation, we count the median and maximum number of ORs in probation state in a given second. We also compute *inaccuracy* and *imprecision* to take into account errors in bandwidth weight calculations. The inaccuracy represents the ratio of the true relative bandwidth of an OR to the relative consensus bandwidth of the same OR. The true relative capacity of an OR, in turn, is the ratio of the true bandwidth of that OR to the true total network bandwidth. The relative consensus bandwidth of an OR is the ratio of the consensus bandwidth of that OR to the total network consensus bandwidth. We also calculate the imprecision of the estimated consensus bandwidth of each OR over different consensuses. In the rest of the paper, the inaccuracy is averaged across all ORs and consensuses whereas the imprecision is averaged across all ORs in an experiment. For both metrics, we show the average and standard deviation across the three experiments.

Adversarial model: To analyze different attacks against PeerFlow, we assume an active adversary aiming to manipulate the operation of PeerFlow by giving the impression that it has a high amount of bandwidth by using a meager amount of resources. Similar to the authors of PeerFlow, we focus on an adversary who controls several ORs representing 4% of the total network bandwidth.

5 Probation Periods in PeerFlow

In PeerFlow, an OR enters a probation state when the amount of traffic relayed by this OR in its last measurement period was lower than the amount of traffic that the OR was expected to transmit. During its probation period, the consensus bandwidth of that OR is not decreased yet and the OR has the possibility to prove that it is able to transmit traffic at a given rate by exchanging dummy traffic with a set of (measuring) ORs. The authors of [JJH 17] did not fully develop the probation in their proposal, but simply measured how often ORs that sent little traffic would be put into probation state. However, this is not sufficient to explore the performance of PeerFlow and its resistance to bandwidth manipulations in real settings. For instance, a fraction of the measuring ORs may be malicious and may not cooperate with honest ORs in probation. Here, we analyze different strategies for the practical realization of probation periods in PeerFlow and reveal their influence on the Tor performance.

To implement the probation in PeerFlow, we assume that all ORs can estimate the set of currently measuring ORs in Tor (i.e., ORs with positive probability of choosing them in a given circuit position and positive voting weights). This is a realistic assumption as the voting weights of the ORs used to determine the set of measuring ORs are proportional to the consensus bandwidth of these ORs. We also assume that all ORs know the start- and end-time of their measurement periods and when they enter the probation state. To accomplish this in reality, the ORs can keep track of which measuring ORs they communicated with (and the bytes exchanged with them) and can roughly estimate when their measurement period finishes and the amount of traffic that would be observed by the DAs. Once an OR enters the probation state, it starts exchanging dummy traffic with measuring ORs to prove that it is capable to transmit traffic at a given rate [JJH 17]. Since all ORs know their maximum bandwidth advertised to the network and can keep information about the number of bytes exchanged with other ORs, the ORs in probation are also able to calculate the amount of free bandwidth that can be used for dummy traffic with measuring ORs. The measuring ORs, in turn, send the collected dummy reports to the DAs. Contrary to [JJH 17], in our work the DAs do not automatically (regardless of the measurement results) recover ORs in probation in the next measurement period. If an OR in probation does not manage to recover in the next measurement period, the DAs decrease its capacity to a fraction ϵ_{dec} over its last consensus bandwidth. As suggested in [JJH 17], we used $\epsilon_{dec} = 0.25$ to allow a natural variation in traffic amounts and at the same time to avoid too much underperformance of ORs. We also implemented the update of voting weights of the measuring ORs in the PeerFlow prototype based on the design presented in [JJH 17]. The practical realization of voting updates is important for our work as the number of inferred bytes computed from reports of a measuring OR relies on the voting weight of this OR. The larger the voting weight of a measuring OR is, the larger is the final inferred estimate of the measured OR.

Based on what amount of dummy traffic is distributed among different measuring ORs, we study several strategies for the realization of dummy traffic generated by ORs in probation. The first strategy, called *dummy for all measuring ORs*, assumes that an OR in probation knows the set of all measuring ORs for a given circuit position in the current measurement

Tab. 1: Weight errors, median, and maximum number of relays in probation.

PeerFlow	Inaccuracy		Imprecision		ORs in probation	
	Avg.	Std. Dev.	Avg.	Std. Dev.	Median	Max
Baseline	0.229	0.001	0.398	0.0052	0	13
Dummy for all measuring ORs	0.226	0.0048	0.369	0.0067	25	121
Dummy for measuring ORs without connections	0.228	0.0078	0.365	0.011	7	71
Dummy for measuring ORs with connections	0.263	0.0041	0.410	0.0038	7	78
Optimistic strategy	0.233	0.0069	0.369	0.0045	16	46

period. The OR exchanges equal amount of dummy traffic with all of these measuring ORs regardless whether it has some existing exchange with some of them or not. If the OR in probation can be simultaneously selected for two circuit positions, i.e., it acts as entry and middle or exit and middle, we first check if it has enough amount of unused bandwidth to exchange dummy traffic for both positions. If not, we select the set of measuring ORs that contains fewer measuring ORs. Thus, the OR in probation can exchange more dummy bytes with each of the measuring ORs. As shown in Table 1, out of the 498 ORs the median number of ORs in probation is at most 25 and the maximum is at most 121. A possible reason for the significantly high number of ORs in probation compared to the original approach may be that most of these ORs do not have enough capacity to communicate with all measuring ORs. In other words, they exchange a little amount of dummy traffic with each of the measuring ORs, which is not sufficient to influence the final estimation by the DAs. Moreover, as the DAs do not consider the reports from all measuring ORs (i.e., the collected statistics are trimmed before computing the total number of inferred bytes), a part of the real traffic will not be considered by the DAs at all. This, in turn, hinders ORs in probation to recover from the probation state as the remaining (dummy) traffic is not sufficient for that.

Instead of connecting to all measuring ORs for a given circuit position, in the second strategy, called *dummy for measuring ORs without connections*, the OR in probation prefers to contact only those measuring ORs, which it has not communicated with in the current measurement period yet and exchanges an equal amount of dummy traffic with each of them. Here, the median number of ORs in probation reduces to at most 7 and the maximum is at most 71. The obtained results confirm our hypothesis presented in the previous paragraph. Hence, this is a superior strategy for probation period. We also examine the strategy, called *dummy for measuring ORs with connections*, where an OR in probation exchanges an equal amount of additional, dummy traffic only with those measuring ORs it has already communicated with in the current measurement period. I.e., the measuring ORs add the dummy traffic to already existing reports for the ORs in probation and send them to the DAs. The median number of ORs in probation remains 7 and the maximum is at most 78.

We further try to optimize the latter scenario. We consider an *optimistic strategy* where

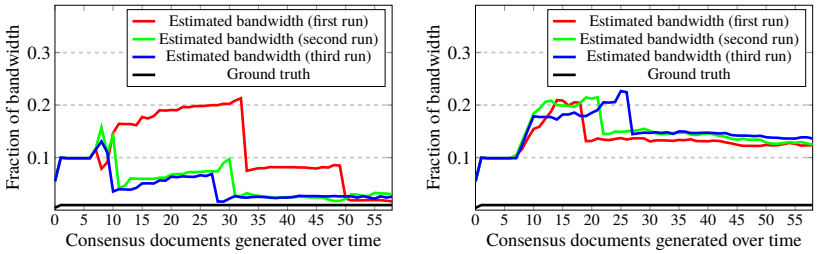
we assume that an OR in probation can correctly filter only those measuring ORs whose statistics will be considered by the DAs after trimming. Having this knowledge would allow ORs to optimally distribute their bandwidth to boost the ranking. The OR in probation sums the number of bytes already reported by those measuring ORs for it and its unused bandwidth in order to compute the average amount of traffic that can be exchanged with each of the non-trimmed ORs in total. The difference between the total average number of bytes and the real inferred bytes is transmitted in the form of dummy traffic and added by the measuring ORs to their statistics reported to the DAs. To the best of our knowledge, this is the best strategy that ORs in probation phase can apply to increase their ranking without tearing down user connections. While this strategy significantly reduces the maximum number of ORs in probation to 46, the median number of ORs in probation increases to 16, as shown in Table 1. Similar to the first strategy, a possible reason for this may also be the fact that a larger number of relays in probation do not manage to recover from the probation state in the next measurement period. Although each of the measuring ORs reports the maximum number of bytes that can be exchanged with the OR in probation on average, the DAs assign different weights to the distinct reports based the voting weight of the measuring OR and the time interval needed to transmit this data.

To sum up, both strategies *dummy for measuring ORs with connections* and *dummy for measuring ORs without connections* achieve the minimum median number of ORs in probation in a given second whereas the *dummy for measuring ORs without connections* reduces the maximum number of ORs in probation to 7 ORs more contrary to *dummy for measuring ORs with connections*. For the maximum number of ORs in probation, the best results are achieved by the *optimistic strategy*. Nevertheless, the set of ORs in probation remains significantly higher than expected by the authors [JJH 17]. In addition, we also observed that most of the ORs falling in probation were not able to recover within a single measurement period. In terms of inaccuracy and imprecision, all strategies behave consistent and similar. Therefore, to optimize median number of ORs in probation periods in PeerFlow, we apply the strategy *dummy for measuring ORs without connections* in the rest of our work.

6 Attacks on PeerFlow

In this section, we propose and analyze three new attacks against PeerFlow. In our evaluation, we assume that the attacker exploits the fact that PeerFlow uses TorFlow measurements to determine the bandwidth capacity of newly bootstrapping ORs. As shown in [JJH 17], a malicious OR can dramatically inflate its consensus bandwidth by providing more bandwidth to measuring TorFlow circuits and suppressing all others. We simulate this when our malicious ORs join in the network and inflate their consensus bandwidth by factor of 10.

Attack 1: We first analyze a simple attack, in which our malicious ORs communicate with other honest ORs as usual, but do not establish connections between each other. At the same time, they send dummy statistics for other malicious ORs with high bandwidth values and artificially decrease the bandwidth values reported for the honest ORs. Figure 2(a)



(a) Attack 1 (b) Attack 2

Fig. 2: Fraction of total consensus bandwidth obtained by PeerFlow for our malicious ORs.

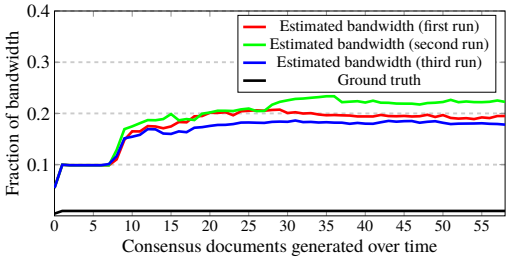


Fig. 3: Fraction of total consensus bandwidth obtained by PeerFlow for our malicious ORs in Attack 3.

shows the fraction of bandwidth estimated by PeerFlow for these ORs versus the fraction of their real bandwidth capacity. After completing the first measurement period and taking into account the first Peerflow measurements, we confirm that the bandwidth inflation of the malicious ORs is limited according to the security boundaries presented in [JJH 17]. Nevertheless, we see that at the beginning of the experiment for a certain period of time our malicious relays manage to gain bandwidth inflation factor of up to 9 (compared to the security boundary of 4.6 for small adversaries defined by the authors).

Attack 2: Although we were able to keep a high bandwidth inflation of the malicious ORs for a given time period, the success of the attack depends on the fact whether malicious ORs are selected as measuring ORs. According to the authors, the DAs will publish the last known bandwidth value of an OR in the consensus as long as the current PeerFlow measurement period does not finish. Based on this knowledge, we try to improve the latter attack by exploiting the fact that the length of the next measuring period of our malicious ORs rely on the number of measuring ORs they communicate with and the amount of traffic they transmitted to those measuring ORs. In other words, if our malicious ORs connect to a restricted number of measuring ORs and transmit very little amount of traffic or they do not connect to them at all, their measuring period will never end. This, in turn, will force the DAs to keep their initial bandwidth estimate obtained by the TorFlow measurements. As

shown in Figure 2(b), we see that in the first half of the experiment the fraction of consensus bandwidth gained by the malicious ORs is approx. 30 times higher than their actual capacity.

Attack 3: In spite of the large bandwidth inflation achieved above, we still observe that some of the malicious ORs are not able to keep their measuring period infinitive. A reason for this is the fact that our malicious ORs try to avoid connections to currently measuring ORs. However, the set of measuring ORs is changing permanently. This means that measurements done during previous connections to other ORs can be considered later by the DAs if these ORs become measuring relays. Therefore, we further improve the latter attack by avoiding malicious ORs to connect to other high-capacity ORs, i.e., whose consensus bandwidth is higher than the real capacity of our malicious ORs. Thus, we try to prevent communications with possible future measuring ORs. Figure 3 shows the obtained results. As it can be seen, we are able to keep the inflated consensus bandwidth of our malicious ORs constantly high.

To sum up, we showed that low-bandwidth adversaries are able to gain significantly higher bandwidth estimates exceeding the theoretically defined security boundaries of PeerFlow. The main reason for this is the fact that PeerFlow completely rely on the already vulnerable TorFlow measurements during the bootstrapping period of new ORs joining Tor. We further demonstrated that the variable duration of measurement periods in PeerFlow introduces fundamental flaws in the approach. In particular, malicious ORs can selectively refuse connections with measuring ORs and make their first PeerFlow measurement period never end. As a result, the inflated bandwidth of the malicious ORs obtained by TorFlow will be kept in the consensus, which, in turn, will attract more Tor users to use these ORs.

7 Conclusion

In this paper, we presented the first analysis of the most recent proposal, PeerFlow, aiming to solve the Tor bandwidth estimation problem. To this end, we implemented and evaluated different strategies for the practical realization of probation periods in PeerFlow and showed that many honest ORs cannot recover to their normal state after one measuring period. A possible solution would be that ORs in probation prioritize the exchange of dummy traffic to recover faster to their normal state. Nevertheless, we argue that the practical realization of probation periods needs further research before adopting PeerFlow into Tor. Thereafter, we showed that low-bandwidth adversaries can gain significantly higher bandwidth estimates exceeding the security boundaries of PeerFlow due to two undesirable issues. First, PeerFlow relies on initial bandwidth measurements prone to bootstrapping attacks. Second, as the measurement periods in PeerFlow have a variable length, malicious ORs can selectively refuse connections with measuring ORs and make their first measurement period never end. A possible solution would be to keep track of ORs with often unsuccessful connection establishments and report them to the DAs. Expect evaluating this, for future work we plan to extend our analysis on PeerFlow to identify whether low-bandwidth adversaries can gain a control over a set of measuring ORs and, thus, further manipulate PeerFlow estimations.

Bibliography

[bib18] Tor Directory Protocol, Version 3. `https://gitweb.torproject.org/torspec.git/tree/dir-spec.txt`, 2018.

[BMG 07] Kevin Bauer, Damon McCoy, Dirk Grunwald, Tadayoshi Kohno, and Douglas Sicker. Low-resource Routing Attacks Against Tor. In *Workshop on Privacy in Electronic Society*, pages 11–20, Alexandria, USA, October 2007. ACM.

[BPW13] Alex Biryukov, Ivan Pustogarov, and Ralf-Philipp Weinmann. Trawling for Tor Hidden Services: Detection, Measurement, Deanonymization. In *Symposium on Security and Privacy*, pages 80–94, Berkeley, USA, May 2013. IEEE.

[DM18a] Roger Dingledine and Nick Mathewson. Tor Path Specification. `https://gitweb.torproject.org/torspec.git/plain/path-spec.txt`, 2018.

[DM18b] Roger Dingledine and Nick Mathewson. Tor Protocol Specification. `https://gitweb.torproject.org/torspec.git/plain/tor-spec.txt`, 2018.

[DMS04] Roger Dingledine, Nick Mathewson, and Paul Syverson. Tor: The Second-generation Onion Router. In *13th conference on USENIX Security Symposium*, San Diego, USA, August 2004. USENIX Association.

[JH12] Rob Jansen and Nicholas Hopper. Shadow: Running Tor in a Box for Accurate and Efficient Experimentation. In *19rd Network and Distributed System Security Symposium*, San Diego, USA, February 2012. Internet Society.

[JJH 17] Aaron Johnson, Rob Jansen, Nicholas Hopper, Aaron Segal, and Paul Syverson. PeerFlow: Secure Load Balancing in Tor. *17th Symposium on Privacy Enhancing Technologies*, pages 74–94, July 2017.

[JWJ 13] Aaron Johnson, Chris Wacek, Rob Jansen, Micah Sherr, and Paul Syverson. Users Get Routed: Traffic Correlation on Tor by Realistic Adversaries. In *20th conference on Computer and Communications Security*, pages 337–348, Berlin, Germany, November 2013. ACM.

[LPG11] Karsten Loesing, Mike Perry, and Aaron Gibson. Bandwidth Scanner specification. `https://github.com/AdrienLE/torflow/blob/master/NetworkScanners/BwAuthority/README.spec.txt`, 2011.

[Met19] Tor Metrics. `https://metrics.torproject.org/`, 2019.

[Per09] Mike Perry. TorFlow: Tor Network Analysis. In *2nd Hot Topics in Privacy Enhancing Technologies*, Seattle, WA, USA, August 2009. Springer.

[PLZ 16] Andriy Panchenko, Fabian Lanze, Andreas Zinnen, Martin Henze, Jan Pennekamp, Klaus Wehrle, and Thomas Engel. Website Fingerprinting at Internet Scale. In *23rd Network and Distributed System Security Symposium*, San Diego, USA, February 2016. Internet Society.

[PR09] Andriy Panchenko and Johannes Renner. Path Selection Metrics for Performance-Improved Onion Routing. In *9th IEEE/IPSJ Symposium on Applications and the Internet*, Seattle, USA, July 2009. IEEE.

[SB08] Robin Snader and Nikita Borisov. A Tune-up for Tor: Improving Security and Performance in the Tor Network. In *16th Network and Distributed System Security Symposium*, San Diego, USA, February 2008. Internet Society.

[SB09] Robin Snader and Nikita Borisov. EigenSpeed: Secure Peer-to-peer Bandwidth Evaluation. In *8th International Conference on Peer-to-peer Systems*, Boston, USA, April 2009. USENIX Association.

[SB11] Robin Snader and Nikita Borisov. Improving Security and Performance in the Tor Network through Tunable Path Selection. *IEEE Transactions on Dependable and Secure Computing*, 8(5):728–741, September–October 2011.

[SBL09] Micah Sherr, Matt Blaze, and Boon Thau Loo. Scalable Link-based Relay Selection for Anonymous Routing. In *9th Symposium on Privacy Enhancing Technologies*, Seattle, USA, August 2009. Springer.

[SIJW18] Payap Sirinam, Mohsen Imani, Marc Juarez, and Matthew Wright. Deep Fingerprinting: Undermining Website Fingerprinting Defenses with Deep Learning. In *25th conference on Computer and Communications Security*, pages 1928–1943, Toronto, Canada, October 2018. ACM.

[SNDW06] Atul Singh, Tsuen-Wan "Johnny" Ngan, Peter Druschel, and Dan S. Wallach. Eclipse Attacks on Overlay Networks: Threats and Defenses. In *25th International Conference on Computer Communications*, Barcelona, Spain, April 2006. IEEE.

D. Reinhardt, H. Langweg, B.C. Witt, M. Fischer et al. (Hrsg.): Sicherheit 2020,
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Göttingen 2020 41

Passive Angriffe auf kanalbasierten Schlüsselaustausch

Kompromittierung der Zufallsquelle unter Verwendung deterministischer Kanalmodelle

Paul Walther, Rober Knauer, Thorsten Strufe [1]

Abstract: Im Bereich der Physical Layer Security stellt der Schlüsselaustausch auf Basis von Kanalreziprozität eine effiziente Methode der Schlüsselaushandlung dar. Eine zentrale Annahme ist, dass die gemessenen Kanaleigenschaften ein geteiltes Geheimnis darstellen, worauf die Schlüsselgenerierung basiert. In dieser Arbeit wird diese Grundannahme angegriffen, in dem mit Hilfe von deterministischen Kanalmodellen und dem Wissen über Raumgeometrie und Terminalpositionen diese Kanaleigenschaften approximiert werden. Es wird eine Methodik entwickelt, um derartige Angriffe zu bewerten und auf Basis des *Kunisch-Pamp* Modells werden Approximationen durchgeführt. Die Resultate zeigen, dass die rekonstruierten Kanaleigenschaften trotz nicht-optimaler Voraussetzung sehr stark mit den tatsächlichen korrelieren und dies einen validen Angriffsvektor darstellt.

Keywords: Schlüsselaustausch; PhySec; Angriff; CRKG

1 Einleitung

Physical Layer Security (PhySec) stellt energieeffizient kryptographische Primitive bereit, deren Sicherheit informationstheoretisch beweisbar ist [BB11]. Ausgehend von Ahlswedes Quellenmodell [AC93] ist der Schlüsselaustausch, basierend auf Kanalreziprozität (Channel Reciprocity based Key Generation, CRKG), ein Primitiv zur Ableitung symmetrischer Schlüssel. Hierbei wird der reziproke Funkkanal zwischen zwei Kommunikationsteilnehmern als gemeinsame Entropiequelle für die Schlüsselgenerierung verwendet. Eine der Grundannahmen dieses Prozesses ist, dass die charakteristischen Eigenschaften dieses reziproken Kanals bereits ein geteiltes Geheimnis darstellen: basierend auf Jakes Streuungstheorem wird davon ausgegangen, dass alle weiteren Funkterminals, die weiter als eine halbe Wellenlänge entfernt sind, abweichende Kanalcharakteristika messen und damit keine Informationen erhalten, die ihnen helfen können, den generierten Schlüssel zu erraten.

Orthogonal dazu wurden durch umfangreicher Messstudien und Analysen immer exaktere und leistungsfähigere Kanalmodelle für Funkkanäle entwickelt, z.B. das IEEE Ultraweitband Modell [MFP03] oder das Modell von Kunisch und Pamp [KP02].

[1] TU Dresden, Datenschutz und Datensicherheit, {vorname.nachname}@tu-dresden.de
Diese Arbeit wurde mit Unterstützung des Bundesministerium für Bildung und Forschung (BMBF) im Rahmen des Projektes "5G NetMobil"(Förderkennzeichen: 16KIS0689) erstellt.

doi:10.18420/sicherheit2020_03

In der vorliegenden Arbeit werden diese beiden Felder zusammengeführt und daraus ein Angriff auf CRKG entwickelt. Basierend auf Messungen im Vorfeld und Kenntnissen über die Raumgeometrie und aktuellen Terminalpositionen während des Schlüsselaustausches werden mit Hilfe des deterministischen Kanalmodells von Kunisch und Pamp die Charakteristika des Funkkanals zwischen legitimen Kommunikationspartnern rekonstruiert und damit die Kanalimpulsantworten zwischen beiden Parteien geschätzt. Basierend auf einer hinreichend genauen Rekonstruktion kann ein Angreifer die Schlüsselgenerierung parallel ausführen und potentiell den gleichen Schlüssel ableiten.

Die Resultate zeigen, dass das gewählte Kanalmodell in der Lage ist Kanaleigenschaften zu rekonstruieren, die denen des legitimen Kanals ähnlich sind. Dadurch stellt dieser Ansatz einen validen Angriffsvektor dar.

Es werden die folgenden Beiträge gemacht:

- Vorstellung eines Angriffsansatzes basierend auf deterministischen Kanalmodellen
- Entwicklung einer geeigneten Methodik zur Bewertung eines derartigen Angriffs
- Umsetzung und Analyse des Konzept auf Basis des *Kunisch-Pamp* Kanalmodells

Abschnitt 2 gibt aktuelle Angriffskonzepte zu CRKG wieder. In Abschnitt 3 werden das System- und das Angreifermodell beschrieben. Abschnitt 4 beschreibt die entwickelte Methodik und Abschnitt 5 legt die Realisierung und erreichten Resultate dar. Abschließend fasst Abschnitt 6 zusammen und gibt einen Ausblick auf weitere Fragen.

2 Stand der Wissenschaft

CRKG unterliegen einige sehr starke Systemannahmen, was zu einer Vielzahl darauf in der Literatur vorgeschlagener passiver, wie aktiver Angriffe geführt hat. Die folgende Auswahl beschreibt die Kernideen, die die Grundlagen der unterschiedlichen Angriffe bilden.

In [EKY11] stellten die Autoren in Frage, ob die Kanaleigenschaft mit steigender Entfernung tatsächlich dekorrelieren. Hierbei konnten nicht vernachlässigbare Korrelationen selbst bei Entfernungen von mehreren Wellenlängen festgestellt werden. Die Autoren entwickeln daraus jedoch keinen erfolgreichen Angriff. Döttling et al. schlugen einen sogenannten *reradiation side-channel attack* vor [Dö10]. Da diesem Angriff zahlreiche und mitunter gleichfalls starke Annahmen zu Grunde liegen, welche in lediglich vereinfachten Szenarien überprüft wurden, erscheint dieses Szenario wenig praxisrelevant. In [Ja09] wurden die Veränderungen von RSSI-Werten in diversen statischen und dynamischen Szenarien untersucht. Sie zeigen, dass in statischen Umgebungen die Varianz der Messwerte sehr gering ist, was darauf schließen ließe, dass Approximationen praktisch möglich wären.

Bezüglich aktiven Angriffen ist die Arbeit [JZ15] von Jin und Jeng erwähnenswert: durch die Injektion von bekannten Signalen in den Messvorgang sollten die Eingangswerte der Schlüsselgenerierung manipuliert werden. In [Eb12] wurde dieser Ansatz erweitert, so dass

entweder bekannte Signale zur Schlüsselgenerierung genutzt wurden oder der Austausch behindert wurde. Beide Angriffe sind aktiver Natur und betrachten ausschließlich RSSI-Werte. Kürzliche Arbeiten schlagen für CRKG die Nutzung der Kanalimpusantworten vor, die strikt mehr Information – und dem Vernehmen nach mehr Entropie als RSSI-Werte enthalten. Die hier vorgestellten Angriffe lassen sich auf diese Schlüsselgenerierung also nicht anwenden.

3 Systemmodell und Angreifermodell

Das **Systemmodell** geht von Innenraum-Funkkommunikation im Ultraweitband (UWB) Bereich aus. Die Kanaleigenschaften werden in Form von Kanalimpulsantworten (Channel Impuls Response - CIR) aufgezeichnet, wobei die charakteristischen Eigenschaften durch die im UWB deutlichen Mehrwegekomponenten bereitgestellt werden.

Dem **Angreifermodell** liegt ein praktisch orientierter Ablauf des Angriffes zugrunde, was zwei unterschiedliche Phasen definiert, in denen sich der Angreifer unterschiedlich verhält. Die Grundidee des Angriff kann folgendermaßen beschrieben werden:

Phase 1 - Vorbereitung Ein lokaler Angreifer kann in dem Raum, in dem die zukünftige Kommunikation des Schlüsselaustausches stattfinden wird, beliebige aktive Messungen vornehmen, z.B. bzgl. der Raumgeometrie oder Referenzmessungen von Kanalimpulsantworten mit bekannten Terminalpositionen.
Auf Basis dieser Messungen wird eine repräsentative Darstellung des Funkkanals durch den Angreifer entwickelt.

Phase 2 - Angriff Während des eigentlichen Angriffs auf die Zufallsquelle der Schlüsselgenerierung werden die Informationen aus der Vorbereitung mit denen der aktuellen Kommunikation kombiniert, um die Schlüsselgenerierung zu kompromittieren. Hierzu muss sich der Angreifer nicht in der Nähe befinden und theoretisch weder aktiv noch passiv mit den berechtigten Kommunikationspartner interagieren . Auf Basis der in Phase 1 gewonnen Repräsentation wird versucht die Messwerte der legitimen Kommunikationspartner zu rekonstruieren. Im Übrigen wird davon ausgegangen, dass außer den Positionen der Terminals keine weiteren Informationen vorliegen.

Beide Phasen sind in der Praxis problemlos realisierbar, wobei der Angreifer über keinerlei besondere Ressourcen bzgl. verwendeter Technik, Rechenaufwand oder Zeit verfügen muss. Vor allem die zweite Phase stellt besonders wenig Anforderungen an den Angreifer, da Positionsdaten einfach erhoben werden können (z.B. physische Nähe oder auch visuell durch Fenster oder Überwachungskameras).

Aus praktischer Sicht wäre es natürlich sinnvoll, Daten aufzuzeichnen, die mit diesem Schlüssel verarbeitet wurden, um ein höher gelegenes Ziel des Angreifers zu realisieren. Für die vorliegenden Angriff auf den Schlüsselaustausch ist das aber nicht notwendig.

4 Methodik

Methodisch muss zwischen zwei Komponenten unterschieden werden: zu einem ist relevant, welches **Kanalmodell** der Angreifer verwendet und zum anderen mit welcher **Metrik** der Erfolg des jeweiligen Modells gemessen wird.

Kanalmodell Hauptkriterium der Modellwahl ist der Determinismus des Modells, damit aus bekannten Positionen bidirektional eindeutige CIRs berechnet werden können. Daher sind bekannte randomisierte Kanalmodelle, wie das Saleh-Valenzuela Modell [SV87] oder das IEEE UWB Channel Model [MFP03] für den vorliegenden Fall ungeeignet.

Im Bereich der deterministischen Kanalmodelle hat sich das Modell von Kunisch und Pamp bewährt [KP02]. Da es primär für die Beschreibung von Funkkanälen in Gebäuden entworfen wurde, entspricht es genau dem verwendeten Systemmodell.

Das *Kunisch-Pamp*-Modell hat eine Vielzahl an Parametern — die Wichtigsten für den vorliegende Fall sind die Geometrie des Raumes und die Positionen der Funkterminals. Laut Angreifermodell ist die Raumgeometrie bekannt, so dass die Terminalpositionen die einzigen freien Simulationsparameter sind. Die weiteren Modellparameter werden entweder festgelegt oder an Hand von Messungen optimiert. Eine wichtige Einschränkung ist, dass dieses Modell in der gegenwärtigen Realisierung nur rechteckige Räume unterstützt.

Metriken Um die rekonstruierten Kanaleigenschaften mit den gemessenen zu vergleichen, muss eine Metrik mit folgenden Eigenschaften gefunden werden:

Hohe Trennschärfe Die Hauptaufgabe der Metrik ist eine klare Unterscheidung zwischen “ähnlichen” und “unähnlichen” Kanalimpulsantworten. Die CIRs für gleiche Positionen sollten also einen hohen Wert dieser Metrik erzielen, während CIRs an unterschiedliche Positionen niedrige Werte erzielen sollten.

Normalisierung Laut den Annahmen von CRKG sind die reziproken Eigenschaften von Kanalimpulsantworten die einzelnen Mehrwegecluster und deren Eigenschaften [Ja09]. Explizit ausgeschlossen wird die absolute Stärke der Impulsantwort, die variieren kann. Daher sollten Unterschiede in den absoluten Werten ignoriert, besser noch kompensiert werden.

Zeitsynchronität Da die lokalen Terminals nicht zeitlich synchronisiert sind, muss die verwendete Metrik robust gegenüber zeitlichen Verschiebungen sein.

Die Haupteigenschaft hierbei ist eine hohe Trennschärfe der Metrik, da dies das Hauptziel ist. Die beiden weiteren Eigenschaften sind grundsätzlich durch zusätzliche Vorverarbeitungsschritte erreichbar — da zusätzliche Verarbeitungsschritte im Sinne von Komplexitätsreduktion und Fehleranfälligkeit nicht optimal sind, ist eine Metrik, die diese Eigenschaften implizit hat, zu bevorzugen.

Durch die Grundidee des Signalvergleichs bieten sich für diese Metrik besonders an:

Mean Squared Error Der klassische MSE [SS07] ist weder robust gegen Unterschiede der Verstärkung noch gegen zeitliche Verschiebung. Daher wird die in [MBS98] vorgestellte Variante verwendet, die normalisiert:

$$m_1(g,h) = \min_{\beta} \frac{\|\vec{g} - \beta\vec{h}\|^2}{\|\vec{g}\|^2} = 1 - \left(\frac{\vec{g}^T \vec{h}}{\|\vec{g}\|\|\vec{h}\|} \right)^2$$

Korrelationskoeffizient Für den vorliegenden Fall ist der Korrelationkoeffizient definiert als

$$m_2(g,h) = \frac{\sum_{k=0}^{N-1} g[k]h[k] - N\bar{g}\bar{h}}{\sqrt{\sum_{k=0}^{N-1}(g[k]-\bar{g})^2 \sum_{k=0}^{N-1}(h[k]-\bar{h})^2}}$$

mit

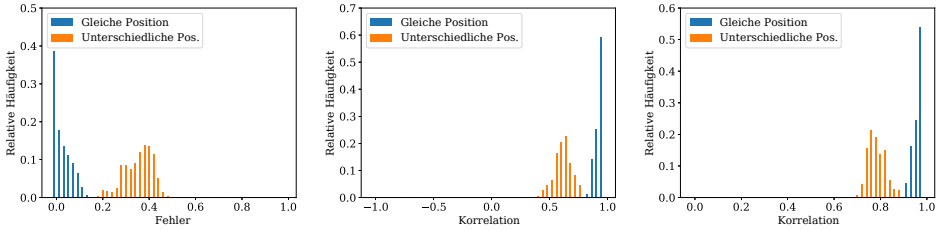
$$\bar{g} = \frac{1}{N}\sum_{k=0}^{N-1} g[k] \qquad \bar{h} = \frac{1}{N}\sum_{k=0}^{N-1} h[k]$$

Er ist normalisierend, aber anfällig für zeitliche Verschiebungen.

Kreuzkorrelation Wird die klassische Kreuzkorrelation (siehe [Mi16, p. 393]) auf Basis der Energien der Impulsantworten normalisiert, dann stellt deren Maximum eine Metrik dar, die sowohl normalisierend als auch robust gegen zeitliche Verschiebungen ist.

$$m_3(g,h) = \max_k \frac{r_{gh}(k)}{\sqrt{E_g E_h}} = \max_k \frac{\sum_{i=-\infty}^{\infty} g[i]h[i-k]}{\sqrt{\sum_{i=0}^{n_g-1} g[i]^2 \sum_{i=0}^{n_h-1} h[i]^2}}$$

Als Voruntersuchung wurden diese 3 Metriken auf alle in Kapitel 5.1 beschriebenen Messungen angewandt, um deren Eigenschaften zu untersuchen. Die Abbildung 1 zeigt die unterschiedlichen Metriken angewandt auf die verfügbaren Kanalimpulsantworten.



(a) m_1 : Mean Squared Error (b) m_2 : Korrelationskoeffizient (c) m_3 : Kreuzkorrelation

Abb. 1: Resultate der verschiedenen Metriken m_1, m_2 und m_3.

Aus der Abbildung 1 ist ersichtlich, dass alle drei Metriken die notwendige Trennung erreichen. m_1 und m_2 mussten hierfür zusätzlich bzgl. der zeitlichen Verschiebung der

einzelnen Realisierungen optimiert werden. Das heißt, dass m_3 bei geringem algorithmischen Aufwand die gleiche Trennschärfe erreicht. Tabelle 1 fasst diese Eigenschaften zusammen.

Metrik	Trennung	Normalisierung	Synchronisation
m_1: MSE	ja, mit Vorverarbeitung	ja	nein
m_2: Korrelationkoeffizient	ja, mit Vorverarbeitung	ja	nein
m_3: Kreuzkorrelation	ja	ja	ja

Tab. 1: Eigenschaften der betrachteten Metriken

Auf Basis dieser Voruntersuchung wird für die Evaluation die Metrik m_3 basierend auf der Kreuzkorrelation verwendet.

5 Umsetzung und Resultate

Nachfolgend werden die Realisierungen des Angriffes und die erzielten Resultate dargelegt.

5.1 Messungen

Um die praktische Realisierbarkeit des Angriffs zu zeigen und die dabei erzielbaren Resultate zu analysieren, wurden dem Systemmodell folgend umfassende Messungen von UWB Kanalimpulsantworten vorgenommen. Um für den Innenraum typische Messwerte zu generieren, wurden die Messungen in einem Büroraum durchgeführt. Der Raum und die entsprechenden Terminalposition sind in Abbildung 2 abgebildet.

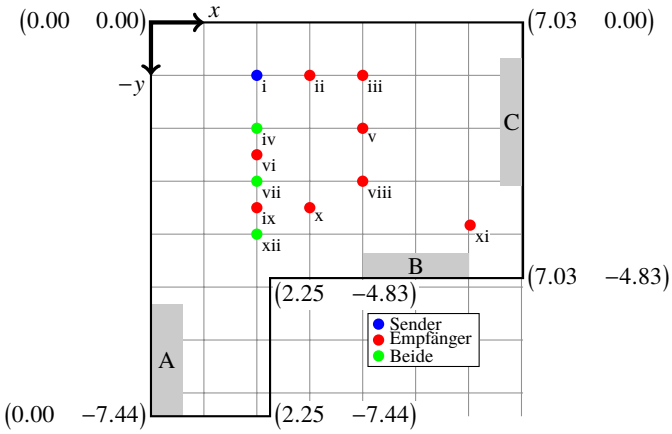


Abb. 2: Die Messumgebung mit der Raumgeometrie, den Terminalpositionen i–xii und den Hindernissen A–C. Alle Koordinaten in Metern.

Für die in Tabelle 2 aufgelisteten Kombinationen von Terminalpostionen wurden jeweils zwischen 1014 und 1112 Kanalimpulsantworten aufgezeichnet.

#	Sender	Empfänger	#	Sender	Empfänger	#	Sender	Empfänger
1	i	xi	7	i	x	13	iv	ii
2	i	iv	8	i	viii	14	vii	xi
3	i	vi	9	i	v	15	vii	ii
4	i	vii	10	i	iii	16	xii	xi
5	i	ix	11	i	ii	17	xii	ii
6	i	xii	12	iv	xi			

Tab. 2: Testläufe mit jeweiligen Positionskombinationen

Die Messungen wurden mit Hilfe von Decawave EVB1000 UWB Evaluation Boards durchgeführt, da diese durch ein Abtastintervall von $\Delta\tau = 2ns$ sehr feingranulare CIRs bereitstellen. Entsprechend der Definitionen von UWB wurden die Messungen im $4\,GHz$ Band mit einer Bandbreite von $500\,MHz$ durchgeführt. Um innerhalb der Kanalkohärenzzeit des Kanals zu bleiben, wurden die einzelnen Impulsantworten mit einem Abstand von $\Delta t = 200\,ms$ aufgezeichnet. Hierbei liefern die Transceiver pro CIR $N = 1016$ Werte, womit eine einzelne Messung eine Länge von $\approx 2\mu s$ hat.

Die komplexen Impulsantworten $g(\tau)$ wurden vorab in reellwertige umgewandelt, indem der Betrag gebildet wurde $|g(\tau)| = \sqrt{\Re(g(\tau))^2 \quad \Im(g(\tau))^2}$, da die Phase keine signifikante Reziprozität liefert [Wa18].

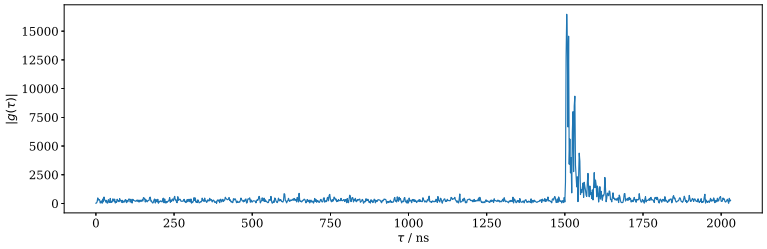


Abb. 3: Beispielhafte reellwertige Kanalimpulsantwort aus dem Messlauf #2

Abbildung 3 zeigt eine beispielhafte Kanalimpulsantwort. Da messtechnisch bedingt ein Großteil der Messungen Rauschen beinhaltet, wurde an Hand des Rauschlevels die Messung auf den “interessanten” Teil zugeschnitten.

5.2 Simulation und Optimierung

Basierend auf dem Angriffskonzept wurden für die Messungen Simulationen auf Basis des gewählten Kanalmodells durchgeführt.

Die Parameter des Modells sind in zwei Klassen unterteilt: zum einen *feste Parameter*, deren Werte durch die Umgebung vorgegeben sind (z.B. Raumgeometrie, Basisband, . . .). Zum anderen *optimierte Parameter*, deren Werte auf Basis der Referenzmessungen durch Bayes’sche Optimierung bestimmt wurden [Sh15].

Tabelle 3 zeigt die Modellparameter mit entsprechenden Werten/Wertebereichen .

Optimierte Parameter		**Feste Parameter**	
Parameter	Wertebereich	Parameter	Wert
K	Virtuelle Quellen	f	$4\,GHz$
d_0	$0,5\,d_{LOS} - 1,5\,d_{LOS}$	f_s	$500\,MHz$
α	$2,0 - 3,0$	n	$2\,ns$
G_{MP}	$-25\,dB - -10\,dB$	G	$120\,db$
$G_{MP,LOS}$	$-15\,dB - 0\,dB$	β	1.2
γ	$9\,ns - 13\,ns$		

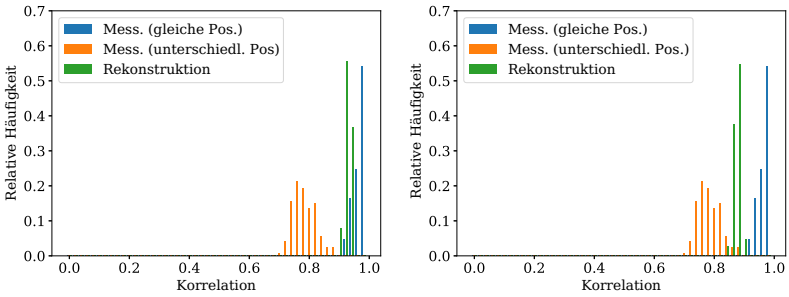
Tab. 3: Modellparameter des *Kunisch-Pamp* Kanalmodells.

Für die Optimierung wurden 2 unterschiedliche Strategien verfolgt:

Individuelle Optimierung Hierbei wurden die Modellparameter für die Simulation eines Testlaufes mit Messwerten ausschließlich aus diesem Messlauf optimiert. Das vorrangige Ziel dieser Optimierung ist die allgemeine Machbarkeit der Signalrekonstruktion zu zeigen und zu belegen, dass das gewählte Kanalmodell ein geeigneter Kandidat dafür ist.

Angreiferorientierte Optimierung Die Testläufe $R = 1, 12, 14, 16$ stellen mögliche Angreiferpositionen dar. Bei diesem Optimierungsansatz wurden ausschließlich Messungen dieser 4 Messungen verwendet, um die Modellparameter zu optimieren. Der Fokus dieser Optimierung ist die Praxisrelevanz des beschriebenen Angriffs.

5.3 Resultate



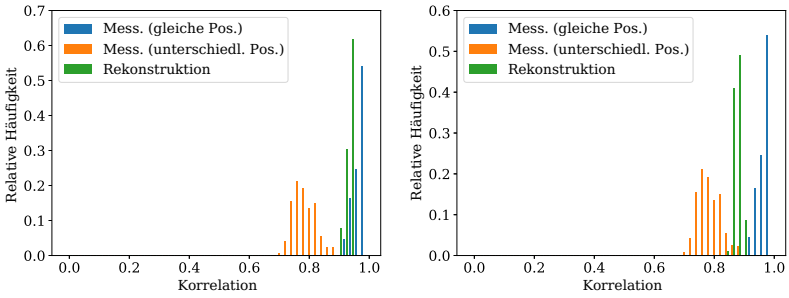
(a) Bester Testlauf #15, $\bar{c} = 0.936$ (b) Schlechtester Testlauf #16, $\bar{c} = 0.883$

Abb. 4: Bestes und schlechtestes Ergebnis der individuellen Optimierung.

Im Kontext der individuellen Optimierung konnten die folgenden Resultate erzielt werden. Abbildung 4 zeigt die Histogramme des besten als auch des schlechtesten Testlaufs. Hierbei wurden die physischen Messungen jeweils mit den rekonstruierten Signal mit der gewählten Metrik ausgewertet. Das Histogramm zeigt die relativen Häufigkeiten der Metrik-Resultate.

Die Bedeutungen der einzelnen Modellparameter können der Veröffentlichung [KP02] entnommen werden.

Nachdem dem Kanalmodell realistisches Rauschverhalten hinzugefügt wurde, konnten die Resultate nochmals verbessert werden. Die erreichten durchschnittlichen Kreuzkorrelationen konnte auf bis zu $0,941$ erhöht werden. In den Abbildungen 5 sind erneut die besten und schlechtesten Testläufe für diese Optimierung zu sehen.



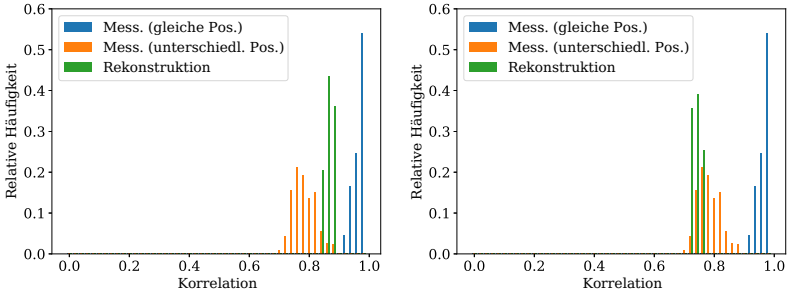
(a) Bester Testlauf #10, $\bar{c} = 0.941$ (b) Schlechtester Testlauf #16, $\bar{c} = 0.883$

Abb. 5: Bestes und schlechtestes Ergebnis der individuellen Optimierung mit Rauschen.

Diese Werte zeigen, dass das Modell grundsätzlich geeignet ist, UWB Kanalimpulsantworten auf Basis von Raumgeometrie und Terminalpositionen zu approximieren.

Aufbauend auf diesen Erkenntnissen wurde nun die **angreiferorientierte Optimierung** durchgeführt, welche ausschließlich die Testläufe $R = 1, 12, 14, 16$ zur Optimierung verwendet und anschließend CIRs für alle Positionen approximiert (siehe Abbildung 6).

Abbildung 7 zeigt die Mittelwerte aller Optimierungen im Vergleich.



(a) Bester Testlauf #13, $\bar{c} = 0.874$ (b) Schlechtester Testlauf #5, $\bar{c} = 0.748$

Abb. 6: Bestes und schlechtestes Ergebnis der angreiferorientierten Optimierung mit Rauschen.

6 Zusammenfassung und Ausblick

In dieser Arbeit wurde eine der Grundannahmen von CRKG angegriffen: dass die Entropiequelle als Grundlage der Schlüsselgenerierung nur den legitimen Kommunikationspartnern

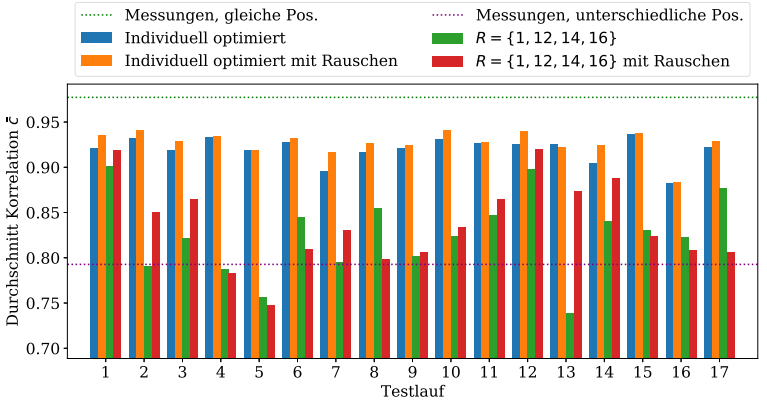


Abb. 7: Zusammenfassung aller Optimierungen und Testläufe

zur Verfügung steht. Ausgehend vom UWB Kontext wurden mit Hilfe des deterministischen *Kunisch-Pamp* Kanalmodells, optimiert durch Vorabmessungen des Angreifers, die der Schlüsselgenerierung zu Grunde liegenden Messungen approximiert.

Die Resultate zeigen deutlich, dass das gewählte Kanalmodell für deterministische Vorausberechnungen von Kanalimpulsantworten geeignet sind: die individuell optimierten Rekonstruktionen erreichen für alle angegriffenen Testläufe eine Kreuzkorrelation im Bereich von $0,93 - 0,94$, was deutlich über dem durchschnittlichen Wert für unterschiedliche Positionen von $0,79$ liegt. Weiterhin liegt es nur 3 Prozentpunkte unter dem durchschnittlichen Wert für gleiche Positionen von $0,97$ und über der empirischen unteren Schranke für gleiche Position von $0,89$. Das heißt, dass ein Großteil der in der Impulsantworten enthaltenen Informationen deterministisch vorausberechnet werden konnten. Berücksichtigt man die weiteren Verarbeitungsschritte, besonders die *Information Reconciliation* [BB11] und deren Methodik mit zusätzlichem Informationsabfluss kann von erfolgreichen Schlüsselberechnung auf Basis dieser Simulation ausgegangen werden.

Bemerkenswert für das Angriffsszenario ist, dass für die deterministische Simulation der Impulsantworten lediglich 4 Messungen des Angreifers nötig sind und der Angreifer *keine perfekten Informationen über den Raum* haben muss. Die vorliegenden Resultate konnten erzielt werden, obwohl der verwendete Raum *nicht* den Anforderungen des Kanalmodells entspricht. Es ist zu erwarten, dass eine entsprechende Erweiterung des Kanalmodells für beliebige Räume eine deutliche Verbesserung der Rekonstruktion bringt.

Für eine finale Bewertung des Angriffs muss die Auswirkung der Annäherung noch konkreter abgeschätzt werden. Dies könnte darüber stattfinden, die Transinformation zwischen Simulation/Messwerten mit der der reziproken Messungen in Verhältnis zu stellen. Da eine analytische Berechnung nicht praktikabel scheint arbeiten wir derzeit daran, geeignete Entropie- oder Transinformationsschätzer zu finden.

Literaturverzeichnis

[AC93] Ahlswede, R.; Csiszar, I.: Common Randomness in Information Theory and Cryptography. I. Secret Sharing. IEEE Transactions on Information Theory, 39(4):1121–1132, 1993.

[BB11] Bloch, Matthieu; Barros, Joao: Physical-Layer Security: From Information Theory to Security Engineering. Cambridge University Press, 2011.

[Dö10] Döttling, Nico; Lazich, Dejan; Müller-Quade, Jörn; de Almeida, Antonio Sobreira: Vulnerabilities of wireless key exchange based on channel reciprocity. In: International Workshop on Information Security Applications. Springer, S. 206–220, 2010.

[Eb12] Eberz, Simon; Strohmeier, Martin; Wilhelm, Matthias; Martinovic, Ivan: A practical man-in-the-middle attack on signal-based key generation protocols. In: European Symposium on Research in Computer Security. Springer, S. 235–252, 2012.

[EKY11] Edman, Matthew; Kiayias, Aggelos; Yener, Bülent: On passive inference attacks against physical-layer key extraction? In: Proceedings of the Fourth European Workshop on System Security. ACM, S. 8, 2011.

[Ja09] Jana, Suman; Premnath, Sriram Nandha; Clark, Mike; Kasera, Sneha K; Patwari, Neal; Krishnamurthy, Srikanth V: On the effectiveness of secret key extraction from wireless signal strength in real environments. In: Proceedings of the 15th annual international conference on Mobile computing and networking. ACM, S. 321–332, 2009.

[JZ15] Jin, Rong; Zeng, Kai: Physical layer key agreement under signal injection attacks. In: 2015 IEEE Conference on Communications and Network Security (CNS). IEEE, S. 254–262, 2015.

[KP02] Kunisch, J.; Pamp, J.: Radio Channel Model for Indoor UWB WPAN Environments. Bericht IEEE P802.15-02/281, IEEE 802.15.3a, 2002.

[MBS98] Morgan, D.R.; Benesty, J.; Sondhi, M.M.: On the evaluation of estimated impulse responses. IEEE Signal Processing Letters, 5(7):174–176, jul 1998.

[MFP03] Molisch, Andreas F; Foerster, Jeffrey R; Pendergrass, Marcus: Channel models for ultrawideband personal area networks. IEEE wireless communications, 10(6), 2003.

[Mi16] Mitra, S.K.: Signals and Systems. Oxford Series in Electrical and Computer Engineering. Oxford University Press, 2016.

[Sh15] Shahriari, Bobak; Swersky, Kevin; Wang, Ziyu; Adams, Ryan P; De Freitas, Nando: Taking the human out of the loop: A review of Bayesian optimization. Proceedings of the IEEE, 104(1):148–175, 2015.

[SS07] Sharif, Zaiton; Sha'ameri, Ahmad Zuri: The Application of Cross Correlation Technique for Estimating Impulse Response and Frequency Response of Wireless Communication Channel. In: 2007 5th Student Conference on Research and Development. IEEE, 2007.

[SV87] Saleh, A. A. M.; Valenzuela, R.: A Statistical Model for Indoor Multipath Propagation. IEEE Journal on Selected Areas in Communications, 1987.

[Wa18] Walther, Paul; Janda, Carsten; Franz, Elke; Pelka, Mathias; Hellbrück, Horst; Strufe, Thorsten; Jorswieck, Eduard: Improving Quantization for Channel Reciprocity based Key Generation. In: 2018 IEEE 43rd Conference on Local Computer Networks (LCN). 2018.

D. Reinhardt, H. Langweg, B.C. Witt, M. Fischer et al. (Hrsg.): Sicherheit 2020,
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Göttingen 2020 53

Context-based Access Control and Trust Scores in Zero Trust Campus Networks

Thomas Lukaseder[1] Maya Halter[1] Frank Kargl[1]

Abstract: Research networks are used daily by thousands of students and scientific staff both for education and research. Therefore, they contain a vast number of sensitive and valuable resources. The currently predominant perimeter security model is failing more and more often to provide sufficient protection against attackers. This paper analyses to what extent the *Zero Trust Model* that is popular in some commercial networks can also be applied to the open and heterogeneous research network of a German university. The concept presented herein to implement such an identity-based network model focuses in particular on the components which are necessary for authentication and authorization. The feasibility of the model is demonstrated by a self-implemented prototype that protects access control to a prominent eLearning system called Moodle. Non-functional performance tests show an increase in performance compared to the current system where access control is only conducted inside the web application. The Zero Trust Model enables the determination of the trustworthiness of individual identities and thus offers valumable new ways to secure a research network.

Keywords: Network Security; Network Management; Zero-Trust; Trust Scores; Subjective Logic

1 Introduction

Research networks on university campuses are used daily by thousands of students and scientific staff for education and research. This means that they contain a large number of sensitive and valuable resources such as grades or research results. Many users bring their own computers, smartphones, or IoT devices. As device security is often not centrally managed but left to device owners, this creates a rather large attack surface.

These are some reasons why the currently predominant perimeter security models — relying mostly on firewalls and securing access to the network — fail more and more often to provide sufficient protection against attackers. Some organizations have realized this early. One example is Google that was hit by severe attacks from nation state attackers in 2009. The attack was termed "Operation Aurora"[2] and targeted GMail accounts of Chinese dissidents along with source code repositories. Googles reacted with the project *"BeyondCorp"*, a complete restructuring of their network security approach based on the idea of Zero Trust. They describe the design, implementation, and roll-out in a series of articles [Be17;

[1] Universität Ulm, Institut für Verteilte Systeme, Albert-Einstein-Allee 11, 89081 Ulm, Germany, firstname.lastname@uni-ulm.de

[2] https://en.wikipedia.org/wiki/Operation_Aurora

doi:10.18420/sicherheit2020_04

Es17; Ja18; Os16; Sp16; WB14]. The core is an access proxy that resides in front of the service and controls all accesses to services. The basic architecture that is also basis for our implementation can be seen in Figure 1. The access control decision itself is taken by a policy engine enforcing policies set by administrators in combination with a trust engine which calculates trust scores of users and devices accessing the network. BeyondCorp relies heavily on the exclusive use of managed devices, a policy that cannot be enforced in University networks where students and staff are heavily relying on their own, private devices. The term Zero Trust does not mean the absence of trust, it means that trust must be earned. A client wanting to access a service must explicitly establish this trust through appropriate means and demonstrate that an access can securely be granted. This *context-based access control* is typically based on a combination of checks such as device certificates, 2-factor authentication, or patch status of the accessing device.

First preliminary implementations of this concept are available, for example by DeCusatis et al. [De16] and Eidle et al. [Ei17]. The latter even describe first concepts of automatic reaction to adjust access policies in case of attacks. What both publications as well as Google's model are missing are documented concepts how trust engines are calculating a trust score based on different inputs. Moreover, none of the publications publish their software implementation.

Based on the issues identified in related work, in this paper, we present the following contributions:

- Faced with many private devices and BYOD policies, we list possible inputs for trust score calculation.
- We propose a subjective logic based concept for the trust engine component.
- We present Alekto[3], an implementation of a Zero Trust network framework.

We focus in particular on the components which are necessary for authentication and authorization and especially how to flexibly take context-based access control decisions and calculate trust scores.

We evaluate the flexibility and feasiblity of Alekto. We use a prominent eLearning system called Moodle that is frequently used in our university as a real-world example. In order to identify potential performance penalties that might occur due to the access proxy, we also conduct a detailed performance analysis. Interestingly enough, these tests show *increased* performance compared to the original system where access control is conducted by the web application itself.

Through our analysis, we could demonstrate that the Zero Trust Model enables the determination of the trustworthiness of individual identities and thus offers valuable new ways to secure a research network.

[3] `https://github.com/vs-uulm/alekto`

2 Concept

In the following, we describe the framework, the trust score calculation, and discuss possible trust score calculation inputs. The framework as described was implemented and published on github. All basic components of the concept were implemented.

2.1 Alekto

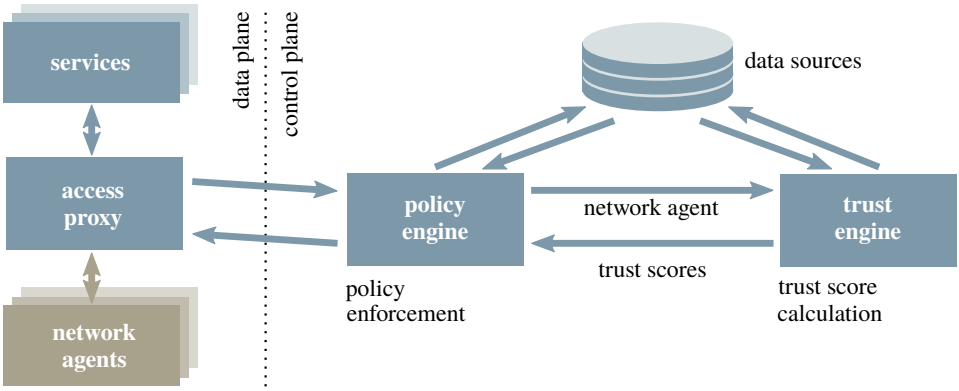


Fig. 1: Architecture of the zero-trust framework.

Figure 1 contains the basic architectural model of Alekto described in the following. It follows Google's *BeyondCorp* terminology and basic setup.

The *network agent* represents the union of a user and a device and contains both the individual trustworthiness of user and device and the trustworthiness of the combination of both. Additionally, user role and category as well as authentication method are important. This way, the security risk can be determined based on capabilities of a user.

Different *data sources* provide information about the network agent. As many independent data sources as possible should be used to mitigate the effects of compromised individual databases or sensors. An authenticatable identity of each network participant is necessary to link the data sources. This allows the storage of uniquely assignable information of a network agent. Basically, a distinction is made between two data source types and associated data base types. Inventory data bases are used to store the current state while historical data bases store past activities of a resource or entity.

The *Services* are all university or research network services that require access control. One example of such a service are the Moodle instances of the universities.

The main task of the *access proxy* is to ensure the global availability of a service. For this purpose it is located in the data plane between client and service and manages the direct network flow between both parties. The proxy should be able to listen via different open

ports to the different protocols required for its downstream service. Access to the services are handled by a single sign-on service located on the proxy.

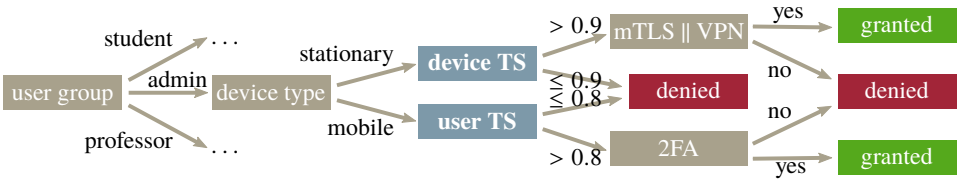


Fig. 2: Example decision tree of the policy engine.

While the access proxy in this system assumes the role of the executive in the sense of separation of powers (i. e. executing the decisions), the *policy engine* represents the judiciary (i. e. judging the network agent). It receives the requests of the applicants forwarded by the Access Proxy and decides on their access rights based on the applicable guidelines. The policy engine is responsible for collecting all information about the requesting entity required for the evaluation of the guidelines and represents it in a network agent. Policies can be defined in form of yaml files. An example for such a policy yaml file can be seen in the appendix (List 4). While the policy engine is very powerful in and of itself, the decisions it can make are exclusively binary. To facilitate more complex decisions, the policy engine is supported by the *trust engine*. The policy engine forwards its created network agent to the trust engine, from which it then receives the trust score determined based on the agent properties. Figure 2 shows an example decision tree the policy engine could follow. The policy engine can feature different policies for different user groups such as admins, students, or professors. Data sources offer a large variety both of device and user properties such as device type or whether the device uses a VPN, two factor authentication (2FA), or mutual TLS. This data can be used for both the trust score calculation and as singular parameters in the decision tree for binary decisions.

2.2 Trust Score Calculation

The *trust engine* is responsible for calculating the trust scores. It can only be addressed by the policy engine for this purpose. As an example, we determine a trust score based on the authentication attempts of the users. These are checked for signs that may indicate an attack or compromised user accounts and devices. We base our trust score calculation on *Subjective logic* by Audun Jøsang [Jø16]. The trust scores are determined using the *binomial opinion*, which is formed over an entity which can be a network agent, a user, or a device. *Subjective logic* is used to calculate the confidence values. Subjective logic is a probabilistic logic that explicitly considers the uncertainty and trustworthiness of a source. Subjective opinions represent the belief in the substance of a statement under consideration of a possible uncertainty. Be $Z \in \{x, \overline{x}\}$ a binary definition range with binomial random

variable $X \in Z$. A binomial opinion about the truth of value x is the ordered quadruple $\omega_x = (b_x, d_x, u_x, a_x)$, with

$$b_x + d_x + u_x = 1 \tag{1}$$

and the corresponding parameters defined as

$$b_x : \text{Belief in the assumption that } x \text{ is TRUE (i. e. } X = x), \tag{2}$$
$$d_x : \text{Belief in the assumption that } x \text{ is FALSE (i. e. } X = \overline{x}), \tag{3}$$
$$u_x : \text{Uncertainty that represents the absence of evidence,} \tag{4}$$
$$a_x : \text{Base rate, i. e. previous probability of } x \text{ without proof.} \tag{5}$$

For this work, exclusively binomial opinions were considered. Here, a statement can assume exactly two values — harmful or not harmful or trustworthy or not trustworthy. The different opinions were linked by a *cumulative belief fusion* in order to reduce the uncertainty of the individual opinions. A belief fusion is used in this work to fuse different opinions about the trustworthiness of an entity. The cumulative belief fusion describes that increased indications for or against the trustworthiness of an entity are confirmed in their statement and reduce the individual uncertainties. Several independent sources, which point to an offense, can thus increase the belief in this offense. In the case that two binomial opinions A and B show a perfect certainty ($u_x = 0$), their values are fused according to:

$$b_x^{(A \circ B)} = \frac{1}{2} \cdot (b_x^A + b_x^B) \tag{6}$$
$$u_x^{(A \circ B)} = 0 \tag{7}$$
$$a_x^{(A \circ B)} = \frac{1}{2} \cdot (a_x^A + a_x^B) \tag{8}$$

and describe the unweighted average of the two opinions. For the implementation of the trust score calculations, the evaluation of past user authentications was chosen as an example, since statements can be made about both the user and their device.

Log entries show when which user attempted authentication. Since device authentication was performed before user authentication, the attempt can also be assigned to a clearly identifiable device. The logs also describe the type of user authentication — for example LDAP login or two factor authentication. In addition, the *failed* flag can be used to indicate whether the login attempt failed. The IP address and the timestamp provide additional information about when and where the attempt took place. This information can be used to detect brute-force attacks or misuse of a user's access data and thus adjust the trustworthiness of the user and device. Two different kinds of brute-force attacks need to be examined here [HMT06; PS02]: For one an attack against a single user account with a list of possible passwords to determine that user's password. This shows up in logs as many failed login attempts for a specific user. Blocking a user account on the basis of this finding would in

turn allow a denial of service attack. The decisive advantage of the Zero Trust Model here is that it is always possible to differentiate between user and device. If such a behavior is discovered, the trustworthiness of the user can be reduced due to a possible compromise of their account. More essential is that the trustworthiness of the executing device is greatly minimized, as it can be assumed that the device is in the hand of an attacker. Finally, the network agent — i. e. the combination of user and device — must also be marked as untrusted, as this combination is likely to be the attacker. The other type of brute-force attacks is directed against a large number of user accounts simultaneously (bulk guessing), without frequent unsuccessful attempts for an account. This attack can be detected by the enormous number of logon attempts made from one device. If such a behavior is detected, trustworthiness of the executing device is reduced.

2.3 Possible Inputs for Trust Score Calculation

To calculate trust scores, Google relies heavily on parameters we cannot obtain. Patch level of private devices or results of virus scanners on the devices are not accessible. Instead, we need different parameters that help us assess the trustworthiness of network participants. In addition to the aforementioned login attempts, we identified other possible inputs that can be used for the trust score calculation.

For one, there are mechanisms that at first glance make it easy to make a binary decision if a network agent is trustworthy or not, e. g. two factor authentication, client TLS, usage of a trusted VPN service, to just name a few. However, the concrete implementation of these services (e. g. which second factor is used) can lead to different trust levels. A fingerprint scanner, an iris scanner, or face recognition technology common on many modern smartphones can support the second factor.

Depending on the service, access from a large distance can be a sign of a possible compromise; additional assurance of identity (e. g. a second factor) could then in turn be required. With rough estimation of the location based on IP, rudimentary tracking can be implemented to see if movement of the device is reasonable (e. g. if a network agent logs in from New York and from Moscow within a time span that is unreasonable for a flight).

Browser fingerprinting can be used to determine additional features of the network agent. Upathilake et al. [ULM15] analyzed and classified several fingerprinting techniques such as using the HTML5 <canvas> element to render an image and determine pixel for pixel how exactly the element was drawn. Subtle differences in the browser implementations allow us to differentiate between the browsers. This should be handled with caution as the features can be falsified. A device suddenly using a different browser to log in can be a sign of an attack. Cookies can make it easy to detect if the browser was used before to successfully log in. Similarly, the type of device (i. e. mobile or stationary) can give insights into how trustworthy it is.

3 Evaluation

We tested the prototype performance which we describe in the following. Two different test scenarios were created for this purpose. The newly developed components will be tested against the existing system. This means that the classical Moodle instance with LDAP authentication is compared to the new Zero Trust components with a Moodle instance extended by JSON web token (JWT) authentication with the goal to examine to what extent the basic performance is influenced by the intermediate Zero Trust System. The Zero Trust components are then re-tested as a target service using a minimal web application. Here, the performance of the components can be tested independently of Moodle.

3.1 Test Setup

All services used for the tests were set up as Docker[4] containers. To create the same conditions for each run, the containers are restarted before the run starts and all existing volumes are deleted after the run ends. For the Moodle instance an already existing image of Bitnami[5] was used, which was extended by a specially developed JWT authentication plugin. The Moodle web server is started with the help of a Docker-Compose file together with the corresponding MariaDB database and an LDAP server. As dummy service, a minimal HTTPS web service was written in Go, which responds to all requests only with a twenty character string.

The Zero Trust components were each created on the basis of the same base image. This image consists of golang:1.10.4[6] and the specially written *shared lib*, which contains all shared basic functions of the different components. The individual containers were then merged using Docker Compose. As LDAP server, a Docker Image by Römhild[7] was used for the Moodle instance as well as for the Zero Trust components.

Three desktop computers were used for the tests. One computer simulates the client requests, one runs Alekto and one runs the service protected by Alekto. The service machine runs either Moodle, including MariaDB database and an LDAP server for Moodle authentication via LDAP or the simple HTTPS web service. Table 2 in the appendix contains hardware specifications of the service machine and the Alekto machine.

Glances[8] was used as hardware monitoring tool. Current status is logged every two seconds. K6[9] is a JavaScript and Go based load testing tool. As one of few tools it offers HTTPS with client certificates. Unfortunately, only one global certificate can be set here for all

[4] https://www.docker.com/
[5] https://github.com/bitnami/bitnami-docker-moodle
[6] https://hub.docker.com/_/golang
[7] https://github.com/rroemhild/docker-test-openldap
[8] https://glances.readthedocs.io/en/stable/quickstart.html
[9] https://docs.k6.io/docs/welcome

requests. For the performance tests this does not matter, because the duration of a certificate validation is measured here and not the semantic meaningfulness of the requests.

3.2 Test Cases

variables	values test 1	values test 2
(1) performance	request duration, CPU, memory	request duration, CPU, memory
(2) systems	proxy + moodle, moodle (LDAP)	proxy + dummy dervice, dummy dervice
(3) # clients	5, 10, 15	25, 50, 62, 75
(4) client groups	mix	students ; admins ; mix

Tab. 1: Test cases to test the Zero Trust components with the help of a dummy service.

The individual test cases are listed in Table 1. Within these cases, the different client groups are additionally considered. The properties of the different groups can be read in the appendix in Tables 3 and 4. There is a big difference between the groups in device authentication, which only takes place in the *admin* and *mix* groups via mTLS. For standard users (*students*) only two guidelines with a total of five requirements are checked. The calculation of the trust values takes place exclusively for the user, since there is no device identity here. Highly privileged users (*admins*) are also checked for two guidelines, but these contain a total of twelve different requirements for users and devices. In addition, their device trust score is calculated based on four properties related to past authentication processes. The policies and trust value calculations are evaluated, but a negative authorization decision for the test runs is ignored and each request is forwarded to the target service to create equal test conditions.

3.3 Results

Preliminary tests already suggested, that the Moodle service was significantly slower than Alekto. First, the implementation with an intermediate proxy is examined. The Moodle instance reached its limits in terms of CPU and memory usage during the tests. It reached 13.6 requests per second ($\frac{r}{s}$) with an average CPU utilization of 51% ($\sigma = 15.79$) with 5 virtual users. Simultaneously, the proxy computer has a CPU utilization of 18% ($\sigma = 8.17$). As the number of users increases, the difference between the two computers increases. While the CPU utilization of the Zero Trust computer increases only slowly with ten ($\mu = 19.72$, $\sigma = 8.33$) and fifteen ($\mu = 20.29$, $\sigma = 8.21$) users, the service machine reaches a critical state with fifteen users with $27\frac{r}{s}$ and a CPU utilization of 93% ($\mu = 93.45$, $\sigma = 18.09$). This confirms the initial assumption that the Moodle web server is the bottleneck of the system.

In the following, the results of the implemented system are compared to the baseline. The baseline consists of the same Moodle instance, but uses the classic LDAP authentication as used in university operations. The goal of this test was to measure the impact of the

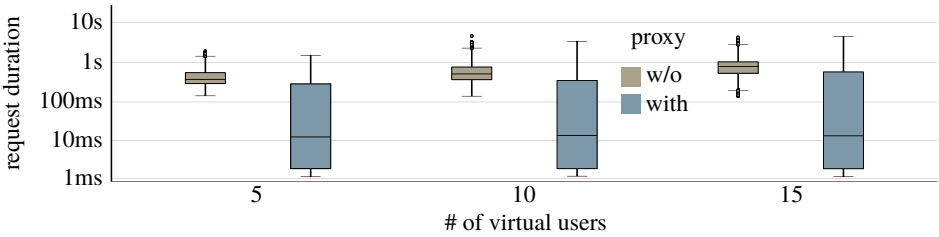


Fig. 3: Diagram of the request duration in logarithmic scale.

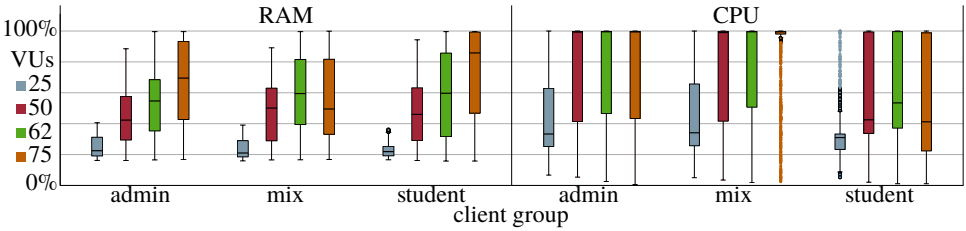


Fig. 4: Resource consumption dependent on user group and number of users.

intermediate proxy. The fact that the requests with proxy and Moodle authentication plugin (as illustrated in Figure 3) are on average shorter (μ = 130.02, σ = 210.00) than those without proxy and with LDAP authentication (μ = 442.58, σ = 230.36) was surprising. Hard to see in logarithmic scale, the response time of the Moodle service without proxy almost doubles from five users (μ = 442.58, σ = 230.37) to fifteen users (μ = 808.05, σ = 354.64).

This result clearly shows that the Zero Trust components have a positive effect on the performance of the web server in terms of response time. This can be attributed to the LDAP authentication outsourced to the single sign-on service which replaced the classic Moodle LDAP authentication. While the CPU utilization of the service machine is the same in both cases (+/- 2 %), it is noticeable that the RAM in the Zero Trust variant is higher. This already starts with five users with 4% (μ = 25.41, σ = 1.06 to μ = 21.08, σ = 0.32) and reaches 8% with fifteen users (μ = 34.196, σ = 7.37 to μ = 26.33, σ = 0.41). Since the only difference here was the authentication plugin used, it can be assumed that this result was caused by the cryptographic validation of the JWT.

Since both the CPU and the memory usage of the Zero Trust components during tests with a Moodle service reached a maximum of 20% (CPU: μ = 20.29, σ = 8.22; RAM: μ = 19.78, σ = 2.02), more detailed tests were performed with a minimal dummy web application running on the service machine instead of Moodle. The limits of the Zero Trust system were firstly explored by using the bisection method. The tests were carried out with 25, 50, 62, and 75 virtual users (VUs).

Figure 4 shows the CPU and memory utilization of the Alekto machine. Clear differences

between the user groups can be seen. This difference most likely results from the fact that students did not send any client certificates in the requests. All in all, the amount of requests per second that were possible ranged from 67 to 148 depending on user group and number of virtual users. Keeping in mind, that the system runs on a regular desktop PC and that the system has proven to be a lot more efficient than Moodle, the evaluation shows that the system scales well for practical applications.

3.4 Discussion

The non-functional performance tests showed that the performance of the baseline is not negatively affected by the use of Alekto. It has even been improved in terms of processed HTTP requests per second and their required response time. As the Moodle web server was the bottleneck of the system, the performance could be optimized by the outsourced LDAP authentication. To make a general statement about the non-functional performance of the Zero Trust components, additional tests against other target services are needed.

4 Conclusion

With regard to the current research on the implementation of the Zero Trust Model, it becomes clear that this model is mainly used in commercial enterprises with the help of proprietary software. Some companies that have successfully implemented the concept provide insights into their fundamental structures; their technologies, however, remain classified as well as the calculation of trust scores. In university networks, many of the factors often used to determine trust cannot be used, such as device patch levels.

In this work, we introduced a model on how to calculate trust scores, listed possible inputs for trust score calculation, and introduced Alekto, a Zero Trust network framework. The evaluation showed the feasibility of our approach to perform its tasks.

In the future, we plan to extend our system, plan to evaluate the system in a wider context closer to a production network, and develop recommended policies for the policy engine as well as recommendations for trust score metrics.

Acknowledgment

This work was supported in the bwNET100G+ project by the Ministry of Science, Research and the Arts Baden- Württemberg (MWK). The authors alone are responsible for the content of this paper.

References

[Be17] Beyer, B. (E.; Beske, C. M.; Peck, J.; Saltonstall, M.: Migrating to BeyondCorp: Maintaining Productivity While Improving Security. Usenix ;login: 42/2, pp. 49–55, 2017.

[De16] DeCusatis, C.; Liengtiraphan, P.; Sager, A.; Pinelli, M.: Implementing Zero Trust Cloud Networks with Transport Access Control and First Packet Authentication. In: 2016 IEEE International Conference on Smart Cloud (SmartCloud). Pp. 5–10, Nov. 2016.

[Ei17] Eidle, D.; Ni, S. Y.; DeCusatis, C.; Sager, A.: Autonomic security for zero trust networks. In: 2017 IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON). Pp. 288–293, Oct. 2017.

[Es17] Escobedo, V. M.; Zyzniewski, F.; Beyer, B. (E.; Saltonstall, M.: BeyondCorp 5: The User Experience. Usenix ;login: 42/3, pp. 38–43, 2017.

[HMT06] Hole, K. J.; Moen, V.; Tjostheim, T.: Case study: Online banking security. IEEE Security & Privacy 4/2, pp. 14–20, 2006.

[Ja18] Janosko, M.; King, H.; Beyer, B. (E.; Saltonstall, M.: BeyondCorp 6: Building a Healthy Fleet. Usenix ;login: 43/3, pp. 24–30, 2018.

[Jø16] Jøsang, A.: Subjective logic. Springer, 2016.

[Os16] Osborn, B.; McWilliams, J.; Beyer, B.; Saltonstall, M.: BeyondCorp: Design to Deployment at Google. Usenix ;login: 41/1, pp. 28–34, 2016.

[PS02] Pinkas, B.; Sander, T.: Securing passwords against dictionary attacks. In: Proceedings of the 9th ACM conference on Computer and communications security. ACM, pp. 161–170, 2002.

[Sp16] Spear, B.; Beyer, B. (E.; Cittadini, L.; Saltonstall, M.: Beyond Corp Part III: The Access Proxy. Usenix ;login: 41/4, pp. 28–32, 2016.

[ULM15] Upathilake, R.; Li, Y.; Matrawy, A.: A classification of web browser fingerprinting techniques. In: 2015 7th International Conference on New Technologies, Mobility and Security (NTMS). Pp. 1–5, July 2015.

[WB14] Ward, R.; Beyer, B.: BeyondCorp: A New Approach to Enterprise Security. Usenix ;login: 39/6, pp. 6–11, 2014.

Appendix

	service machine	Alekto machine
Processor	Intel(R) Core(TM) i5-4590	Intel(R) Core(TM) i5-4590
RAM	DIMM DDR3 8GiB	DIMM DDR3 8GiB
HDD	250GB	500GB
OS	Ubuntu 18.04.2 LTS	Ubuntu 18.04.2 LTS

Tab. 2: Hardware properties of the test environment.

Authentication	Students	Admins	Mix (Students, Admins)
user	LDAP	LDAP	LDAP
device	-	mTLS	mTLS

Tab. 3: Type of user and device authentication by client group. In the *mix* group, each request is authenticated using mTLS.

Authentication	Students	Admins	Mix (Students, Admins)
# policy variables	2	2	2
# policy requirements	5	12	5-12
# trust device variables	0	1	1
# trust device opinions	0	4	4
# trust user variables	1	1	1
# trust user opinions	3	3	3

Tab. 4: Number of required evaluations of the policy and trust engine by client group.

```
#global policy for admin users
- kind: policy
  metadata:
    name: global-admin
    scope: global
    path:
    desciption:
       Highly priviliged users of a service may only access this service
            with the highest security levels
  subjects:
  - kind: userRole
    name: admin
  requires:
    user:
      trustscore: 0.9
      authentication:
        values: [mfa, 2fa]
```

```
    device:
      trustscore: 1.0
      authentication:
        values: [ipsec, mtls]
        operator: and
      type:
        values: [mobile]
        not: true
      networkagent:
        trustscore: 1.0

#global policy for mobile devices
- kind: policy
  metadata:
    name: global-mobile-device
    scope: global
    path:
    desciption:
      Mobile users may never access services with high privileged user
          rolse
  subjects:
  - kind: deviceType
    name: mobile
  requires:
    user:
      role:
        values: [admin, secretary]
        not: true

#global policy for student users
- kind: policy
  metadata:
    name: global-student
    path:
    scope: global
    description:
       Students only need basic authentication, mtls, and an acceptable
           trust score
    subjects:
    - kind: userRole
      name: student
    requires:
      user:
```

```
      trustscore: 0.8
    device:
      authentication:
        values: [mtls]
```

List. 1: Example policy configuration file.

Schwachstellen und Angriffsketten in der Wertschöpfungskette der Fleischproduktion

Eine Analyse an exemplarischen IT-Infrastrukturen

Manfred Hofmeier[1] und Ulrike Lechner[2]

Abstract: Im Rahmen des Deutsch-Österreichischen Forschungsprojekts NutriSafe (Sicherheit in der Lebensmittelproduktion und -logistik durch die Distributed-Ledger-Technologie) wurde eine Analyse zur Identifikation von relevanten Schwachstellen und Angriffsketten in Wertschöpfungsketten von Lebensmitteln durchgeführt. Die vorliegende Arbeit beschreibt die Analyse der IT-Infrastrukturen von fiktiven, aber der Realität nachempfundenen Akteuren der Wertschöpfungskette, basierend auf einem der Szenarien des NutriSafe-Projekts. Ergebnisse sind Beziehungen von Gefährdungen zu IT-Infrastrukturen und Modelle von Angriffsketten in Form von Attack Trees.

Keywords: Lebensmittelsicherheit, Cybersicherheit, Schwachstellen, Attack Trees

1 Einleitung

Im durch das Bundesministerium für Bildung und Forschung (BMBF) und das österreichische Bundesministerium für Verkehr, Innovation und Technik (BMVIT) geförderten Forschungsprojekt NutriSafe (Sicherheit in der Lebensmittelproduktion und -logistik durch die Distributed-Ledger-Technologie) forschen Universitäten, Unternehmen und Behörden daran, die Lebensmittelproduktion sowie deren Logistik unter Nutzung von Distributed-Ledger-Technologie (DLT) sicherer zu machen [Re19][Nu19].

In der Auswertung des Bundesamts für Sicherheit in der Informationstechnik (BSI) über die Häufigkeit von Cybervorfällen in Kritischen Infrastrukturen für das Jahr 2018 ist der Sektor Ernährung derjenige mit den wenigsten IT-Sicherheitsmeldungen [Bu18]. Das zur Meldung verpflichtende IT-Sicherheitsgesetz [Ge15] betrifft Unternehmen, die mehr als 434 500 Tonnen bzw. mehr als 350 Mio. Liter im Jahr produzieren oder verarbeiten [Bu16]. Der Sektor Ernährung ist jedoch von kleinen und mittleren Betrieben sowie Verfahren der kontinuierlichen Belieferung geprägt [Pl05]. Die Idee von NutriSafe soll vor allem die KMUs in Lebensmittelproduktion und -logistik ansprechen. Wenig ist über diesen Sektor aus Sicht der IT-Sicherheit bekannt.

[1] Universität der Bundeswehr München, Fakultät für Informatik, Institut für Schutz und Zuverlässigkeit, Werner-Heisenberg-Weg 39, 85577 Neubiberg, manfred.hofmeier@unibw.de
[2] Universität der Bundeswehr München, Fakultät für Informatik, Institut für Schutz und Zuverlässigkeit, Werner-Heisenberg-Weg 39, 85577 Neubiberg, ulrike.lechner@unibw.de

doi:10.18420/sicherheit2020_05

Im Rahmen von NutriSafe wurde eine Analyse zur Identifikation von Schwachstellen und Angriffsketten durchgeführt. Der Analyse liegen dabei zwei Szenarien [Wi19] und die dazugehörigen IT-Infrastrukturen [Ho19] zugrunde, die mittels Desk Research und Experteninterviews erstellt wurden. Das vorliegende Papier beschreibt die an die Szenarioentwicklung anknüpfende Analyse der Schwachstellen und Angriffsvektoren.

2 Vorgehensweise

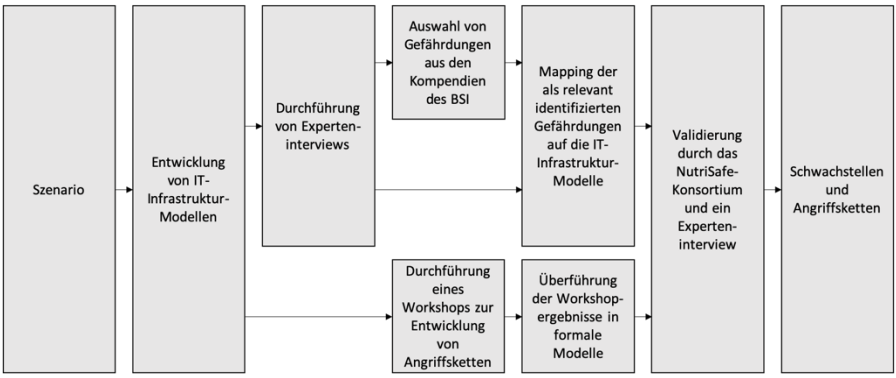


Abb. 1 Vorgehensweise der Schwachstellenanalyse im Projekt NutriSafe

Auf Basis des im Projekt NutriSafe entwickelten Szenarios wurden die IT-Infrastrukturen mittels Desk Research und informeller Notation modelliert [SW06] und im Anschluss durch die Experten im Projektkonsortium validiert (mehr zur Zusammensetzung des Konsortiums in [Nu19]). Diese Infrastrukturmodelle stellen die Grundlage der anschließenden Analyse dar.

Die Kompendien des BSI, speziell der Bereich „IND.1“ des IT-Grundschutz-Kompendiums [Bu19] und das ICS-Security-Kompendium [Bu13], wurden hinsichtlich Gefährdungen ausgewertet und Experteninterviews mit der Bundesvereinigung der Deutschen Ernährungsindustrie (BVE) und dem CERT@VDE wurden im Juli/August 2019 durchgeführt. Der Fokus auf die industriellen Steuersysteme und die Auswahl relevanter Gefährdungen aus den Kompendien fußt u.a. auf den Interviewergebnissen. Dabei stellte sich heraus, dass es eine große Schnittmenge von Gefährdungen in den Kompendien mit in den Interviews genannten Gefährdungen gibt. Die ausgewählten Gefährdungen wurden dann mit den Infrastrukturmodellen abgeglichen, um Schwachstellen zu identifizieren. Parallel dazu fand im August 2019 ein projektinterner Workshop zur Entwicklung von Angriffsketten statt. Im Workshop wurde zunächst eine kurze Einführung in eine einfache Form der Notation mit Attack Trees [Sc99] gegeben, gefolgt durch die kreative Entwicklung von Angriffsketten und Aufzeichnung mittels Attack Trees durch die einzelnen Teilnehmer. Im Anschluss an die kreative Phase folgte eine Gruppendiskussion, in der die einzelnen Angriffsketten in Bezug auf Plausibilität und

Realismus diskutiert und ggfs. verfeinert wurden. Im Nachgang wurden die Angriffsketten entsprechend bereinigt, in formale Modelle (Attack Trees [Sc99]) überführt und durch die Workshopteilnehmer einem Review unterzogen. Zuletzt wurden sowohl die identifizierten Schwachstellen als auch die entwickelten Angriffsketten in einem Interview mit einem IT-Sicherheitsexperten validiert.

3 Ergebnisse

3.1 Informationen aus den Interviews

Das gemeinnützige CERT@VDE des Verbands der Elektrotechnik Elektronik Informationstechnik e.V. (VDE) unterstützt Unternehmen bei IT-Sicherheitsvorfällen im Bereich der industriellen Automatisierungstechnik. Die Bundesvereinigung der Deutschen Ernährungsindustrie (BVE) ist ein Interessensverband der Unternehmen und Fachverbände und verantwortlich für den branchenspezifischen Sicherheitsstandard für die Ernährungsindustrie (B3S Ernährungsindustrie). Im Rahmen der Schwachstellenanalyse wurden mit beiden Organisationen telefonische Interviews durchgeführt. Wesentliche Ergebnisse beziehen sich auf die Relevanz des Themas:

- Es gibt IT-Sicherheitsvorfälle in Lebensmittelproduktion und -logistik, darunter öffentlich bekannt gewordene Beispiele von Ransomware-Vorfällen bei Logistikern. Dabei kann davon ausgegangen werden, dass derartige Vorfälle erst dann öffentlich bekannt gegeben werden, wenn sie bereits an die Öffentlichkeit gedrungen sind.
- Es ist mit der vollen Bandbreite an Angreifertypen mit unterschiedlichen Motiven zu rechnen. Neben Erpressung gibt es vor allem zwei Motive für potentielle Angreifer: Sabotage und (Industrie-)Spionage. So kann man davon ausgehen, dass der Typ des Cyberterroristen sich primär auf die industriellen Steuersysteme (ICS) fokussieren würde, um die Verfügbarkeit und Integrität der Waren anzugreifen. Dabei wäre mit einem komplexen Angriff zu rechnen, bei dem eine Vielfalt an unterschiedlichen Angriffsvektoren, wie etwa (Spear-)Phishing, zum Einsatz kommt. Bei kriminellen Akteuren steht – neben Erpressung mittels Ransomware – eher die Ausleitung von Daten zum Zweck der Industriespionage im Vordergrund. Es ist dabei aber nicht auszuschließen, dass staatliche Akteure eine solche Datenausleitung in Auftrag geben, da für staatliche Akteure oft ein Interesse an strategischen Planungen von fremdstaatlichen Konzernen besteht, besonders wenn es dabei um Planungen zum Erschließen von Ressourcen (z.B. Wasser) geht. Besonders hohe Motivation ist vermutlich bei sog. Hacktivisten vorhanden, beispielsweise in Bezug auf Großunternehmen, die wegen ihrer Beanspruchung von kritischen Ressourcen unter Kritik stehen. Aber auch in Bezug auf die Fleischindustrie ist mit Motivation zu rechnen. Innentäter stellen wie in anderen Branchen ein besonderes Risiko dar, da sie besonderen Zugriff auf Informationen und Systeme haben.

- Insgesamt ist die Prozessautomatisierung ein elementares Rückgrat der Lebensmittelwertschöpfungskette und daher ein besonders schützenswertes Angriffsziel. Dabei ist zu beachten, dass dabei vor allem Zulieferer oder Dienstleister des eigentlich anvisierten Unternehmens (z.B. Anlagenbauer oder Subunternehmer; teils kleine Ingenieurbüros) eine Schwachstelle darstellen. Wichtig für die Sicherheit ist eine Betrachtung der Herstellungskette und Teilbestandteile von eingesetzten Produkten (z.B. Produktionsanlagen). Weitere Themen sind die sichere Ausgestaltung der Fernwartung von Produktionsanlagen sowie Zutritts- und Zugriffsbeschränkungen, wobei auch physische Barrieren eine große Rolle spielen.

Aufgrund der Interviewergebnisse wurde bei der Auswertung der Kompendien der Schwerpunkt auf Produktion und industrielle Steuersysteme (ICS) gelegt.

3.2 Gefährdungen und Schwachstellen

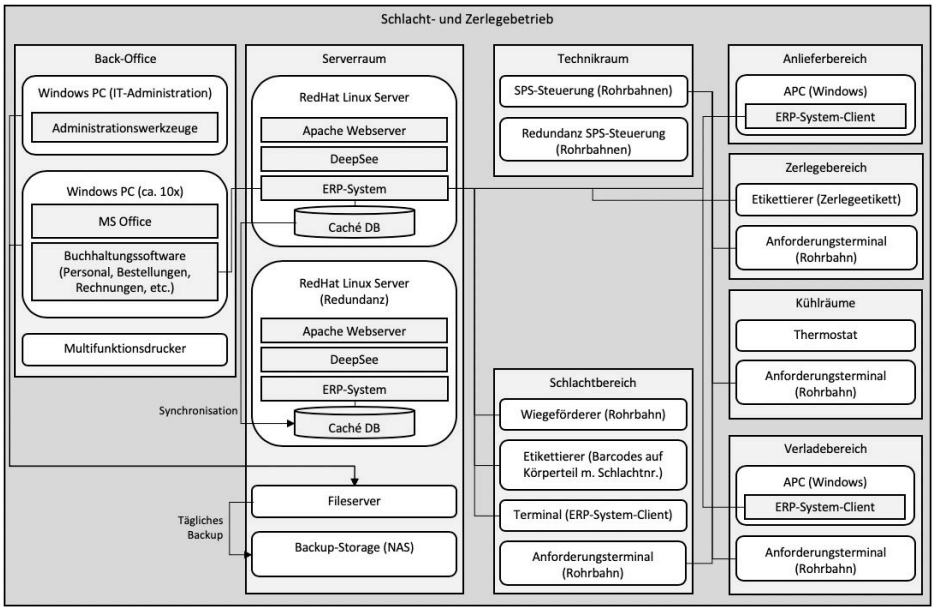


Abb. 2 Beispielhafte IT-Infrastruktur eines Schlacht- und Zerlegebetriebes [Ho19]

Im Folgenden werden die Bezüge der Gefährdungen zu den IT-Infrastrukturen zweier Akteure der Wertschöpfungskette für Kochschinken – Schlacht- und Zerlegebetrieb und Fleischproduktion/Metzgerei – beschrieben. Das Mapping wurde unter Zuhilfenahme von Literatur [Bu13][Bu18][Bu19][Dä18][DC19][Eu19][Fo16][Pl05][Ur13][Wy19], Interviews sowie der Expertise im Projektkonsortium (Zusammensetzung des Konsortiums in [Nu19]) durchgeführt.

Die IT-Infrastruktur des Schlacht- und Zerlegebetriebes (Abb. 2) gliedert sich im Groben in die Bereiche Back-Office, Serverraum und Produktionsbereich. Der Produktionsbereich wiederum ist gegliedert in die verschiedenen physischen Bereiche Anlieferbereich, Schlachtbereich, Zerlegebereich, Kühlräume, Verladebereich und den Technikraum, der die SPS-Steuerung für die Rohrbahnen enthält, mit der die Fleischhaken für den Transport von Tierhälften gesteuert werden. Einige Bereiche in der Produktion enthalten Anforderungsterminals für diese Rohrbahn. Rohrbahn und ERP-System bilden die zwei wichtigsten Bestandteile der technischen Infrastruktur in diesem Betrieb. Das ERP-System und der Produktionsbereich sind durch Netzwerksegmentation vom Back-Office getrennt.

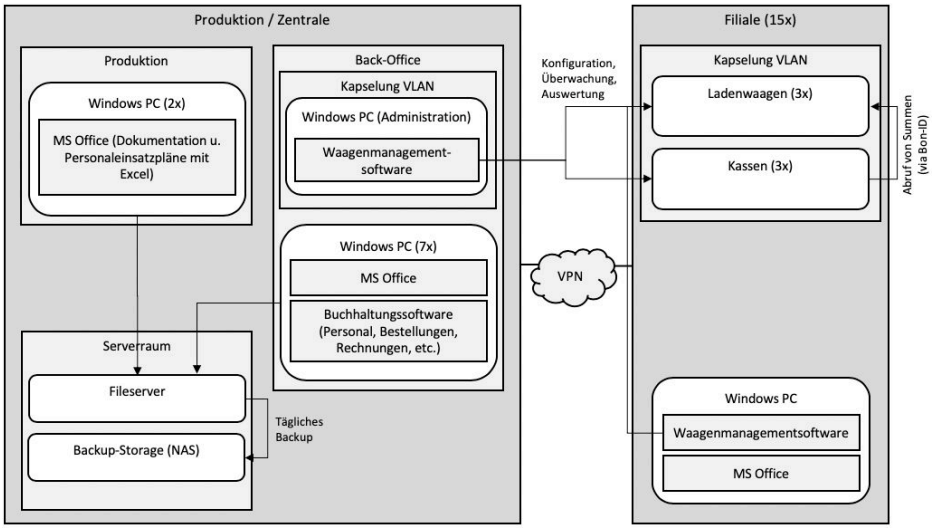


Abb. 3 Beispielhafte IT-Infrastruktur für Fleischproduktion / Metzgerei [Ho19]

Die IT-Infrastruktur der Fleischproduktion im Kochschinken-Szenario von NutriSafe (Abb. 3) gliedert sich im Wesentlichen in die Zentrale mit Produktion und Back-Office und in die 15 Filialen, verbunden durch ein Virtual Private Network (VPN). Elementarer Bestandteil sind die Ladenwaagen und Kassen, die zentral über eine entsprechende Software verwaltet werden (z.B. Einstellen von Produkten und Preisen, Auswerten von Verkäufen). Die Waagen und Kassen sind dabei in ein eigenes VLAN gekapselt. Die Produktion geschieht zwar maschinell (z.B. Kutter), jedoch sind diese in der Regel nicht vernetzt, warum diese auch nicht in der IT-Infrastruktur abgebildet sind. Lediglich die Dokumentation wird IT-gestützt durchgeführt. Dazu wird parallel zum laufenden Betrieb auf Papier dokumentiert und im Nachgang werden die Daten in MS Excel überführt.

Gefährdung	Relevanz für die IT-Infrastruktur Schlacht- und Zerlegebetrieb	Relevanz für die IT-Infrastruktur Fleischproduktion
Organisatorische Gefährdungen (basierend auf dem ICS-Security-Kompendium [Bu13])		
Unzureichende Regelungen zur IT-Security	Es sind nur rudimentäre Regelungen vorhanden, die nur bei der Anstellung neuer Mitarbeiter kommuniziert werden	Es sind nur rudimentäre Regelungen vorhanden, die nur bei der Anstellung neuer Mitarbeiter kommuniziert werden
Unzureichende Dokumentation	Es ist zwar eine Dokumentation der Systeme im segmentierten Netz vorhanden, aber deren genaue Konfiguration sowie die Office-Computer und deren Konfiguration sind i.d.R. nicht dokumentiert	Es ist zwar eine Dokumentation der Waagen und Kassen im segmentierten Netz vorhanden, aber die Office-Computer und deren Konfiguration sind i.d.R. nicht dokumentiert
Unvollständige Absicherung der Fernwartungszugänge	Die Fernwartung der Rohrbahn sowie der Server und PCs ist nicht auf bestimmte, spezifisch freigegebene Zeiträume beschränkt	
Fehlende Überwachung der unterstützenden Infrastruktur	Es gibt keine Überwachung innerhalb des Netzwerks (z.B. keine netzwerkbasierten Intrusion-Detection-Systeme)	Es gibt keine Überwachung innerhalb des Netzwerks (z.B. keine netzwerkbasierten Intrusion-Detection-Systeme)
Mangelnde Awareness		Aufgrund der primär manuellen Tätigkeit spielt IT-Sicherheit für die meisten Mitarbeiter keine große Rolle; Das Personal in der Buchhaltung ist ein wenig sensibilisiert, bringt aber nur rudimentäre IT-Kompetenzen mit
Menschliche Fehlhandlungen (basierend auf dem ICS-Security-Kompendium [Bu13])		
Unzureichende Absicherung oder zu weitreichende Vernetzung	Alle Produktionsanalagen (ERP-Server und Clients, Rohrbahn) befinden sich im selben VLAN	
Unzureichende Validierung von Eingaben und Ausgaben	Dokumentation von Wareneingang, Warenausgang und Verarbeitung erfolgt händisch mit Excel	
Weitere Gefährdungen (basierend auf Interviews und IND.1 IT-Grundschutz [Bu19])		
Unzureichender Zugangsschutz		Es gibt keine reglementierten Zutrittsbeschränkungen für Mitarbeiter
Unzureichender Zugriffsschutz		Der PC für das Waagenmana–gement ist mit einem generischen Account mit leicht zu merkendem Passwort gesichert
Unzureichende Redundanzen (z.B. Single Points of Failure)	Für die SPS-Steuerung ist zwar eine Redundanz vorhanden, aber im selben Raum; Für das ERP-Serversystem ist zwar eine Redundanz vorhanden, aber im selben Raum	Alle wichtigen Daten liegen auf einem Fileserver ohne Redundanz (nur Backup vorhanden); Einige Produktionsanlagen sind aufgrund des hohen Kaufpreises nur einmal vorhanden

Tab. 1 Mapping der Gefährdungen auf die IT-Infrastrukturen (Auszug)

In Bezug auf die Infrastruktur des Schlacht- und Zerlegebetriebes ergeben sich folgende Schwachstellen: Sowohl die SPS-Steuerung als auch da ERP-System haben zwar Redundanzen, jedoch befinden sich diese jeweils im selben Raum. So ist zwar Ausfallsicherheit gegenüber den meisten einfachen Defekten oder Angriffen gegeben, jedoch wären bei physischen Bedrohungen wie Brand oder Wassereinbruch beide Einheiten betroffen. Bei dem ERP-System, welches das Herz in Bezug auf die Daten darstellt und u.a. für die innerbetriebliche Rückverfolgbarkeit von Fleisch wichtig ist, würde auf Dokumentation in Papierform zurückgegriffen werden, wodurch die Produktivität stark verringert werden würde. Bei einem Ausfall der Rohrbahn würde der Transport von Tieren und Tierhälften manuell erfolgen, wodurch ebenfalls die Produktivität stark eingeschränkt würde. Zudem gibt es in der Infrastruktur in diesem Szenario kein versioniertes Backup für das ERP-System. So ist zwar eine Redundanz gegeben, aber Schutz vor Datenmanipulation und dafür notwendige Rollback-Option bestehen so nicht.

Bei der Fleischproduktion/Metzgerei bestehen neben geringer Awareness für IT-Sicherheit, bedingt durch die primär manuelle und analog maschinelle Technik, vor Allem Schwachstellen wie mangelhaft gesicherte Zugänge zu den einzelnen Computern, darunter die Computer für das Waagen- und Kassenmanagement. Zusätzlich fehlen physische Zutrittsregelungen und -beschränkungen innerhalb der Produktionsanlagen und der Filialen. Aufgrund der Natur der Betriebe ist das aber unvermeidlich, wodurch ein grundlegendes Vertrauen in die Mitarbeiter und eine entsprechende Sorgfalt bei der Anstellung neuer Mitarbeiter nötig sind.

Die Schwachstellen geben zwar einen Überblick über Problembereiche, für sich genommen geben sie aber nur eine grobe Sicht auf die IT-Sicherheit. Daher wurden auch mögliche Angriffspfade mit Täterprofilen und Intentionen modelliert, um exemplarisch zu beleuchten, welche Vorfälle bzw. Angriffe existieren können, um besonders kritische und problematische Punkte sowie Ziele von Angreifern sichtbar zu machen.

3.3 Angriffsketten

In einem Workshop unter Verwendung von Kreativtechniken wurden Angriffsketten für diese Infrastruktur mittels Attack Trees [Sc99] entwickelt, anschließend auf Plausibilität und Realitätsnähe diskutiert, entsprechend angepasst und dann bereinigt ausmodelliert. Insgesamt wurden auf diese Weise fünf verschiedene Attack Trees zu den IT-Infrastrukturen der NutriSafe-Szenarien entwickelt, die verschiedene Angriffsstrategien unterschiedlicher Angreifertypen beschreiben. Nachfolgend werden exemplarisch zwei Beispiele für Attack Trees zu den oben beschriebenen IT-Infrastrukturen dargestellt.

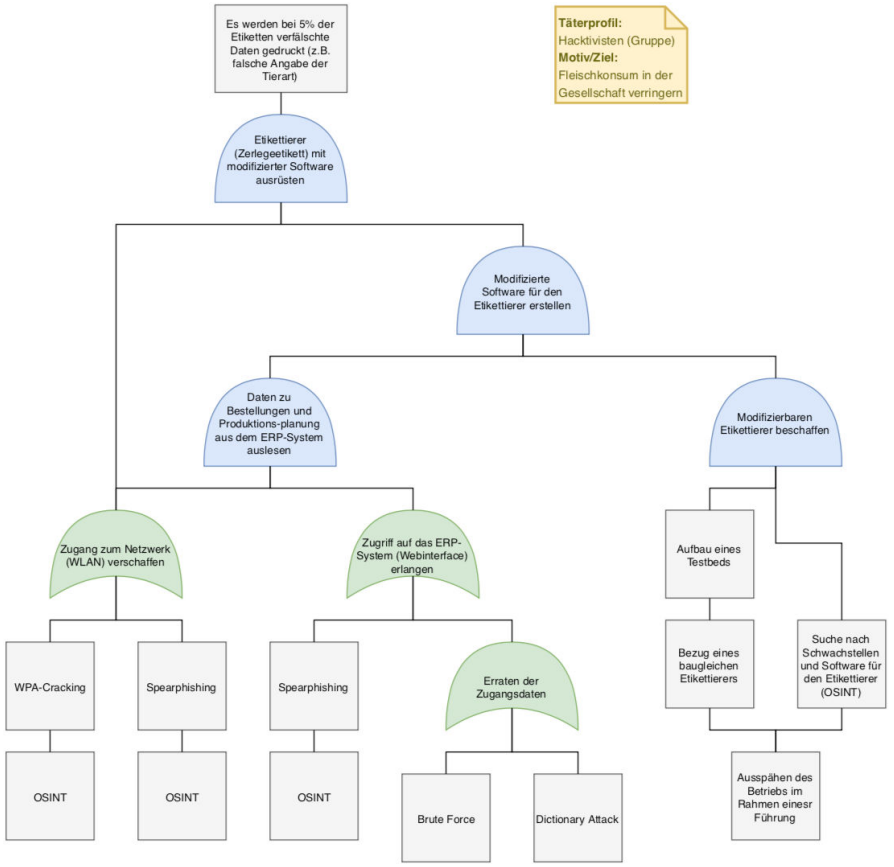


Abb. 4 Attack Tree für einen Hacktivistenangriff auf einen Schlacht- und Zerlegebetrieb

Der Attack Tree in Abb. 4 beschreibt einen komplexen (fiktiven) Angriff durch eine Hacktivistengruppe, die durch gezielte Fehletikettierung der Produkte das Vertrauen in Fleischprodukte verringern möchte. Der Angriff soll so gestaltet sein, dass nur ein kleiner Prozentsatz der Etiketten verfälscht wird, so dass die Fehletikettierung möglichst nicht auffällt und diese Produkte in den Handel geraten. Der Angriff erfordert dabei viel Aufwand und physischen Zugang zum Betrieb.

Ein weiterer Attack Tree (Abb. 5) beschreibt einen klassischen Angriff durch einen Innentäter in einer Metzgereifiliale. Hierbei möchte ein Auszubildender, der in Streit mit dem Filialleiter ist und erwartet, gekündigt zu werden, der Filiale schaden. Das Motiv ist dabei Rache und Ziel des Angriffs ist ein Defacement der Kassenzettel. Da es sich um einen Innentäter handelt, ist der Zugang zu den entsprechenden Räumen und Systemen relativ leicht zu erlangen.

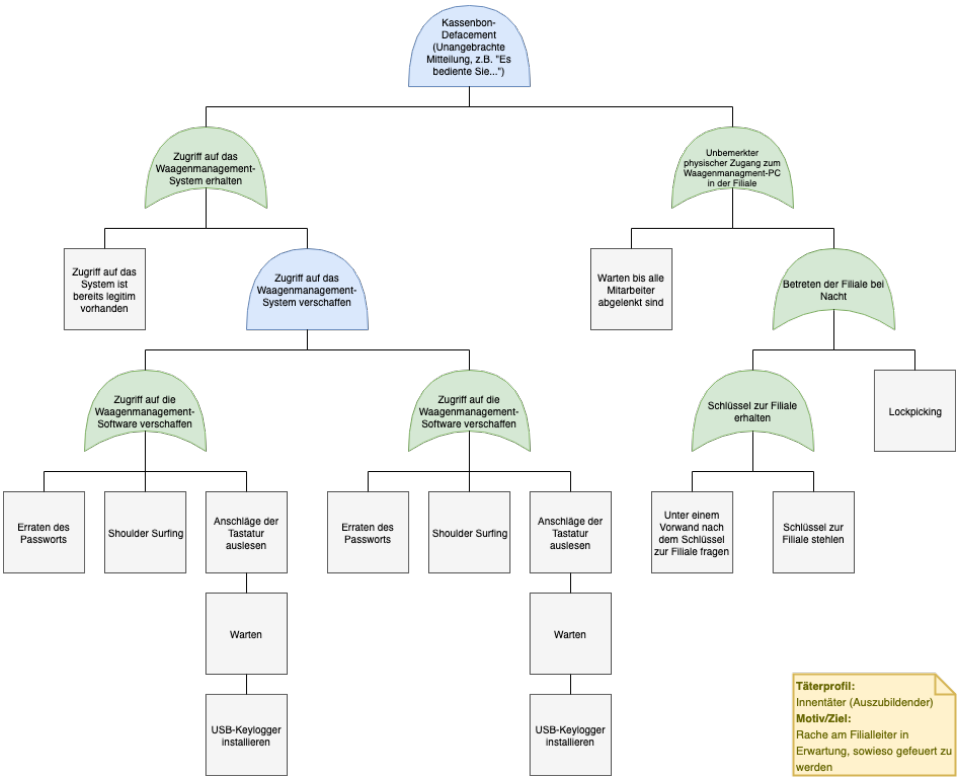


Abb. 5 Attack Tree für einen Angriff durch einen Innentäter in einer Metzgereifiliale

4 Limitationen und Diskussion

Diese Arbeit leistet einen Beitrag, die IT-Sicherheit bei kleinen und mittleren Unternehmen zu verstehen. Es gibt dazu wenige verwertbare Informationen und diese Information ist auch nicht leicht zu beschaffen. Wir verwenden hierzu drei Analyseperspektiven: Die Interviews mit Experten ergeben, dass mehr passiert als in der Öffentlichkeit bekannt ist, also dass das Thema der IT-Sicherheit relevant ist, und dass vor allem bei den industriellen Steuersystemen (ICS) Schwachstellen kritisch sind. Diese Systeme sind aus Sicherheitsperspektive für produzierende Betriebe und Kritische Infrastrukturen besonders schützenswert [RL18]. ICS-Schwachstellen werden sonst eher bei großen industriellen Betrieben thematisiert. Die Szenarioanalyse mit Hilfe der Kompendien des BSI zeigt die Vielfalt der Schwachstellen, selbst bei einem kleinen Betrieb in der Lebensmittelproduktion. Mit dem kreativen Ansatz lassen sich auch spezifische Bedrohungsszenarien für einen Betrieb und seine Kunden und damit die Gesellschaft aufzeigen. So konnten auch sektorenspezifische Angriffspunkte identifiziert

werden, wie etwa Etikettendrucker und Waagenmanagementsysteme. Der dreistufige Prozess über Interviews, Analyse der Referenzwerke und Kreativmethoden stellt dabei die Qualität der gewonnenen Daten sicher.

Die Analyse basiert auf Szenarien, die im Forschungsprojekt NutriSafe entwickelt wurden. Sie sind fiktiv aber realitätsnah und somit sind die Ergebnisse der Schwachstellenanalyse ebenfalls als realitätsnah einzuschätzen und wurden auch durch ein Interview mit einem IT-Sicherheitsexperten validiert. Da die Industriestruktur in der Fleischwirtschaft aber heterogen ist, sind die Ergebnisse nicht ohne weiteres auf die gesamte Branche verallgemeinerbar.

Angriffe auf Betriebe in der Lebensmittelproduktion und -logistik sind nur selten dokumentiert. Vorfälle werden nur selten gemeldet oder veröffentlicht. Die Ergebnisse aus dieser Arbeit schließen diese Lücke. Solche Beispiele sind nicht nur interessant für Forschung und Lehre, sondern auch für die Praxis hilfreich zur Orientierung [Ve18].

5 Acknowledgements

Wir danken dem Bundesministerium für Bildung und Forschung (BMBF) für die Möglichkeit der Forschung im Rahmen des Projektes NutriSafe (FKZ 13N15070 bis 13N15076) sowie dem Sicherheitsforschungsförderprogramm KIRAS, finanziert vom Bundesministerium für Verkehr, Innovation und Technologie (Projektnummer: 867015).

Literaturverzeichnis

[Bu13] Bundesamt für Sicherheit in der Informationstechnik: ICS-Security-Kompendium. Bonn, 2013.

[Bu16] Bundesministerium des Innern: Verordnung zur Bestimmung Kritischer Infrastrukturen nach dem BSI-Gesetz (BSI-KritisV). 2016.

[Bu18] Bundesamt für Sicherheit in der Informationstechnik: Die Lage der IT-Sicherheit in Deutschland 2018. Bonn, 2018.

[Bu19] Bundesamt für Sicherheit in der Informationstechnik: IND.1 Betriebs- und Steuerungstechnik. In: (Bundesamt für Sicherheit in der Informationstechnik): IT-Grundschutz-Kompendium. Bonn, 2019.

[Dä18] Dännart, S: IT-Sicherheit in der Molkerei: Familientradition und Hochverfügbarkeit. In (Lechner, U; Dännart, S; Rieb, A; Rudel, S): CASE|KRITIS - Fallstudien zur IT-Sicherheit in Kritischen Infrastrukturen. Berlin, 2918.

[DC19] DuHadway, S; Carnovale, S: Malicious Supply Chain Risk: A Literature Review and Future Directions. In: Revisiting Supply Chain Risk. 2019.

[Eu19] European Union Agency for Cybersecurity: ENISA Threat Landscape Report 2018 - 15 Top Cyberthreats and Trends. 2019.

[Fo16] Fontanazza, M: FBI Says Terrorists May Target Food Sector. In: FoodSafetyTech. https://foodsafetytech.com/news_article/fbi-says-terrorists-may-target-food-sector, zuletzt abgerufen am 09.01.2020. 2016.

[Ge15] Gesetz zur Erhöhung der Sicherheit informationstechnischer Systeme (IT-Sicherheitsgesetz). In: Bundesgesetzblatt Jahrgang 2015 Teil I Nr. 31. 2015.

[Ho19] Hofmeier, M: Beispiele für IT-Infrastrukturen in den Wertschöpfungsketten der NutriSafe-Szenarien. https://nutrisafe.de/veroeffentlichungen. Neubiberg, 2019.

[Nu19] NutriSafe – Sicherheit in der Lebensmittelproduktion und -logistik durch die Distributed-Ledger-Technologie, https://nutrisafe.de, zuletzt abgerufen am 01.10.2019.

[Pl05] Platz, U: Vulnerabilität von Logistikstrukturen im Lebensmittelhandel. Landwirtschaftsverlag, Münster-Hiltrup, 2005.

[Re19] Reimers, T. et al.: Absicherung von Wertschöpfungsketten in der Lebensmittelproduktion und -logistik mittels Distributed-Ledger-Technologie – Das Forschungsdesign. In: Tagungsband zum 16. Deutschen IT-Sicherheitskongress. Bonn, 2019.

[RL18] Rudel, S; Lechner, U: IT-Sicherheit für Kritische Infrastrukturen – State of the Art. https://www.itskritis.de/state-of-the-art.html. Neubiberg, 2018.

[Sc99] Schneier, B: Attack Trees. 1999.

[SW06] Schubert, P.; Wölfle, R: The Experience Methodology for Writing IS Case Studies. In: Americas Conference on Information Systems (AMCIS). 2006.

[Ur13] Urciuoli, L; Männistö, T; Hintsa, J; Tamanna, K: Supply Chain Cyber Security – Potential Threats. In: Information & Security, Volume 29, Issue 1. 2013.

[Ve18] VeSiKi: Monitor 2.0 IT-Sicherheit Kritischer Infrastrukturen. https://itskritis.de/monitor. Neubiberg, 2018.

[Wi19] Wilhelmi, T: Kurzbeschreibung der NutriSafe-Szenarien Produktion und Logistik von Bio-Kochschinken und Weichkäse. https://nutrisafe.de/veroeffentlichungen. Neubiberg, 2019.

[Wy19] von Wyl, H: Metzger Wechsler und die Hacker. In: Compass Security. https://www.compass-security.com/fileadmin/Datein/News/2019/MOB_01_EPIC_LOW_RES_PDF-pages-4-6.pdf, zuletzt abgerufen am 09.01.2020. 2019.

D. Reinhardt, H. Langweg, B.C. Witt, M. Fischer et al. (Hrsg.): Sicherheit 2020,
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Göttingen 2020 79

CryptoCAN – Ensuring Confidentiality in Controller Area Networks for Agriculture

Till Zimmermann[1], Jan Bauer[2], Nils Aschenbruck[3]

Abstract: The Controller Area Network (CAN) bus is widely used in existing machinery. Facing more and more vertical integration with more complex devices and integration into public communication networks, its nature as a broadcast-only system without security measures poses serious risks to confidentiality of transmitted data. In this paper, we propose a *Lightweight, Length Preserving and Robust Confidentiality Solution (LLPR-CS)* to retrofit encryption in existing systems, while maintaining full interoperability with these systems. The overhead of our approach is negligible. Therefore, it can be used with existing hardware. By reinterpreting unused bits in the CAN frame format of the ISO 11898 standard, it is possible to build a fully transparent encrypted tunnel in non-confidential network parts, while keeping the ability to decrypt all traffic in an out-of-band-system without knowledge of specific cryptographic state details. By conducting a performance evaluation, we highlight the benefits of *LLPR-CS* and discuss its advantages compared to existing approaches.

Keywords: ISO 11783; ISOBUS; Controller Area Network; Security; Privacy; Confidentiality; Smart Farming

1 Introduction

The CAN system is extensively used in many kinds of modern automotive and industrial control systems. Due to its wide adoption and versatile applications, there are many reasons to use this system even though achievable data rates are heavily limited. From a perspective of application developers, one of the main reasons is – beside the good reliability – the simplicity of CANs, which allows to implement all networking functions on the application layer. This advantage lies in the fact that CANs form a simple broadcast-only topology, so that no routing or expensive network coordination is required. However, this topology has important implications for security in such systems. While the security of traditional, air-gapped systems could basically rely on the physical inaccessibility of the network, for current CANs, it is no longer possible to solely rely on this implicit security measure, since the industrial demand for the vertical integration into the Internet of Things (IoT) is very common in many recent developments in areas like the Industry 4.0 or Smart Farming.

[1] University of Osnabrück, Institute of Computer Science, Wachsbleiche 27, 49076 Osnabrück, Germany tzimmermann@uos.de
[2] University of Osnabrück, Institute of Computer Science, Wachsbleiche 27, 49076 Osnabrück, Germany bauer@uos.de
[3] University of Osnabrück, Institute of Computer Science, Wachsbleiche 27, 49076 Osnabrück, Germany aschenbruck@uos.de

doi:10.18420/sicherheit2020_06

Whereas the most obvious security issue, the inability to verify the integrity and authenticity of received messages, was discussed in multiple papers [HKD08; KY17; Wa17], the confidentiality of message delivery still remains challenging in real-world scenarios. To this end, we propose a simple and lightweight approach to seamlessly retrofit encryption into existing systems. This approach preserves the length information of original payload and, thus, increases the interoperability, particularly with higher-layer protocols. Moreover, a suitable block cipher encryption is adapted that offers secure and robust confidentiality. The paper is organized as follows. Our motivational scenario and background details are given in Sec. 2, followed by a brief discussion on related work (Sec.3). Then, in Sec. 4, four different cryptographic variants are introduced including *LLPR-CS* and related approaches. Finally, the performance evaluation is presented in Sec. 5 and Sec. 6 concludes the paper.

2 Background

In this paper, the ISOBUS is exemplarily used. This protocol interconnects agricultural machinery to provide high-resolution control, e.g., of seeding and fertilization. Often heterogenous actors are involved, regarding both technical (machinery) and organizational (contractors) aspects. In agricultural practice, on the one hand, it is common to rely on contractors for specific work on the field. These contractors can either use their own or customers' machines. On the other hand, due to the high cost and specialization one may rent so-called *implements*, which may be trailers, planters or sprayers by third party actors.

The agricultural environment with different actors has neglected privacy aspects for long periods. However, due to the digitalization and networking using farm management information systems (FMISs) that are often cloud-based, these aspect become highly relevant in todays agriculture.

As the efficiency of Precision Agriculture depends on the availability of high-resolution data, it's necessary to interconnect all machinery from different actors, such as farmers, contractors or even rented implements from third parties. Therefore, all actors, who may also be competitors, can record, interpret, and (ab)use collected data that could reveal business information, e.g., soil or yield properties of individual fields, or personal data of employees.

2.1 Controller Area Network (CAN) in Precision Agriculture

CAN (ISO 11898) [ISO15; Ok05] is a simple and robust broadcast-only bus system using differential signaling. There exist different generations of the CAN protocol. The version CAN 2.0A has a frame format consisting of an 11 bit identifier, up to 8 bytes for payload, and several internal fields, such as Cyclic Redundancy Check (CRC) and Data Length Code (DLC). The identifier is used as an arbitration field allowing to avoid collisions, using a physically supported listen-while-talk multiple access scheme, i.e., Carrier Sense Multiple

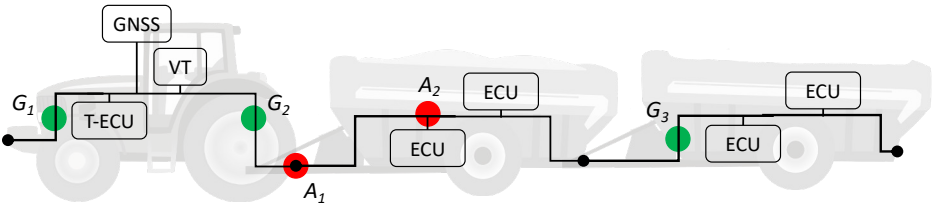


Fig. 1: ISOBUS network with encrypting gateways G and potential points of attacks A (GNSS: global navigation satellite system receiver).

Access/Collision Resolution (CSMA/CR). CAN 2.0B enlarges the identifier field to 29 bit, allowing a total of 93 bits user-defined data to be transferred in a frame.

ISOBUS (ISO 11783) [ISO07] is a protocol based on CAN that additionally defines a full application protocol. It is specified for CAN 2.0B with a fixed data rate of 250 kbit/s. ISOBUS applications reuse the identifier field of CAN frames to transfer message headers by splitting it into multiple fields. Some parts of the message header are used to allow for backwards compatibility with existing protocols and also for message prioritization (based on the arbitration field of the original CAN). The most notable part is the so-called Parameter Group Number (PGN), which, in conjunction with a publicly available database of PGNs [VMDA], enables a bit-wise interpretation of the payload. This allows flexible applications, such as the configuration of a Virtual Terminal (VT) in the tractor's cabin to be shared as central GUI for all implements connected to the tractors' ISOBUS (cf. Fig. 1).

2.2 Scenario & Security Threats

The motivating scenario which was already sketched above is shown in Fig. 1. Possible access points for malicious data collection are ISOBUS connectors between machines and implements that are exemplarily represented by A_1 and A_2. Due to the broadcast medium, a potential attacker could unnoticeably add a logging device to those points with minimum effort. By adding gateways encrypting all private messages ($G_{1,2}$) and its counterpart (G_3) inside trusted implements, these points are no longer useful for malicious actors. This architecture, however, assumes that all units connecting sensors and actors to the bus (called Electronic Control Unit (ECU) in automotive terms) inside both, the tractor as well as the (trusted) second implement, are under control of the data owners. Despite of being practically relevant, this exemplary agricultural scenario is not the only one highly requiring confidentiality. The encryption of certain data prior to the cloud upload is also conceivable, for instance. Although recent industrial standardization (e.g., [AUTO15]) and best-practices put more focus on authorization in automotive networks, confidentiality is still not taken into account.

3 Related Work

Over the recent years, high effort has been put into researching the security aspects of CANs. However, most of these works were focused on message integrity and authenticity, which are – in terms of vehicle networks – mainly necessary to ensure operational safety. Both is vital to protect against physical harms of drivers or the vehicle itself, yet it does not fit well in scenarios in which the direct, physical harm is not the goal of an attacker. Hoppe et al. summarized this considerations in the commonly used terms of cybersecurity aspects (cf. the CIA rule [NIST95]). They declare the reason for the inability to expect integrity and authenticity as the lack of sender identification, in combination with the shared-medium, broadcast nature of CANs [HKD08]. The aspect of data availability is additionally endangered due to the internal priority mechanisms, which could allow an attacker to continuously send high-priority messages preventing any useful communication on the bus. Since the safety issues introduced by missing authentication and integrity mechanisms are practically relevant, Wu et al. [Wu16] present an approach that uses Hashed Message Authentication Codes (HMACs) in CAN. HMACs necessarily require an inherent overhead (additional data to be transmitted), which cannot be realized in conventional CAN messages with a maximum payload size of 8 bytes. Therefore, the approach relies on payload compression in order to achieve sufficient room for HMAC overhead in the payload.

The confidentiality requirement is addressed by Bruton [Br14], who proposes different encryption schemes. To use widely adopted encryption algorithms, especially the Advanced Encryption Standard (AES), it is necessary to implement an adequate segmentation of the encrypted payload, as encrypted AES blocks are 128 bit in length and, thus, do not fit into a single CAN frame. However, segmentation will inevitably impose a significantly increased bus load and drastically reduce the goodput. Hence, RC4 stream cipher is demonstrated as an alternative [Br14]. Beside RC4 is no longer recommended, stream ciphers are generally prone to desynchronization, because even an offset of a single byte in the key stream renders all transferred data to be broken. Thus, they need an additional synchronization mechanism. Moreover, data enciphered with stream ciphers cannot be deciphered afterwards without storing the key stream position and other state information, such as the used initialization vector (IV). In a short demonstration, Jukl and Čupera [JČ16] present a similar approach to encrypt ISOBUS messages using the Tiny Encryption Algorithm (TEA). This algorithm features a 64 bit block size and, therefore, does not need a segmentation when being applied to CANs. Their work is mainly focused on the ability to execute the encryption on a commercially available microcontroller platform. Furthermore, solely ISOBUS communication is considered. This makes their approach very specific and not applicable to pure CAN systems of other domains. Overall, in the current literature, there is a lack of approaches for a feasible confidentiality solutions that can be applied to existing CANs with minimal modifications of existing systems in terms of hardware, software, and bus load.

4 Adding Confidentiality to CANs

4.1 System Requirements

In order to meet different requirements, it is our goal that an encrypted communication should be usable in two architectures: The most obvious solution is to insert stand-alone gateways G_i as proxies into an existing network to realize encrypted tunnels as shown in Fig. 1, where the whole section between G_2 and G_3 could be encrypted, preventing malicious attackers from reading data either at the physically accessible port A_1 or an integrated ECU (A_2). This may especially be effective in environments where clearly defined network sections in the topology exists, in which data should be encrypted, but existing devices cannot be updated or replaced. For a clear modularization, in our approach, each encryption scheme features two software components, one for encryption and one for decryption and therefore can be easily embedded in existing ECUs. In this paper, the encryption key is expected to be present on both the en- and decryption side. The actual key exchange is out of scope but could be realized by corresponding key exchange protocols.

4.2 Cryptographic Variants

Because there are varying system requirements for different use cases, multiple encryption variants were developed and implemented in this work. These include block and stream cipher schemes. Even though all described schemes can generally be used in CANs, every variant has its own limitations, advantages, and disadvantages, which are summarized in Tab. 1 and discussed in the following.

4.2.1 Block Cipher

***Bruton* (AES)** To verify the usability and to have a homogenous measuring environment, the proposed solution from [Br14] using AES encryption was implemented in our software framework. The payload of each received CAN frame is encrypted with a pre-shared key using the Cipher Block Chaining (CBC) operation mode, which is strictly necessary to prevent attacks based on differential cryptanalysis. As the payload only consists of maximum

Variant	Encryption	Length preservation	Network overhead	Fault resistance	Practical applicability	Cryptogr. security
Bruton	AES	X	–	–	+	+
Inline-ISOBUS	XTEA	X	○	+	–	○
LLPR-CS	XTEA	✓	○	+	+	○
Stream Cipher	ChaCha20	✓	+	–	○	+

Tab. 1: Advantages (+) and drawbacks (–) of different encryption variants.

64 bits and AES block size is always 128 bit, the additional bits are used to encode the length of the received payload. This additional information is necessary to send the decrypted message with its original length and without padding as upper layer protocols may rely on the message's length as part of their protocol. Unused bits are simply set to zero, as the knowledge of partial bits in a single block does not have any security implications. Once encrypted, the resulting block is split into two message segments that are transmitted using the original identifier. The receiving side simply decides how to handle even and odd packets, and resembles both segments. After decrypting such a resembled frame, it is sent to the unencrypted bus segment or unmodified software expecting unencrypted data. Obviously, one can safely rely on *cryptographic security* of this method, since AES is considered as state-of-the-art and used in nearly every modern encryption system. However, this variant is expected to highly affect the *bus load*, as every unencrypted frame needs to be sent in two different frames (with maximum payload size) containing a part of the ciphertext.

Inline-ISOBUS To prevent excessive overhead and to prevent degradation of fault tolerance as present in the AES-method, it is possible to choose a 64 bit block cipher [JČ16]. The use of TEA is one possibility, while the underlaying block cipher can be chosen freely depending on known attacks against any of them, and there exist multiple alternatives such as Blowfish or 3DES. By using TEA as demonstrated in [JČ16] in CBC mode, the design is fairly simply, requiring only knowledge of the previous block and an IV or – given that single incorrect transmission do not present a problem for the application – even without this. The biggest advantage of the CBC mode is its self-synchronizing functionality, i.e., there is no counter involved and the decryption for block n only relies on the ciphertext of block $n-1$. Basically, it is not recommended to use 64 bit block ciphers as common attack vectors – so-called *birthday attacks* [BL16] – exist, especially when used in conjunction with the CBC mode. Nonetheless, these vulnerabilities are unlikely to present a problem in CAN-based systems, as the data rate is strictly limited and therefore it is nearly impossible for an attacker to capture the necessary large amount of cipher text. While this solution does not require any state information to decrypt captured traffic and, therefore, can be used in heterogenous systems as well, it still shows a severe limitation compared to the AES method. Because the encryption is a strictly bijective transformation for the whole payload of 64 bit and cannot be used for partial blocks, it is impossible to encrypt blocks shorter than 64 bit. To encrypt messages smaller than 8 byte, they are padded with arbitrary values to a full 8 byte-block. As traditional padding schemes which allow for the removal of the padding before transferring the payload to the upper layer protocol are built for messages with multiple consecutive blocks, these are not applicable in CANs. Thus, this variant is unable to *preserve the length* of the unencrypted message. Hence, suchlike encryption schemes can only be used if the upper layer protocol ignores additional data, which is the case in ISOBUS, but cannot be necessarily expected for other CAN-based protocols.

LLPR-CS To overcome this limitation of missing length preservation, a few bits for additional metadata are sufficient that simply specify the number of padding bytes used.

The DLC field in the CAN frame header allows for that few extra bits of information to be encoded in a standard-compatible way, without breaking compatibility to existing hardware. As only 9 of the 16 available states representable by 4 bit have a valid meaning, the CAN standard declares all other states to be interpreted as a 8 byte payload [ISO07, Part 4]. Therefore, the unused states can be safely used to indicate the number of padding bytes which have to be removed by the decrypting side. The case of a zero-byte payload cannot be signaled this way, however it is feasible to simply send an empty frame in this case, because no data content is available for encryption.

4.2.2 Stream Cipher

The most relevant challenge for using block ciphers in CANs is the fixed block size, which either leads to the need for fragmentation, or any form of padding with the shown difficulty of discarding the original length. To overcome these problems, it may be promising to use stream ciphers instead. They allow for encryption of any chunk of data with arbitrary lengths, as the encryption consists only of the bitwise addition with an independent key stream. Accordingly, this would not require any padding and, therefore, can preserve the length of transmitted frames. In some situations however it may be useful to record messages sent on the CAN. Especially in agricultural use cases, this is done to allow for offline analysis of collected data. As every message encrypted using a stream cipher needs to be decrypted with the correct portion of the key stream, it would be necessary to store the messages' position in the encryption session. Furthermore, even when this should not happen in simple bus scenarios, a single lost message invalidates the counter for the specific entity. Hence, this variant has a very limited *fault tolerance*, as only reordered messages may happen. Thus, an additional mechanism for synchronizing the counter is necessary.

5 Evaluation

5.1 Implementation

For our proof-of-concept evaluation, a fully-featured single board computer is used in this paper, which allows for a more flexible development system. The introduced encryption variants were prototypically implemented on a Raspberry Pi and were integrated in the open-source *CAN't* [Ba19] framework[1]. All variants were implemented in stand-alone binaries to allow for easy adoption to existing software using SocketCAN. Publicly available and maintained libraries were used whenever available, namely *OpenSSL* for the AES-based variant (*Bruton*), and *libsodium* for the ChaCha20 algorithm (*Stream Cipher*). For the implementation based on TEA, the reference implementation of the corrected and eXtended TEA from the authors was used [WN98].

[1] `https://sys.cs.uos.de/cant`

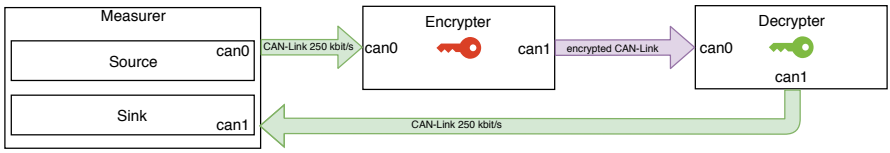


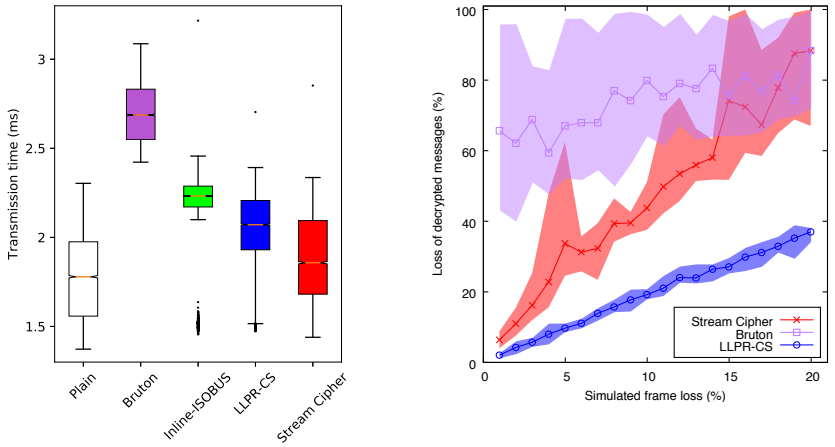
Fig. 2: Evaluation setup.

5.2 Methodology & Evaluation Setup

To achieve a deeper insight on the impact of the different encryption schemes on a real system, we set up a testbed consisting of four logical communication units as shown in Fig. 2. Two of them form a pair of legitimate bus nodes (*Source* and *Sink*), which need to communicate in a confidential manner. The other two units serve as an *Encrypter* and a *Decrypter*, establishing an additional and encrypted bus segment that could be publicly exposed. This setup represents the worst case in terms of delay, because the single bus segment of the original setup (without the encrypted tunnel segment) is divided into three independent CANs. Therefore, the time frames spend on the communication medium is expected to get at least tripled. Because *Source* and *Sink* are integrated in a single physical node (*Measurer*, i.e., a Raspberry Pi with two CAN interfaces), precise delay measurements are possible without potential time synchronization inaccuracies. As described in Sec. 4.2.1, all block ciphers were used in CBC mode.

5.3 Results

The **transmission delay** induced by our method is measured by the duration for the successful transmission of random payloads with uniform distributed, random lengths from *Source* to *Sink*. Even though every application may have specific limits on maximum message delays, the additional delays the encryption adds are fairly low as shown by the boxplots in Fig. 3(a) (10 000 frames for each variant). The most obvious delay factor may be the encryption itself, which was evaluated in an additional measurement. However, as all encryption algorithms are developed to allow fast encryption and decryption of large data, the differences are mostly negligible compared to the differences resulting from larger or additional data transfers. Therefore, and because there are many publicly available benchmarks for cryptographic algorithms, these results are left out here. As expected, the *Bruton* variant that splits one frame into multiple frames is obviously the slowest one. It increases the mean delay from 1.77 ms to 2.69 ms. According to [ISO07], the encrypting gateway is to be handled as a network interconnection unit (NIU). The standard recommends a maximum processing time of 10 ms, which is found to be uncritical for all variants. Nonetheless, the novel *LLPR-CS* variant enables a significant lower delay than *Bruton*'s variant and, due to its length-preserving property, it allows to remove padding bytes after decryption which leads to a lower transmission delay than with *Inline-ISOBUS*. Thus, *LLPR-CS* is feasible in delay-critical systems while preserving the advantages of block encryption schemes over stream ciphers.



(a) Transmission delay (*Source* to *Sink*). (b) Loss sensitivity (min, max, mean for each variant).

Fig. 3: Testbed evaluation results of different encryption schemes.

In many scenarios, it may be vital to keep the **bus load** low, because of CANs limited throughput. Similar to the transmission delay, we also investigated the additional bus load induced by each variant, but do not present details here due to space limitations. However, expected results were found. It is obvious that *Bruton*'s AES solution is only suitable in environments with a prior load below 50% capacity since the fragmentation doubles the load. Whereas the stream cipher variant has no impact on bus load, the overhead of both variants, *LLPR-CS* and *Inline-ISOBUS*, depends directly on the typical length of transmitted messages, as their overhead is exactly the padding required to form 8-byte-blocks. Because *LLPR-CS* is able to remove the padding before the second transmission, it has a lower delay, but also a higher variance due to the random message sizes in our evaluation.

So far, a fully reliable CAN was used. This may, of course, not always apply. Thus, it is necessary to consider the effects on transfer **robustness** of all variants. Even though CAN is definitely focused on preventing frame loss and itself guarantees an in-order transmission of messages, this assumption may be invalidated as higher priority messages prevent the transmission of lower priority ones in nearly saturated networks. Additionally, in very noisy environments or near-fault conditions, this expectation may no longer hold true. This is especially relevant when any kind of gateway is part of a network, as they may reorder messages during the forwarding process. By developing a simple loss emulation tool, it was possible to measure the impact of CAN frame loss with a specified probability. All measurements were carried out with 1000 messages and 10 replications. In Fig. 3(b), the results concerning the impact of frame loss are shown for all variants (except *Inline-ISOBUS* that is identical to *LLPR-CS* here). The *loss of decrypted messages* is the percentage of messages, which either are never delivered or received with a corrupted payload. For both variants of 64 bit-block ciphers, the resulting loss of correct messages behaves proportional to the loss of underlying CAN messages. This matches the expectations. If the payload of a

certain frame is lost, the next frame cannot be correctly decrypted. However, the second following frame can be decrypted correctly. The reason for the factor not being exactly two is the fact that when n consecutive frames get lost in a burst, only $n + 1$ messages are lost or incorrectly decrypted. To evaluate the stream-cipher solution, it was necessary to extend the system by a minimal synchronization method. Without this, all messages transmitted after the first loss occurred would be considered lost.

Overall, the evaluation showed a wide variety of advantages and disadvantages, both in terms of performance characteristics as well as provided features and requirements of the implemented variants. The selection of one of these variants, therefore, relies on a deep knowledge of the characteristics for an existing system. Based on differentiating between them, it is possible to give rough suggestion which variant may be useful. Whenever it is possible to save additional metadata like algorithm's state and the reliability of the CAN is ensured, using a stream cipher can be recommended, as it has only minimal overhead due to its fast processing time. If these prerequisites are not met, the upper layer protocol has to be considered. Specially in ISOBUS-networks, any 64 bit block cipher can be used directly. However, it is important to limit the number of messages sent with a single key to avoid problems when using the CBC mode.

To use block ciphers for generic upper layer protocol, our proposed LLPR-CS allows to do this in a backward-compatible manner without violating the CAN standard by reusing the DLC-bits to mark the padding's length. A small drawback of this solution is the handling of frames received by unmodified gateways whose may discard the additional bits used to indicate the padding. Nonetheless, this solution is lightweight, as it requires no extra storage of state information and robustness by the underlaying block cipher. Additionally, with our solution it is possible to establish a completely transparent encrypted channel, as it preserves the length of transmitted frames.

6 Conclusion

In this paper, we propose *LLPR-CS*, an efficient and interoperable confidentiality solution that leverages unused bits in the header of CAN frames. Although our work is focused on ISOBUS of agricultural machines, the presented solution is applicable to general CAN systems beyond Smart Farming. With prototypical implementations, we compared our novel solution against related approaches. The performance evaluation shows the benefits of *LLPR-CS* with regard to the additional transmission delay and frame loss sensitivity. However, due to space limitations, both, the low impact on additional bus load and its robustness against frame disordering, could not been presented here. Moreover, we also implemented a feature that allows to selectively encrypt ISOBUS messaged according to their PGN, which was also not explicitly addressed in this paper. In our future work, we plan to evaluate our solution on real agricultural machines. Here, the feature of selective encryption is expected to become more crucial to meet the confidentiality requirements, particularly from a privacy perspective, without interrupting the operation of the machinery.

Acknowledgments

This work was partially funded by the German Federal Ministry of Education and Research (BMBF) within the Program "Innovations for Tomorrow's Production, Services, and Work" (02K14A194) and managed by the Project Management Agency Karlsruhe (PTKA). The authors are responsible for the contents of this publication.

References

[AUTO15] AUTOSAR: Specification of Module Secure Onboard Communication, Standard Release 4.2.2, July 2015.

[Ba19] Bauer, J.; Helmke, R.; Bothe, A.; Aschenbruck, N.: CAN't track us: Adaptable Privacy for ISOBUS Controller Area Networks. Computer Standards & Interfaces 66/103344:1–13, DOI: 10.1016/j.csi.2019.04.003, 2019.

[BL16] Bhargavan, K.; Leurent, G.: On the Practical (In-)Security of 64-bit Block Ciphers: Collision Attacks on HTTP over TLS and OpenVPN. In: Proc. of the Conf. on Computer and Communications Security (SIGSAC). CCS '16, DOI: 10.1145/2976749.2978423, Vienna, Austria, pp. 456–467, 2016.

[Br14] Bruton, J. A.: Securing CAN Bus Communication: An Analysis of Cryptographic Approaches, MA thesis, National University of Ireland, Galway, Aug. 2014.

[HKD08] Hoppe, T.; Kiltz, S.; Dittmann, J.: Security Threats to Automotive CAN Networks – Practical Examples and Selected Short-Term Countermeasures. In (Harrison, M. D.; Sujan, M.-A., eds.): Computer Safety, Reliability, and Security. Springer, Berlin, Heidelberg, pp. 235–248, Sept. 2008, ISBN: 978-3-540-87698-4.

[ISO07] International Organization for Standardization (ISO): Tractors and machinery for agriculture and forestry – Serial control and communications data network – Parts 1–14, ISO 11783-{1–14}:2007–17.

[ISO15] International Organization for Standardization (ISO): Road vehicles - Controller area network (CAN) – Part 1: Data link layer and physical signalling, ISO 11898-1:2015, 2015.

[JČ16] Jukl, M.; Čupera, J.: Using of tiny encryption algorithm in CAN-Bus communication. In: Research in Agricultural Engineering. Vol. 62, DOI: 10.17221/12/2015-RAE, pp. 50–55, June 2016.

[KY17] King, Z.; Yu, S.: Investigating and securing communications in the Controller Area Network (CAN). In: Proc. of the Int. Conf. on Computing, Networking and Communications (ICNC). DOI: 10.1109/ICCNC.2017.7876236, Santa Clara, CA, USA, pp. 814–818, 2017.

[NIST95] National Institute of Standards and Technology: An Introduction to Computer Security: The NIST Handbook. Special Publication 800-12, DOI: 10.6028/NIST.SP.800-12r1, Oct. 1995.

[Ok05] Oksanen, T.; Öhman, M.; Miettinen, M.; Visala, A.: ISO 11783 – Standard and its Implementation. IFAC Proceedings Volumes 38/1, pp. 69–74, 2005.

[VMDA] Agricultural Machinery Association: ISOBUS 11783 Online Data Base – PGNs and SPNs, German Engineering Federation (VMDA), URL: https://www.isobus.net/isobus/pGNAndSPN.

[Wa17] Wang, E.; Xu, W.; Sastry, S.; Liu, S.; Zeng, K.: Hardware Module-Based Message Authentication in Intra-vehicle Networks. In: Proc. of the Int. Conf. on Cyber-Physical Systems (ICCPS). Pittsburgh, PA, USA, pp. 207–216, 2017.

[WN98] Wheeler, D. J.; Needham, R. M.: Correction to xtea, tech. rep., Computer Laboratory - Cambridge University, Oct. 1998, URL: https://www.movable-type.co.uk/scripts/xxtea.pdf, visited on: 09/30/2019.

[Wu16] Wu, Y.; Kim, Y.-J.; Piao, Z.; Chung, J.; Kim, Y.-E.: Security protocol for controller area network using ECANDC compression algorithm. In: Proc. of the Int. Conf. on Signal Processing, Communications and Computing (ICSPCC). DOI: 10.1109/ICSPCC.2016.7753631, Hong Kong, China, pp. 1–4, 2016.

D. Reinhardt, H. Langweg, B.C. Witt, M. Fischer et al. (Hrsg.): Sicherheit 2020,
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Göttingen 2020 91

A Survey on Sender Identification Methodologies for the Controller Area Network

Marcel Kneib[1]

Abstract: The connectivity of modern vehicles is constantly increasing and consequently also the amount of attack vectors. Researchers have shown that it is possible to access internal vehicle communication via wireless connections, allowing the manipulation of safety-critical functions such as brakes and steering. If an Electronic Control Unit (ECU) can be compromised and is connected to the internal bus, attacks on the vehicle can be carried out in particular by impersonating other bus participants. Problematic is that the Controller Area Network (CAN), the most commonly used bus technology for internal vehicle communication, does not provide trustworthy information about the sender. Thus ECUs are not able to recognize whether a received message was sent by an authorized sender. Due to the limited applicability of cryptographic measures for CAN, sender identification methods were presented which can determine the sender of a received message based on physical characteristics. Such approaches can increase the security of internal vehicle networks, for instance to limit manipulations to a single bus segment and thus prevent the propagation of attacks. In this paper different methods are presented, which can mainly be divided into the categories time-based and voltage-based. In this context, challenges as well as open questions are identified and the approaches are compared. Thus, the work offers an introduction, identifies possible research fields and enables a quick evaluation of the existing technologies.

Keywords: Sender Identification; CAN Fingerprinting; CAN Security; Automotive Security

1 Introduction

The automotive industry is massively affected by the ongoing digitalization which leads to an increase in the connectivity of vehicles. However, these innovations not only offer advantages in terms of comfort, but also provide completely new possibilities for attackers to manipulate a vehicle without physical access [ZAA14]. Wireless connections such as Bluetooth, WiFi or cellular radio can be used to access the vehicle's internal network [Lu14]. Software vulnerabilities of connected ECUs allow attackers to manipulate the software accordingly, which, depending on the architecture of the vehicle, allows direct access to the internal communication [HKD11; IV13; Ko10; MV13]. If only limited security measures are implemented, vehicle functions can be remotely manipulated or controlled by a compromised ECU. But even with security measures, such attacks cannot be avoided. The severity became particularly apparent by the attack of Miller and Valasek [MV15], who succeeded in manipulating various vehicle functions of a Jeep Cherokee via a remote connection. In addition to various comfort functions, these included safety-critical systems

[1] Robert Bosch GmbH, Mittler Pfad 9, 70499 Stuttgart, Germany, marcel.kneib@de.bosch.com

doi:10.18420/sicherheit2020_07

such as steering, engine and brake, leading to a recall of 1.4 million vehicles. That this is not a single incident shows the vulnerability recently published by Keen Security Lab [Ca19].

The main problem is that no access protection or verification regarding the authenticity of messages is performed or is only possible to a limited extent. This is due to the functionality of the Controller Area Network (CAN) [Ro91], the most commonly used technology for in-vehicle communication. On the basis of a received CAN message, it is not possible to determine which bus participant sent the message, since only the expected sender can be identified via the information contained in the header [Li17; Pa17]. If a message was sent from a compromised ECU, there is no way to detect the misuse of a message. Due to the low bandwidth and payload [GM13; LS12], as well as the limited computing capacities of the ECUs [GM13; JAC18], the use of cryptographic measures for all signals is only possible with restrictions. Besides, digital signatures would be required to ensure non-repudiation, which have much higher requirements [GM13]. Thus, various Intrusion Detection Systems (IDSs) have been developed in the past in order to identify manipulations or anomalies on the basis of a predefined or learned set of rules [Ax00; Du19; Lo19]. Although these systems can enhance the security, the sender of a message provides an essential additional information which can supplement the overall security concept or allows a reliable recognition of different attacks [HKD11; MGF10]. For this reason, several methods have been introduced that use physical characteristics of the communication to either determine the sender of a message or at least use changes in the characteristics to detect anomalies.

This paper provides an overview of the different types of sender identification methods for CAN, which can essentially be divided into two categories, the time- and voltage-based methods. In addition to the presentation of the methods and corresponding approaches including their evaluation, the survey also includes the assessment of the methods with regard to possible applications, challenges and open questions. Thus, the paper facilitates the familiarization with the technology, provides background knowledge and enables a quick assessment for potentially interested manufacturers.

2 Controller Area Network

Vehicle architectures are becoming more complex as advances in the automotive industry are mainly achieved through electronic components [Br07; St14]. For these architectures, different bus technologies can be used in parallel, whereby nowadays gateways provide an interface between the individual segments and thus offer an ideal point for the integration of sender identification. Although the architectures will change considerably in the future [Br17], it is expected that Ethernet will be used for high transmission rates and CAN, respectively its developments [CA20], will be retained for lower rates [BSG16]. Accordingly, CAN will also be used in future vehicles, including its security weaknesses.

CAN messages are transmitted via frames, which contain a message identifier, representing the meaning and priority of the message and further is used by only one ECU. As the

communication is statically configured, each ECU can receive required messages based on the message identifier. The physical connection consists of a twisted pair cable, one line for CAN high (CANH) and one line for CAN low (CANL), terminated at each end with 120 Ω resistors. If a recessive bit, a logic 1-bit, has to be represented on bus, the voltage of 2.5 V is present on both lines. In case a dominant bit, a logic 0-bit, has to be transmitted, CANH is driven to 3.5 V and CANL to 1.5 V. The actual signal is determined by the differential voltage and thus minimizes potential electromagnetic interference. Since it is possible for several ECUs to access the bus simultaneously, as CAN is a broadcast bus, the arbitration phase ensures that the message with the highest priority is retained. This allows the message with the highest priority to be sent undisturbed, while the ECU that loses arbitration re-queues its message for a later transmission attempt. With regard to sender identification, it should be noted that the physical characteristics of the symbols during the arbitration phase may be distorted.

3 Sender Identification Methodologies

The approaches for sender identification can be divided into two categories. While the time-based approaches use timing differences in CAN communication, the voltage-based approaches analyze differences in the analog signals of received frames. For both methods, models for the observed ECUs are created in an initial learning phase, which are used to analyze subsequent messages for deviations of the expected behavior. If a deviation occurs, the sender of the message can be identified on the basis of the physical parameters.

3.1 Time-based Sender Identification

The approaches in this category use timing differences in CAN communication to identify anomalies or the sender of a present frame.

Clock Skew The Clock-based IDS (CIDS), presented by Cho *et al.* [CS16], exploits the differences in the clocks of the ECUs to detect anomalies within periodically transmitted frames. Each ECU is equipped with its own quartz in order to generate clock frequencies for the microcontroller. One of the applications of these frequencies is the operation of the internal timers, which in turn are responsible for the punctual transmission of periodic messages on the bus. Since the quartzes are subject to natural variations and no synchronization is performed, there are measurable differences between the individual ECUs. By measuring the transmission intervals from an external point, the deviations between the expected and actual periodicity can be observed. On the basis of these deviations, for example, an impersonation attack can be detected.

Specifically, CIDS observes the expected and actual arrival times of each periodically sent message, as illustrated in Figure 1. Here the observed and the expected message flow of two

periodically sending ECUs are shown. The messages are sent in statically defined intervals from which, together with the previous arrival time, the arrival time of the following message can be calculated. The deviation between the measured and the expected value specifies the clock offset O. Whenever a predefined number of clock offsets has been measured, an average clock offset is calculated which is then used to estimate an error value indicating the difference between the expected and the actual clock offset of the corresponding ECU. The expected offset is calculated from the cumulative sum of the previous average clock offsets using the Recursive Least Squares algorithm [Ha05]. An increase of the error indicates a deviation of the clock offset and thus an anomaly can be detected if the error rises above a given threshold value. If the forged messages are sent over a longer period of time, the compromised ECU can be identified using the known average clock offsets of all ECUs. CIDS was evaluated on a CAN prototype assembly and three vehicles, whereby missing and additionally transmitted frames and the takeover of a periodically sent signal by a compromised ECU could be reliably detected.

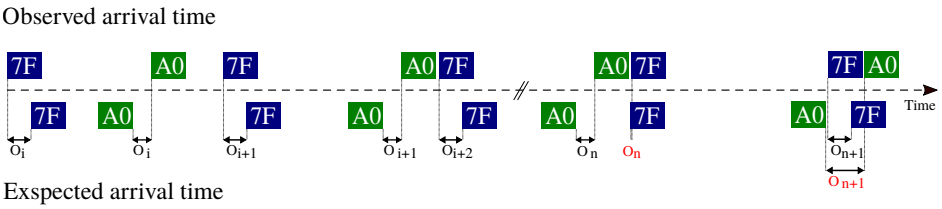


Fig. 1: Expected and observed clock skew from two different ECUs.

Propagation Time Until a transmitted signal arrives at the receiver, a certain period of time expires, which mainly depends on the distance between both nodes. More precise, a signal propagates on the bus medium in the direction of both bus ends, whereby the time of reaching both ends depends on the position of the sender. The TCAN approach of Biham *et al.* [BBG18] is based on these time spans in order to determine the location of an ECU and thus to distinguish between the senders. For this purpose, a measuring unit and a repeater are attached to the opposite ends of the bus. The function of the repeater is to analyze the signals of the messages in order to send an echo to the bus when the first falling edge, i.e. the change from a dominant to a recessive state, is detected after the arbitration phase. However, the CAN communication must not be disturbed by the echo signal and it must be recognizable by the measuring unit, which is also responsible for the determination of the time spans. Therefore, it also has to recognize the first falling edge of the message after the arbitration phase and the echo signal of the repeater. Subsequently, the time interval t_s between these two points is considered for the calculation of the position of the sender. For this purpose, the distance between the measurement unit and the sender of the message is calculated. For the assignment of the ECU to a measured distance, an authentication table is created before operation, for whose initialization the authors specify several options. The process is illustrated in Figure 2 for two different senders. Biham *et al.* [BBG18] also mentions that

repeater and measuring unit could be housed in one device, eliminating the need for the echo signal, which corresponds to the approach of Moreno *et al.* [MF19]. Their approach was additionally evaluated on a vehicle bus with four ECUs. Altogether, the position of the ECUs at a sampling rate of 100 MS/s could be determined with an accuracy between 20 and 30 cm. The distinction was reliable, since there was a sufficiently large distance between the two closest ECUs so that the probability of a misclassification was negligible.

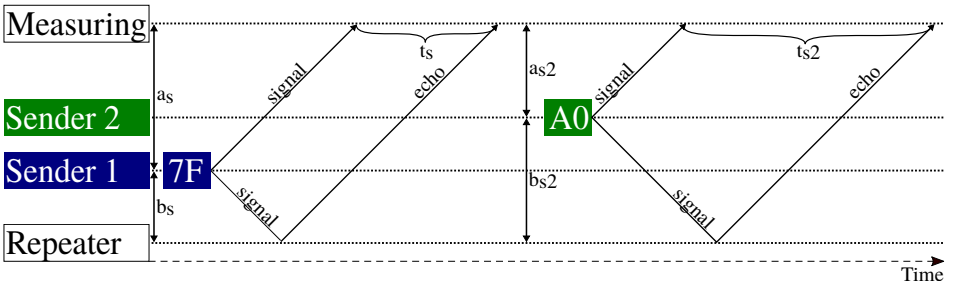


Fig. 2: Propagation delay for two different ECUs using an echo signal.

3.2 Voltage-based Sender Identification

Existing differences in the analog signals, caused by manufacturer-related imperfections of electronic components and the structure of the present CAN topology, can be utilized to identify the sender of received frames. As an example, Figure 3 shows a segment of the signals of six ECUs from a Fiat 500.

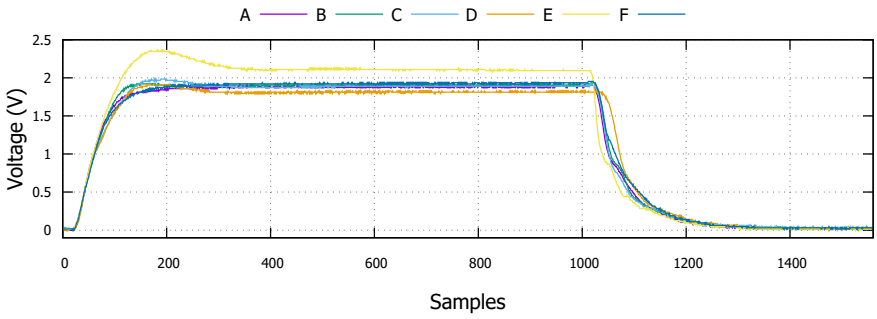


Fig. 3: CAN signals of several ECUs from a Fiat 500.

Comprehensive Signal Characteristics This subcategory considers approaches which are based on the complete or a segment of the analog signal. The idea itself was introduced by Murvay and Groza [MG14] and further developed by Choi *et al.* [Ch18a] by using machine

learning for identification. The two most advanced approaches are VoltageIDS [Ch18b] and Scission [KH18], which consist basically of three phases. In the first phase, the analog CAN signals are recorded and preprocessed. By using the differential signal, potential electromagnetic interferences are compensated, necessary to provide a robust identification. While the sampling rate of VoltageIDS is between 250 and 2500 MS/s, Scission operates with a lower sampling rate of 20 MS/s. In order to provide data-independence, utilizable voltage level changes of the recorded signals are then divided into individual sets. In the second phase, the feature extraction, different characteristics are extracted from the prepared signals, which represent the fingerprint of the sending ECU. Here, the approaches differ with regard to the selected features. Subsequently, the fingerprint is processed in the third phase, the identification and intrusion detection. Here, machine learning algorithms are used in order to determine the sender, resulting in a probability per ECU. The VoltageIDS makes use of Support Vector Machines and Bagged Decision Trees, while Scission uses Logistic Regression for classification. Whereas VoltageIDS marks the ECU with the highest probability as the sender, Scission uses a two-stage threshold procedure to reduce the number of false positives. Only if the probability of the expected sender undercuts a first threshold, the probabilities of the remaining ECUs are calculated. In case one of these probabilities exceeds a second threshold value, the corresponding ECU is labeled as unauthorized sender and the frame is determined to be malicious. The use of comprehensive signal characteristics was analyzed on various prototype structures and four vehicles, whereby Scission shows an identification rate of 99.85 % while all false alarms were prevented.

Voltage Level Compared to the previous category, the approach considered here, Viden [CS17], uses only isolated points of the signal. First, the dominant voltages of CANH and CANL are measured separately with a low sampling rate of 50 kS/s. A voltage instance is then derived from a defined number of measurements, consisting of the most common voltages and various percentiles. Thereby, the deviations of the measured voltages from the expected standardized voltages are calculated and summed up in order to compensate transient deviations as much as possible. This value, which remains approximately constant during operation, is then cumulatively summed up. Using Recursive Least Squares, a voltage profile can be calculated per message identifier, which allows the identification of the sending ECU. Since Viden is designed to complement higher-level IDSs, Viden is used after an attack has been detected in order to identify the attacking ECU. For this purpose, an attacker voltage profile is created for the unauthorized used identifier. As soon as sufficient voltage values have been measured for its calculation, the ECU with the most similar voltage profile is marked as the sender. Since ECUs may have similar profiles, or an attacker may has adjusted his profile to imitate a specific ECU, a further step is taken for identification. For this purpose Viden utilizes a 200 Random Forest classifier, which receives the voltage instance as input. The approach was also evaluated on a prototype structure and two vehicles with an attacker identification of up to 99.8 %.

4 Evaluation

This section assesses the different approaches in terms of purpose, resource needs and unresolved issues. An overview of the assessment can be seen in Table 1.

Tab. 1: Assessment of the sender identification methodologies.

	Maturity level	Hardware Requirements	Computational Requirements	Robustness	Security	Restrictions Architecture	Aperiodic Frames
Clock Skew	+	+	○	○	-	+	-
Propagation Time	-	-	+	+	+	-	+
Signal Characteristics	+	-	-	+	+	+	+
Voltage Level	+	○	○	○	○	+	○

4.1 Time-based Sender Identification

Clock Skew The most significant disadvantage of the procedure is that due to the manner of the clock skew extraction, counterfeiting cannot be completely excluded as demonstrated by the attack of Ying *et al.* [Yi19]. Since the skew can also be observed via a compromised ECU and the transmission frequency can be adjusted via software, the CIDS can be circumvented. Thus the system can only reliably detect missing or additional frames, which can also be achieved by analyzing the transmission schedule [MGF10]. Another disadvantage of the offset extraction is the exclusive consideration of periodic frames as well as a possible shift of frames, shown in the right part of Figure 1, which can lead to fluctuations. However, the software-based approach and especially the methodology is a good demonstration that the clock skew can be used for anomaly detection. Should an alternative source for the extraction of the clock skew be found, the system offers a good and cost-effective way to detect attacks. A method that makes it independent of aperiodic frames would be ideal in order to extend the scope of detection. Furthermore, more detailed investigations are necessary here with regard to changes in the clock skew in fluctuating environmental conditions, such as temperature.

Propagation Time Even though a detailed evaluation of this variant has not yet been carried out, Moreno's [MF19] work indicates that the propagation time is a robust quantity. Especially since the propagation time should be less affected by fluctuations, it might be suitable to falsify malicious frames, however, requiring a much deeper analysis. Since this characteristic is defined by the cable lengths respectively the topology, there is no possibility for a remote attacker to adjust the characteristics of the compromised ECU according to the device to be impersonated. Accordingly, neither complex calculations nor a full update of the time models are necessary, as long as the hardware has not been changed. In addition, the propagation time is not limited to periodically transmitted messages. However, the implementation is not trivial in terms of hardware. For the timely transmission of the echo

signal used by Biham *et al.* [BBG18], the measuring unit and the repeater must reliably recognize and analyze the individual edges within a frame, which exceeds the capabilities of standard CAN devices. Besides, the echo signal must be viewed critically with regard to the CAN specification. Basically, a high clock frequency respectively a high sampling rate is necessary for the accurate measurement of the timings, which represents a high demand on the measuring unit. For example, Biham *et al.* [BBG18] specify a resolution of 1 GHz for an accuracy of 15 cm and 100 MHz for 1.5 m, respectively. If a low frequency is used, only longer distances can be distinguished, which limits the electrical/electronic architecture of the vehicles. However, Moreno *et al.* [MF19] show that even with a low resolution it is possible to make precise predictions. Another limitation is that the location of the implementing devices is restricted. This means that either a device must be placed at both ends of the bus or, if implemented in a single ECU, the already large wiring harness will be further enlarged.

4.2 Voltage-based Sender Identification

Comprehensive Signal Characteristics Although there has been progress in these approaches in the last years and reliable identification has been demonstrated, they still require a considerable amount of resources due to the high sample rates. Accordingly, high demands are placed on the processing ECU with regard to computing performance for the processing of large amounts of data, the calculation of machine learning algorithms and the recording of signals. This is illustrated by the fact that no demonstration has yet been shown on a resource-limited platform. Basically, for a CAN with 500 kB data rate, a payload of 8 bytes and a maximum bus utilization, it can be assumed that the entire classification must be completed in roughly 200 μs in order to be able to react in real time and prevent a buffering of the fingerprints. Although high identification rates have already been achieved in production vehicles, longer investigations are necessary for a reliable assessment. This also includes more concrete information regarding the life cycle of the models, i.e. how to react to possible variations in the signals. In particular, training a classifier is very computationally intensive, which is why it must be shown that this is feasible on the target platform.

Voltage Level Viden is designed for attacker identification only and therefore requires a high-level IDS for the detection of attacks. Only when an attack has been detected, the approach is used to identify the compromised ECU. Due to the methodology and the low sample rate, Viden is designed for periodic messages. Thus, several messages are necessary per iteration, i.e. per creation of a voltage instance. Accordingly, the applicability could be extended to aperiodic messages by increasing the sampling rate to perform the necessary measurements within a single frame. However, just as with the approaches of the previous section, further investigations would be necessary to assess the suitability of Viden as an IDS. At the same time, the assumption made about the equal changes in the voltage levels

among all ECUs should be evaluated, as our measurements show different fluctuations for different ECUs. The deviations of two ECUs during a drive are shown in Figure 4, whereas A has a deviation of 12.2 mV and B of 4.8 mV on average. If it is not possible to simply adjust the voltage signal on the basis of a single value, the model must be regularly retrained, for which the integrity of the vehicle must be ensured. Although the main part of the approach requires few resources, a 200 Random Forest classifier is used for the verification and thus has comparable requirements to the approaches of the previous chapter. In order to benefit from the performance advantage, a solution is required which makes the verification step obsolete.

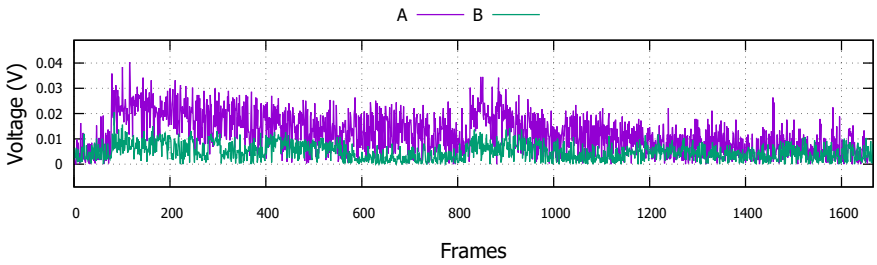


Fig. 4: Plateau voltage fluctuation of two ECUs from a Fiat 500.

5 Comparison and Challenges

Overall, the voltage-based approaches have been evaluated more fully and comprehensively, but further investigations are necessary, particularly regarding robustness. Further vehicles and a longer observation period have to be considered, ideally with different climatic conditions. Even if the evaluations of Scission and Viden show a robustness against fluctuations, the performance of VoltageIDS decreases considerably with temperature changes, unless the models are continuously updated. Considering that the temperature ranges depend on the positioning of ECUs, the performance must be evaluated at least between −40°C and 85°C [Au07]. Furthermore, due to the high production volumes it is essential that especially the methods based on comprehensive signal characteristics are optimized with regard to hardware requirements. Considering available automotive microcontrollers, e.g. STM SPC [ST17] or Infineon TriCore [In19], the implementation of one of these approaches is not realizable due to their high data rates. In terms of reliability, however, these methods offer an advantage over voltage level methods due to the use of numerous characteristics.

In comparison, the time-based procedures require a more extensive evaluation, modification and optimization. The clock skew procedure can be considered broken and therefore requires rework or a new and immutable way to extract clock skews. In addition, an optimization for the monitoring of aperiodic messages would be desirable. Also the temperature dependence

of the clock skew has to be examined in detail, as fluctuations affect the clock rate of the ECUs. In comparison, the propagation time can be assumed to be less fluctuating, since the speed of a signal is assume to vary less. Furthermore, it must be clarified how the positioning of the monitored ECUs, the resolution or sampling rate and potential fluctuations interact in order to be able to configure the system accordingly. Here, questions regarding the implementation on automotive hardware are still open, as the evaluation of the existing approach was carried out offline. However, in view of the current results, the propagation time is also a promising method for implementing sender identification.

Furthermore, it has not yet been clearly elaborated how IDSs should be used in general, i.e. how they can react in the event of a detected intrusion. An interesting possibility for the sender identification systems is that a detection is possible during the actual transmission, which, with a correspondingly fast calculation, allows the invalidation of the message, e.g. by overlaying with the dominant state. Here, extensive research as well as enhancements in robustness and detection rates are necessary, so that wrong reactions are compliant with safety requirements. The reduction of false alarms is particularly important due to the high number of messages, as otherwise, depending on the reactions, the usability of the vehicle could be restricted. Therefore, especially the propagation time approaches and Scission show potential, due to the lack of false alarms.

6 Conclusion

This paper presents the methods for sender identification using physical characteristics for CAN. They provide a reliable detection of impersonation attacks and complement existing IDSs, but still require further research. Open questions have been identified in this paper, representing a starting point for future work. The methods were also compared in order to show the respective advantages and disadvantages, which allows a fast assessment. Generally, the hardware requirements must be further reduced and the robustness must be analyzed in greater detail by extending the evaluation to several vehicles while observing them for a longer duration. In addition, a suitable method for updating the models must be found, especially for the voltage-based methods, as changes of the signal characteristics during the life time of a vehicle are expected. In summary, sender identification offers a considerable benefit with regard to the security of connected vehicles, if the open challenges can be solved.

References

[Au07] Automotive Electronics Council: AEC-Q100 Failure Mechanism Based Stress Test Qualification for Integrated Circuits. 2007.

[Ax00] Axelsson, S.: Intrusion detection systems: A survey and taxonomy, Technical report, 2000.

[BBG18] Biham, E.; Bitan, S.; Gavril, E.: TCAN: Authentication Without Cryptography on a CAN Bus Based on Nodes Location on the Bus. In: Proceedings of the 2018 Workshop on Embedded Security in Cars (ESCAR). 16, Nov. 14, 2018.

[Br07] Broy, M.; Kruger, I. H.; Pretschner, A.; Salzmann, C.: Engineering Automotive Software. Proceedings of the IEEE 95/2, pp. 356–373, Feb. 2007, ISSN: 0018-9219.

[Br17] Brunner, S.; Roder, J.; Kucera, M.; Waas, T.: Automotive E/E-architecture enhancements by usage of ethernet TSN. In: 2017 13th Workshop on Intelligent Solutions in Embedded Systems (WISES). Pp. 9–13, June 2017.

[BSG16] Braun, L.; Sax, E.; Gauterin, F.: Abschlussbericht: Experteninterview zur Anforderungsanalyse heutiger und zukünftiger E/E Architekturen im Kraftfahrzeug. DOI: 10.5445/IR/1000054216, 2016.

[Ca19] Cai, Z.; Wang, A.; Zhang, W.; Gruffke, M.; Schweppe, H.: 0-days & Mitigations: Roadways to Exploit and Secure Connected BMW Cars. Black Hat USA 2019/, p. 39, 2019.

[CA20] CAN in Automation: CAN XL is knocking at the door./, Jan. 3, 2020, URL: https://www.can-cia.org/news/cia-in-action/view/can-xl-is-knocking-at-the-door/2020/1/3/, visited on: 01/03/2020.

[Ch18a] Choi, W.; Jo, H. J.; Woo, S.; Chun, J. Y.; Park, J.; Lee, D. H.: Identifying ECUs Using Inimitable Characteristics of Signals in Controller Area Networks. IEEE Transactions on Vehicular Technology 67/6, pp. 4757–4770, June 2018, ISSN: 0018-9545.

[Ch18b] Choi, W.; Joo, K.; Jo, H. J.; Park, M. C.; Lee, D. H.: VoltageIDS: Low-Level Communication Characteristics for Automotive Intrusion Detection System. IEEE Transactions on Information Forensics and Security 13/8, pp. 2114–2129, Aug. 2018, ISSN: 1556-6013.

[CS16] Cho, K.-T.; Shin, K. G.: Fingerprinting Electronic Control Units for Vehicle Intrusion Detection. In: Proceedings of the 25th USENIX Conference on Security Symposium. SEC'16, USENIX Association, Berkeley, CA, USA, pp. 911–927, 2016, ISBN: 978-1-931971-32-4, URL: http://dl.acm.org/citation.cfm?id=3241094.3241165.

[CS17] Cho, K.-T.; Shin, K. G.: Viden: Attacker Identification on In-Vehicle Networks. In: Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security. CCS '17, ACM, New York, NY, USA, pp. 1109–1123, 2017, ISBN: 978-1-4503-4946-8, URL: http://doi.acm.org/10.1145/3133956.3134001.

[Du19] Dupont, G.; Hartog, J. d.; Etalle, S.; Lekidis, A.: Network intrusion detection systems for in-vehicle network-Technical report. arXiv preprint arXiv:1905.11587/, 2019.

[GM13] Groza, B.; Murvay, S.: Efficient Protocols for Secure Broadcast in Controller Area Networks. IEEE Transactions on Industrial Informatics 9/4, pp. 2034–2042, Nov. 2013, ISSN: 1551-3203.

[Ha05] Haykin, S. S.: Adaptive filter theory. Pearson Education India, 2005.

[HKD11] Hoppe, T.; Kiltz, S.; Dittmann, J.: Security threats to automotive CAN networks—Practical examples and selected short-term countermeasures. Reliability Engineering & System Safety 96/1, pp. 11–25, 2011, ISSN: 0951-8320, URL: http://www.sciencedirect.com/science/article/pii/S0951832010001602.

[In19] Infineon Technologies: AURIX™ 32-bit microcontrollers for automotive and industrial applications. 2019.

[IV13] Illera, A.; Vidal, J.: Dude, WTF In My Car. DEFCON 21, 2013.

[JAC18] Junior, E. A. S.; Araujo-Filho, P. F. d.; Campelo, D. R.: Experimental Evaluation of Cryptography Overhead in Automotive Safety-Critical Communication. In: 2018 IEEE 87th Vehicular Technology Conference (VTC Spring). Pp. 1–5, June 2018.

[KH18] Kneib, M.; Huth, C.: Scission: Signal Characteristic-Based Sender Identification and Intrusion Detection in Automotive Networks. In: Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security. CCS '18, ACM, New York, NY, USA, pp. 787–800, 2018, ISBN: 978-1-4503-5693-0, URL: http://doi.acm.org/10.1145/3243734.3243751.

[Ko10] Koscher, K.; Czeskis, A.; Roesner, F.; Patel, S.; Kohno, T.; Checkoway, S.; McCoy, D.; Kantor, B.; Anderson, D.; Shacham, H.; Savage, S.: Experimental Security Analysis of a Modern Automobile. In: 2010 IEEE Symposium on Security and Privacy. Pp. 447–462, May 2010.

[Li17] Liu, J.; Zhang, S.; Sun, W.; Shi, Y.: In-Vehicle Network Attacks and Countermeasures: Challenges and Future Directions. IEEE Network 31/5, pp. 50–58, 2017, ISSN: 0890-8044.

[Lo19] Loukas, G.; Karapistoli, E.; Panaousis, E.; Sarigiannidis, P.; Bezemskij, A.; Vuong, T.: A taxonomy and survey of cyber-physical intrusion detection approaches for vehicles. Ad Hoc Networks 84/, pp. 124–147, 2019.

[LS12] Lin, C.; Sangiovanni-Vincentelli, A.: Cyber-Security for the Controller Area Network (CAN) Communication Protocol. In: 2012 International Conference on Cyber Security. Pp. 1–7, Dec. 2012.

[Lu14] Lu, N.; Cheng, N.; Zhang, N.; Shen, X.; Mark, J. W.: Connected Vehicles: Solutions and Challenges. IEEE Internet of Things Journal 1/4, pp. 289–299, Aug. 2014, ISSN: 2327-4662.

[MF19] Moreno, C.; Fischmeister, S.: Sender Authentication for Automotive In-Vehicle Networks through Dual Analog Measurements to Determine the Location of the Transmitter. In: Proceedings of the 5th International Conference on Information Systems Security and Privacy - Volume 1: ICISSP, SciTePress, pp. 596–605, 2019, ISBN: 978-989-758-359-9.

[MG14] Murvay, P.; Groza, B.: Source Identification Using Signal Characteristics in Controller Area Networks. IEEE Signal Processing Letters 21/4, pp. 395–399, Apr. 2014, ISSN: 1070-9908.

[MGF10] Müter, M.; Groll, A.; Freiling, F. C.: A structured approach to anomaly detection for in-vehicle networks. In: 2010 Sixth International Conference on Information Assurance and Security. Pp. 92–98, Aug. 2010.

[MV13] Miller, C.; Valasek, C.: Adventures in automotive networks and control units. Def Con 21/, pp. 260–264, 2013.

[MV15] Miller, C.; Valasek, C.: Remote exploitation of an unaltered passenger vehicle. Black Hat USA 2015/, p. 91, 2015.

[Pa17] Pan, L.; Zheng, X.; Chen, H.; Luan, T.; Bootwala, H.; Batten, L.: Cyber Security Attacks to Modern Vehicular Systems. J. Inf. Secur. Appl. 36/, pp. 90–100, Oct. 2017, ISSN: 2214-2126, URL: https://doi.org/10.1016/j.jisa.2017.08.005.

[Ro91] Robert Bosch GmbH: CAN Specification. 1991.

[St14] Staa, P. v.: How KETs can contribute to the reindustrialisation of Europe, European Technology Congress, Wrocław, June 12, 2014, URL: http://docplayer.net/21724658-Date-2012-how-kets-can-contribute-to-the-re-industrialisation-of-europe.html, visited on: 06/27/2019.

[ST17] STMicroelectronics: SPC58EEx, SPC58NEx 32-bit Power Architecture\textsuperscript® microcontroller for automotive ASIL-D applications. 2017.

[Yi19] Ying, X.; Sagong, S. U.; Clark, A.; Bushnell, L.; Poovendran, R.: Shape of the Cloak: Formal Analysis of Clock Skew-Based Intrusion Detection System in Controller Area Networks. IEEE Transactions on Information Forensics and Security 14/9, pp. 2300–2314, Sept. 2019, ISSN: 1556-6013.

[ZAA14] Zhang, T.; Antunes, H.; Aggarwal, S.: Defending Connected Vehicles Against Malware: Challenges and a Solution Framework. IEEE Internet of Things Journal 1/1, pp. 10–21, Feb. 2014, ISSN: 2327-4662.

Urheberrecht ./. Sicherheitsanalyse

Zivilrechtliche Aspekte der Sicherheitsforschung

Nico Müller, Tilo Müller, Felix Freiling[1]

Abstract: 2007 hat die Bundesregierung im Rahmen des 41. Strafrechtsänderungsgesetzes zur Bekämpfung von Computerkriminalität den umstrittenen Tatbestand des Ausspähens und Abfangens von Daten, inklusive deren Vorbereitung, unter Strafe gestellt. Durch das schon damals äußerst umstrittene Gesetz drohte die Tätigkeit von Sicherheitsforschern kriminalisiert zu werden, weswegen eine Klarstellung der Gesetzeslage durch eine Verfassungsbeschwerde erzwungen wurde. Jedoch wird neben der strafrechtlichen Relevanz von Sicherheitsanalysen oftmals die zivilrechtliche Seite des Urheberrechts übersehen, die eine beinahe ebenso große Bedrohung für Forscher darstellt, die sich kritisch mit der Sicherheit von Software auseinandersetzen. Im Folgenden soll daher die rechtliche Lage zu Sicherheitsanalysen kommerzieller Software für Forschungszwecke erörtert und die existierenden Grauzonen rechtlicher Unsicherheit ausgeleuchtet werden.

Keywords: Urheberrecht, Sicherheitsanalyse, Reverse Engineering

1 Einführung

Seit Anbeginn der menschlichen Zivilisation wurden im Laufe der Zeit immer neuere Technologien entwickelt. Während sich in den ostasiatischen Kulturkreisen der Grundsatz, dass Ideen nicht schützbar sind, durchgesetzt hat, bildete sich in den westlichen Ländern das Patentwesen und Urheberrecht. Das Urheberrecht garantiert dem Urheber für seine Werke eine gewisse Exklusivfrist, um ihm eine Vergütung seiner Arbeit zu gewähren. In Deutschland hat der Gesetzgeber erst im Laufe der 1990er Jahre mit dem WIPO-Urheberrechtsvertrag die Grundlagen für die geistigen Eigentumsrechte an digitalen Medien gelegt. Erste Implementierungen fanden 2001/2003 durch die EG-Softwarerichtlinie und die Novellierung des Urheberrechtsgesetzes statt.

Nicht jede neue Technologie ist von Anfang an sicher, weswegen sich insbesondere Firmen im Laufe der letzten Jahre die Frage gestellt haben, ob verwendete Computerprogramme Sicherheitslücken enthalten, die wie im Falle von STUXNET [FMC12] und dem WPA2-Key-Reinstallation-Angriff [VP17] von Dritten ausgenutzt werden können, um schwere Schäden an Firmennetzwerken anzurichten. Damit Sicherheitsforscher proaktiv solche Gefahren abwenden können, stehen sie im Konflikt mit den Interessen der Softwareentwickler, die

[1] Friedrich-Alexander-Universität Erlangen-Nürnberg, Department Informatik, Martensstr. 3, 91058 Erlangen, nico.mueller, tilo.mueller, felix.freiling@fau.de

doi:10.18420/sicherheit2020_08

oft explizit das *Reverse Engineering* der zu untersuchenden Software untersagen. Im Fall gekaufter Software hat der ursprüngliche Entwickler keine über die Urheberpersönlichkeitsrechte hinausgehenden Ansprüche mehr. Da die meisten Entwickler aber üblicherweise ein Lizenzmodell verwenden, behält sich eine Entwicklungsfirma als Lizenzgeber alle Rechte vor. Dennoch sieht der Gesetzgeber Ausnahmen vor, die durch Lizenzvereinbarungen nicht berührt werden dürfen. Aufgrund dieser Tatsache gestaltet sich die rechtliche Situation des Interessenkonflikts zwischen Lizenzgeber (Softwareentwickler) und Lizenznehmer (Sicherheitsforscher) als schwierig. Zur praktischen Orientierung leuchtet dieser Beitrag die Grauzonen rechtlicher Unsicherheit in diesem Kontext aus.

2 Rechtliche Natur eines Computerprogramms

Um weitere Aussagen über die rechtliche Situation zur Sicherheitsanalyse von Computerprogrammen zu treffen, wird zuerst die Natur eines Computerprogramms untersucht. Ein Computerprogramm besteht prinzipiell aus zwei juristisch relevanten Teilen: Schnittstellen, die eine Ein- und Ausgabe zu Menschen oder anderen Computerprogrammen ermöglichen und meist einem Standard folgen, sowie der eigentlichen Leistung des Entwicklers, den verarbeitenden Algorithmen, welche in Kombination miteinander die Funktionalität des Programms gewährleisten. Beide Teile und deren Unterkomponenten müssen sicher sein, um das Programm als Ganzes sicher zu machen. Als *Computerprogramm* wird lediglich der auszuführende Code gesehen, im Falle des Vorhandenseins von Begleitmaterialien, wie Grafiken und Musik, wird von *Computersoftware* gesprochen.

2.1 Schnittstellen

In der Computerprogramm-Richtlinie 2009/24/EG der Europäischen Union wird eine Schnittstelle im 10. Erwägungsgrund wie folgt definiert „Die Teile des Programms, die eine [...] Verbindung und Interaktion zwischen den Elementen von Software und Hardware ermöglichen sollen, sind allgemein als Schnittstellen bekannt". Diese Schnittstellen sind oft standardisiert und technisch notwendig, was gegen eine urheberrechtliche Schutzfähigkeit spricht. In der Tat wird die Schutzfähigkeit von technisch notwendigen „Designentscheidungen" in der Rechtswissenschaft verneint (vgl. BGH, 12.05.2011 - I ZR 53/10 - Seilzirkus). Zu dieser Einschätzung kommt auch die Fachliteratur [Ma18].

Dennoch könnten Abweichungen von dem Standard, einen urheberrechtlichen Schutz des Teilwerks nach sich ziehen, da diese eine eigene geistige Leistung darstellen. In der Rechtswissenschaft wird dabei der Begriff der sogenannten *kleinen Münze* verwendet, bei dem es sich um das kleinstmögliche, aber dennoch schutzfähige Werk handelt. Triviale Änderungen spielen bei der Anwendbarkeit dieser Regelung allerdings keine Rolle.

2.2 Algorithmen

Bei der Datenverarbeitung werden zwar oft Bibliotheksfunktionen verwendet, allerdings werden diese in ihrem Zusammenspiel ein schützenswerter Ausdruck der dahinterliegenden Idee. So ist ein hinreichend kleiner Teil eines jeden urheberrechtlich geschützten Werks nicht mehr urheberrechtlich geschützt, da ab einem gewissen Punkt keine persönliche Leistung mehr erkennbar ist. Dieser Punkt wird als, wie bereits oben genannt, kleine Münze bezeichnet. Für die verschiedenen vom Urheberrecht geschützten Werksarten galten verschiedene Maßstäbe, welche in den letzten Jahren auf ein einheitliches Niveau abgesenkt wurden.Die genauen Grenzen sind in den Gesetzestexten nicht ersichtlich, da meist für die Interpretation offene Formulierungen genutzt werden. So hat Bisges [Bi14, S. 224ff.] in seinem Buch konkrete Fälle der kleinen Münze ausgewertet und ist zu dem Schluss gekommen, dass für die *Vermutung der Schutzfähigkeit* vielmehr der investierte Aufwand und die wirtschaftliche Signifikanz der Werke für eine Entscheidung ausschlaggebend sind. Daraus ableitend kann man sagen, dass die meisten zu untersuchenden Computerprogramme in einem Gerichtsverfahren als schützenswert gelten werden.

3 Schutzbereich des Urheberrechts

Zuallererst stellt sich die Frage, was genau vom Urheberrecht erfasst ist. Die entsprechenden Vorschriften finden sich dabei im Urheberrechtsgesetz §69 a-g, wobei diese ein sogenanntes „lex specialis" zum normalen Urheberrecht bilden, d.h. die dort verankerten Grundsätze haben Vorrang vor anderen Definitionen, die beispielsweise in §2 UrhG getroffen werden.

So findet man in §69a die Bestimmungen zum Schutzgegenstand, die sich wesentlich von den in §2 festgelegten Kriterien unterscheiden. Dort ist von „individuelle[n] Werken in dem Sinne [...], dass sie das Ergebnis der eigenen geistigen Schöpfung des Urhebers sind" die Rede, was jedoch mit der sonst geforderten „persönliche[n] geistige[n] Schöpfung" nicht deckungsgleich ist, wie Marly beschreibt [Ma18]. Zielsetzung dieser Gesetzgebung ist es, dass möglichst alle Programme vom Urheberrechtsschutz abgedeckt sind, sofern sie nicht äußerst einfach strukturiert sind.

Eine Festlegung auf Programmcode ist dabei nicht erforderlich, sofern der Schöpfungsgedanke hinreichend fixiert worden ist. Dies umfasst beispielsweise Diagramme und technische Grafiken, die mit Zeichnungen eines Architektengebäudes vergleichbar sind, welche ebenfalls Urheberrechtsschutz genießen.

Um den komplexeren Sachverhalt der Schutzfähigkeit zu erörtern, wird nun für die Fälle eines kleinen Programms und eines komplexeren Programms eine Prüfung der Schutzfähigkeit anschaulich vorgenommen.

3.1 Beispiel Hallo-Welt-Programm

Das Hallo-Welt-Programm ist das seit jeher implementierte Programm, mit dem eine Programmiersprache erlernt wird. Bei dieser Art von Programm ist es die Aufgabe für Anfänger beispielsweise eine einfache Botschaft auf der Kommandozeile auszugeben. Dabei lässt sich eine nicht unerhebliche Banalität des Programms feststellen die, neben der quasi-Standardisierung, ein individuelles Werk nicht gestatten. So ist insbesondere bei Kleinstprogrammen, wie dem Hallo-Welt-Beispiel, der technische Hintergrundgedanke überwiegend, nicht jedoch der persönliche Schöpfungsgedanke, was einen urheberrechtlichen Schutz ausschließt. Marly knüpft die Schutzvermutung an die tatsächliche Komplexität des Programms, wobei jedoch darauf verwiesen wird, dass es sich lediglich um eine Vermutung, nicht aber um die tatsächliche Schutzfähigkeit handelt [Ma18].

3.2 Beispiel Professionelles Programm

Falls der Programmumfang größer sein sollte, insbesondere wenn das Programm von einer Firma entwickelt wurde, ist davon auszugehen, dass das Programm in den Schutzbereich des Urheberrechts fällt. Jedoch ist, wie bereits oben erwähnt, meist nur ein Indiz für die Individualität eines Programms vorhanden. Ob und inwiefern der persönliche Charakter eines Programms durch Übersetzungen und Optimierungen, wie sie von einem Compiler vorgenommen werden, zerstört wird, wurde jedoch bisher in der Fachliteratur nicht erörtert. Dennoch kann man sagen, dass wenn zur Lösung eines konkreten Problems, beispielsweise der symmetrischen Verschlüsselung von Daten, es aus technischer Sicht nur eine oder wenige Wahlmöglichkeiten gibt, hier AES oder 3DES, eine Schutzfähigkeit nicht gegeben ist. Die Rechtssprechung sagt dazu, dass es „an der nötigen Individualität [fehlen kann], wenn betriebswirtschaftliche, technische und funktionale Zwänge, etwa solche der Effizienz oder solche, die auf externen Faktoren (z.B. Kompatibilitätsanforderungen) oder auf dem jeweiligen Sachgebiet beruhen, den Gestaltungsspielraum einschränken" [Wa19]. Konkret war die GUI eines Programms, das zur Verwaltung durch den Betriebsrat entwickelt wurde, nach Ansicht des OLG Karlsruhe - 13.06.1994 6 U 5294 - „Bildschirmmasken" nicht schutzfähig, da diese „durch Gesetz oder durch Betriebsvereinbarungen und betriebliche Übung festgelegt" war.

4 Eingriff in den Schutzbereich

Um eine Beurteilung der während einer Sicherheitsanalyse typischen Untersuchungsmöglichkeiten vorzunehmen, werden diese im Folgenden vorgestellt. Da eine Vervielfältigung des Programms bei allen vorgestellten Analysemethoden einen wesentlichen Bestandteil darstellt, muss zwingend mindestens eine Nutzungsberechtigung der Software vorliegen, damit die Analyse erlaubt ist.

4.1 Blackboxing

Beim sogenannten *Blackboxing* [BZ17] wird das Programm geladen und anhand verschiedener Eingaben getestet, wobei die Ausgaben des Programms betrachtet werden. Da dadurch ein möglicher Anwendungsfall simuliert wird, handelt es sich um eine erlaubte Handlung nach §69d Abs. 3 UrhG. Da keine darüberhinausgehende Interaktion mit dem Programm stattfindet, bleibt der Quellcode unberührt und damit ist, falls eine Benutzungsberechtigung vorliegt, kein Eingriff in das Urheberrecht gegeben. In der Praxis eignet sich Blackboxing meist nur für die oberflächliche Suche nach Fehlern, insbesondere eignet es sich nicht dazu die Ursache eines Fehlers zu identifizieren. So können unübliche Eingaben, wie sie beispielsweise durch *Fuzzing* generiert werden, Fehlerzustände verursachen. Die technischen Gründe, warum ein Fehler auftrat, können aber nicht geklärt werden.

4.2 System Monitoring

Das *System Monitoring* stellt eine Sonderform des Blackboxing dar. Dabei wird die Kommunikation des Programms mit anderen Programmen und dem Betriebssystem überwacht, wobei die internen Vorgänge des Programms nicht genauer untersucht werden. Zwar sind Systemaufrufe und gesendete Pakete ein signifikanter Bestandteil eines jeden Programms, allerdings kann ähnlich wie oben argumentiert werden, dass der eigentliche Quellcode nicht betrachtet wird, sondern nur die Auswirkungen einer Programmausführung. Dadurch lassen sich Informationen über den Programmablauf auslesen, so dass sich System Monitoring wie Blackboxing zur initialen Fehlersuche eignet. Allerdings stellt auch das System Monitoring kein ausreichendes Werkzeug dar, um technische Ursache eines Fehlers zu identifizieren. Aus juristischer Sicht sind entstehende Ausgaben wie Pakete und Systemaufrufe durch Veranlassung des Bedieners entstanden, wenn auch unter Zuhilfename des zu untersuchenden Programms. Damit sind die so erzeugten Daten nicht durch das Recht des Programmurhebers geschützt.

4.3 Statische Analyse

Bei der statischen Analyse wird der zu untersuchende Code in seiner Gesamtheit von Maschinencode in Assemblercode und anschließend, je nach Analysesoftware bzw. Interesse seitens der Analysten, in eine höhere Programmiersprache umgewandelt.

Auf den ersten Blick betrachtet setzt also die statische Analyse eine Übersetzung von Maschinen- in Assemblercode und bei weiteren Analysen eine Übersetzung in eine höhere Programmiersprache voraus. Somit stellt die statische Analyse eine genehmigungspflichtige

Im weitesten Sinne stellt die Umwandlung ebenfalls eine Vervielfältigung dar, wobei nach Ansicht der Autoren diese Vervielfältigung nach §69 d Abs. 2 f UrhG erlaubt ist.

Handlung nach §69c UrhG dar und ist reglementiert nach §69e UrhG. Jedoch stellen sich essenzielle Fragen aus der Sicht des Untersuchenden, darunter

- Stellt eine Disassemblierung eine Umarbeitung oder eine Übersetzung im Sinne des §69c UrhG dar?
- In welchem Umfang ist die statische Analyse eines Programms erlaubt?
- Ist eine (Teil-)Nachprogrammierung/Dekompilierung des ursprünglichen Quellcodes noch durch das Urheberrecht geschützt?

Zum ersten Punkt kann mit ziemlicher Sicherheit gesagt werden, dass durch Disassemblierung keine Übersetzungshandlung im Sinne von §3 UrhG gegeben ist. Grund dafür ist, dass um ein urheberrechtlich geschütztes Werk zu erzeugen eine „persönliche geistige Schöpfung" nach §2 UrhG erforderlich ist. Da in den meisten Fällen die Disassemblierung eines Programms nach fest definierten Regeln durchgeführt wird, kann nicht von einer persönlichen geistigen Schöpfung die Rede sein. Dies sieht auch Schweyer [Sc14, S. 97 f.] ähnlich.

Der Punkt der Umarbeitung ist jedoch bedeutend interessanter, da bei einer Be- beziehungsweise Umarbeitung kein eigenständig schutzfähiges Programm entstehen muss. Die Umarbeitung ist dabei als Erweiterung des Bearbeitungsbegriffs zu verstehen, welche auch die Übersetzung, das Arrangement und andere Formen der Umarbeitung erfasst [Ma18]. Dabei hat §69c Abs. 2 Vorrang vor der in §23 definierten Bearbeitung, die das Kriterium der Eigenständigkeit verlangt. Darüber hinaus ist schon die Vornahme der Umarbeitung eine in §69c gelistete Handlung, und damit genehmigungspflichtig.

Zum zweiten Punkt gibt es nach Kenntnis der Autoren noch keine gerichtliche Entscheidung. Im 2. Gesetz zur Änderung des Urheberrechtsgesetzes [Bu93] findet sich „§69 d Abs. 3 ist nicht nur auf einzelne Programmelemente, sondern auch auf vollständige Programme anzuwenden." Damit impliziert der Gesetzgeber, dass eine Betrachtung des Assemblercodes für das gesamte Programm zulässig ist, da eine teilweise Betrachtung nur durch Betrachten der Teile des Assemblercodes möglich ist, wobei eine Dekompilierung des Programms in seiner Ganzheit wie unten beschrieben trotzdem nicht erlaubt ist. Weitere Bestätigungen dieser Sichtweise lassen sich in der Fachliteratur finden [Ka18].

4.4 Dekompilierung

Bei der *Dekompilierung* wird der Objektcode automatisiert in eine höhere Programmiersprache umgewandelt. Eine originalgetreue Rekonstruktion des ursprünglichen Quellcodes ist in der Praxis unmöglich. Automatisierung der Rückübersetzung wurde zwar versucht, allerdings mit begrenztem Erfolg, beispielsweise mittels „boomerang" [Pr06]. Eine Dekompilierung von beschränkten Programmteilen kann dagegen ohne Weiteres möglich sein.

Dennoch ist es aus praktischer Sicht fast unmöglich, den exakten Programmcode aus dem gegebenen Assemblercode zu rekonstruieren. Wo die Grenzen zwischen Nachahmung und Übernahme der Funktionalität liegen, kann im Allgemeinen auch nicht beantwortet werden.

Etwas schwieriger verhält es sich mit der Nachprogrammierung in einer höheren Sprache. Die Idee hinter einer jeden Software ist durch das Urheberrecht zwar nicht geschützt, allerdings die konkrete Ausdrucksweise, also das Programm an sich, ist es. Bei einem Plagiat, ob 1:1 übernommen oder nachgeahmt, das zum Ziel hat, den vorliegenden Assemblercode in einer höheren Programmiersprache nachzubilden, ist von einer Umarbeitung auszugehen [Ma18].

Rechtlich betrachtet ist anzunehmen, dass wenn es sich um ein vom Urheberrecht geschütztes Programm handelt, die Dekompilierung untersagt ist. Nicht nur stellt die Dekompilierung eine Umarbeitung dar, sondern auch eine Art Übersetzung, da Computerprogrammen den Sprachwerken zugeordnet werden. Die Dekompilierung wird durch die gesetzliche Schranke in §69e nur dann erlaubt, wenn eine Bereitstellung des Quellcodes durch den Urheber nicht erfolgt ist und Interoperabilität zu einem anderen Programm hergestellt werden muss. Wenngleich eine Teildekompilierung zwar wahrscheinlich erlaubt ist, insbesondere mit Hinblick auf die fehlende Schutzfähigkeit von Kleinstprogramm(-abschnitten) oder Schnittstellen, ist bei der vollständigen Dekompilierung von Illegalität auszugehen.

4.5 Dynamische Analyse

Die dynamische Analyse basiert darauf, dass das Programm im laufenden Zustand betrachtet wird. Dabei wird der Inhalt des Arbeitsspeichers und der Register ausgelesen und dargestellt.

Ein Beispiel für ein solches Programm ist der *GNU Enhanced Debugger* (gdb) [Fo19]. Dargestellt in der Benutzeroberfläche wird konkret der Registerinhalt, der Stack und der noch auszuführende Code, allerdings lassen sich durch Werkzeuge dieser Art auch alle Daten, die durch die Ausführung des Programms im Arbeitsspeicher vorhanden sind, ohne Weiteres auslesen. Ähnlich wie bei der statischen Analyse wird hierbei der Bytecode in Assemblercode umgewandelt. Hier ist jedoch der große Unterschied, dass nur ein kleiner Teil des Programmcodes ausgewertet wird, ohne das Gesamtprogramm zu übersetzen.

Dies ermöglicht eine genaue Betrachtung der internen Programmablaufe unter Realbedingungen und ist ein gängiges Mittel beim Finden von Sicherheitslücken dar. In Verbindung mit der statischen Analyse stellt sie das faktisch notwendige Werkzeug dar, um eine gründliche Analyse durchführen zu können. Jedoch gibt es auch hier rechtliche Bedenken, wenn es um die Kontrollflussveränderung von Programmen geht. So hat das OLG Hamburg, 23.04.2012 - 5 U 11/11, entgegen der Vorinstanz entschieden, dass eine Umarbeitung schon gegeben ist, wenn diese nicht manifestiert wurde, also nur im Arbeitsspeicher stattfindet. Entgegen dieser Entscheidung steht jedoch die Ansicht des LG München I, 27.05.2015 - 37 O 11673/14, ebenso die Ansicht des OLG München, die in dem als Adblock-Prozess bekannten Verfahren

In diesem Fall fällt das Teilprogramm aus der sog. kleinen Münze heraus.

einen Eingriff in den Kontrollfluss, sofern dieser nur zur Laufzeit vorgenommen wird, nicht ausreichend manifestiert sei, um eine Umarbeitung zu begründen. So heißt es in dem Urteil „Dieser Schutz des Rechteinhabers vor der unberechtigten Herstellung einer Bearbeitung eines Computerprogramms macht nach Ansicht der Kammer dann Sinn, wenn in das Computerprogramm als solches eingegriffen wird, insbesondere in den Quellcode, nicht jedoch, wenn im Rahmen einer Vervielfältigung im Arbeitsspeicher lediglich ein Teil eines Programms unterdrückt bzw. nicht aufgerufen wird." Jedoch konnte nicht abschließend geklärt werden, inwiefern sogenanntes „Patching" eine Umarbeitung darstellt, da es meist einen wesentlichen Bestandteil der Kontrollflussveränderung darstellt.

5 Gesetzliche Lizenzen

In fast jedem Lizenztext findet man Vertragsklauseln, die explizit eine Dekompilierung oder Disassemblierung des Programmcodes verbieten. Diese sind jedoch ausschließlich für den amerikanischen Rechtsraum relevant, da der deutsche Gesetzgeber in §69g Abs. 2 UrhG solche Klauseln für nichtig erklärt, so es sich um erlaubte Testhandlungen handelt.

Der deutsche Gesetzgeber sieht ebenfalls eine gesonderte Regelung für nicht kommerziell forschende Stellen vor. So darf man nach §60c UrhG bis zu 15% eines Werkes in einer nicht öffentlichen, klar abgegrenzten Umgebung verbreitet werden, bei der eigenen Forschung dürfen bis zu 75% eines urheberrechtlich geschützten Werkes kopiert werden, sofern das Werk der Hauptgegenstand der Forschung ist.Eine solche nicht öffentliche Umgebung stellt zum Beispiel eine Konferenz mit weiteren Wissenschaftlern dar, bei der das entsprechende Material nur für Besucher der Konferenz zugänglich gemacht wird.

5.1 Sonderfall Schadsoftware

Schadsoftware stellt eine spezielle Herausforderung an den Gesetzgeber und an den Untersuchenden dar, da eine uneingeschränkte Sicherheitsanalyse wünschenswert ist. Um eine dynamische Analyse zu verhindern setzt manche Schadsoftware Maßnahmen ein, um zum Beispiel virtuelle Arbeitsumgebungen zu erkennen [WF12]. Falls diese von Analysten umgangen werden, könnten sich Analysten theoretisch nach §95a UrhG schuldig machen, wobei es sich bei dem untersuchten Virus um Computersoftware, nicht aber um ein Computerprogramm handeln muss. Allerdings kann dahingehend argumentiert werden, dass durch die Verbreitung einer Schadsoftware der Entwickler derselben die Ansprüche an der exklusiven Verbreitung seiner Schöpfung aufgibt. Da sich Aufgrund der selbstpropagierenden Natur von Computerviren so gut wie jeder eine legitime Kopie des

Die Ansicht wird von der Kammer für Computerprogramme im Allgemeinen geteilt, ohne auf die Schutzfähigkeit von Webseiten näher einzugehen.
Die Sichtweise des OLG Hamburg wird im Rahmen dieses Urteils zwar erwähnt, jedoch wird die Gültigkeit des Urteils explizit als ebenfalls valide Sichtweise dargestellt.

Virus aneignen kann, ohne einer Lizenzvereinbarung zuzustimmen, darf auch jeder den Virus untersuchen. Oder anders ausgedrückt, weil ein Virus üblicherweise ohne allgemeine Geschäftsbedingungen daherkommt, kann man davon ausgehen, dass ein Entwickler keine Einwände gegen eine Untersuchung seines Werkes hat. In der Tat lasst sich eine Konstruktion der Einwilligung vollziehen.

Abschließend könnte man bei einem vorliegenden Virus ebenfalls mit §904 BGB beziehungsweise §34 StGB argumentieren, dass bei einer Abwägung der Rechtsgüter der Selbstschutz durch Analyse überwiegt.

5.2 Beauftragen eines Dritten mit der Untersuchung

Einer der ersten Schritte bei Sicherheitsanalysen ist meist die Beauftragung einer Drittfirma oder eines Forschers, da beide das nötige Fachwissen besitzt, um Sicherheitslücken zu erkennen. Allerdings ist dazu Zugang zur Software nötig, welcher meist durch Vervielfältigung und Weitergabe an die untersuchende Firma realisiert wird. Dies stellt allerdings nach §69c Abs. 1 UrhG eine genehmigungspflichtige Handlung dar, die der Nehmer in der Regel nicht tun darf. Zudem ist die kommerzielle Nutzung, die im Rahmen einer beauftragten Untersuchung stattfindet, von für Privatanwender bestimmter Software meist auch durch vertragliche Vereinbarungen untersagt. Dazu hat der I. Zivilsenat des Bundesgerichtshofs im Grundurteil vom 6.10.2016 - I ZR 25/15, bekanntgegeben: „[Der dazu berechtigte Lizenznehmer darf genehmigungspflichtige Handlungen nach §69c Abs. 1] auch dann ohne Zustimmung des Rechtsinhabers vornehmen, um das Funktionieren dieses Programms zu beobachten, zu untersuchen oder zu testen und die einem Programmelement zugrundeliegenden Ideen und Grundsatze zu ermitteln, wenn er dabei gewerbliche oder berufliche Zwecke verfolgt und der Lizenzvertrag lediglich eine Nutzung des Programms zu privaten Zwecken gestattet.”. Da zudem eine im Auftrag der Lizenznehmerfirma arbeitende Sicherheitsfirma oder Sicherheitsforscher eine handlungsbevollmächtigte Partei ist, darf, so der ursprüngliche Lizenznehmer dazu berechtigt ist, die beauftragte Firma die gleichen Handlungen vornehmen wie der Auftraggeber. Eine Außnahme bilden hierbei sogenannte „höchstpersönliche” Lizenzen, die explizit eine Person als Lizenznehmer bestimmen. In diesem Fall darf nur die im Vertrag benannte Person die Software verwenden und damit testen.

6 Fazit

Eine Sicherheitsanalyse ist prinzipiell erlaubt, wenngleich es viele verschiedene Arten gibt, auf die diese Analyse stattfinden kann. Zusammenfassend scheint in der Literatur der Konsens zu bestehen, dass alle Blackbox-Verfahren in Deutschland erlaubt sind, Whitebox-Verfahren bedürfen jedoch einer gesonderten Rechtfertigung. Dekompilierung ist im Rahmen einer Sicherheitsanalyse neben der technischen Problematik auch rechtlich

bedenklich, wohingegen die Dekompilierung von kleinsten Programmabschnitten geringer Komplexität im Sinne der kleinen Münze nach aktuellem Recht akzeptabel ist, muss bei der Dekompilierung ganzer Programme von Illegalität ausgegangen werden. Diese Grauzone ist aus Sicht der Sicherheitsforschung bedauerlich, da sich Software-Schwachstellen häufig gerade in komplexeren Code-Teilen verbergen. Ein Ausnahmetatbestand, sei es in Form einer gesetzlichen Schranke oder der Erweiterung des Testbegriffs aus §69d Abs. 3, ähnlich dem zur Herstellung von Interoperabilität wäre hier wünschenswert.

Abschließend ist zu sagen, dass das Urheberrecht bei weitem nicht das einzige Recht ist, das bei einer Sicherheitsanalyse beachtet werden muss. Vertragsrecht, Patentrecht, Gebrauchsmusterschutz und das Geschäftsgeheimnisgesetz sind oftmals ebenso wichtig wie das Urheberrecht. Dennoch stellt das deutsche Urheberrecht oftmals den ersten und oftmals einzigen Ansatzpunkt für Zivilklagen dar, da durch ungenaue Formulierungen sowie der Interpretationen mancher Autoren juristischer Fachliteratur, in beliebige Richtungen argumentiert werden kann. So hat selbst das BSI bereits Sicherheitsforschern mit einer Unterlassungsklage gedroht [Bö13]. In Abhängigkeit der Sichtweise der Kammer und der Qualität der Klage und Verteidigung kann keine verlässliche Voraussage über Prozesse getroffen werden.

Ferner kann auch die urheberrechtliche Schutzfähigkeit von Programmen in Frage gestellt werden, da Computercode oft nur Mittel zum Zweck ist, um Software zu verwirklichen. Oft sind Programme, die im professionellen Umfeld eingesetzt werden, auf Leistung optimiert und demzufolge die dahinterliegenden Entwicklungsentscheidungen technisch bedingt. Auf der anderen Seite wurde bis zum Urteil „Geburtstagszug" des BGH, 13.11.2013 - I ZR 143/12, unterschieden zwischen sogenannter angewandter Kunst und zweckfreier Kunst. Erstere hatte „[...] seit jeher höhere Anforderungen [an das Mindestmaßder Schöpfungshöhe] gestellt", wie der BGH in seinem bis dahin nicht aufgegebenen Urteil „Silberdistel", 22.06.1995 - I ZR 119/93, vermutet hat. Inwiefern sich ein doch so umfangreicher Schutz aufrecht erhalten lässt, kann jedoch im Rahmen dieser Arbeit nicht geklärt werden.

Die Leidtragenden dieser Unsicherheit sind die Sicherheitsforscher, die durch Klageandrohung und Prozesse unter Druck gesetzt werden können, da Urheberrechtsverletzungen mit hohen Geldstrafen oder Freiheitsstrafen von bis zu drei Jahren nach UrhG bedroht werden. Juristisch müsste Sicherheitsforschern heute empfohlen werden dem Grundsatz „In dubio mitius" (Im Zweifel für das Mildere) zu folgen und lieber nichts anstelle etwas falschem zu tun, also nach Schwachstellen gar nicht erst zu suchen. Unabhängige Sicherheitsanalysen aber erhöhen das Sicherheitsniveau unserer digitalen Infrastruktur und dienen damit dem Gemeinwohl. Das Recht und die Rechtsprechung sollte diesem Interesse Vorrang einräumen vor der Möglichkeit, die Aufdeckung von Sicherheitsschwachstellen in kommerzieller Software durch Anwendung des Urheberrechts zu unterdrücken.

Acknowledgements

Besonderer Dank gilt Prof. Franz Hofmann für seine Vorlesung über Grundlagenwissen zum geistigen Eigentum und seine Gesprächs- und Hilfsbereitschaft, sowie Dr. Till Jaeger für seine fachliche Beratung zu diesem Thema.

Literaturverzeichnis

[Bi14] Bisges, Marcel: Die Kleine Münze im Urheberrecht - Analyse des ökonomischen Aspekts des Werkbegriffs. Nomos, 2014.

[Bö13] Böck, Hanno: , GSTool: BSI bedroht Sicherheitsforscher, September 2013. https://www.golem.de/news/gstool-bsi-bedroht-sicherheitsforscher-1309-101531.html.

[Bu93] Bundestag: , Zweites Gesetz zur Änderung des Urheberrechtsgesetzes, June 1993. eingesehen unter http://www.urheberrecht.org/law/normen/urhg/1993-06-09/materialien/bgbl_I_910.php.

[BZ17] Bertoglio, Daniel Dalalana; Zorzo, Avelino Francisco: , Overview and open issues on penetration test, February 2017. https://journal-bcs.springeropen.com/articles/10.1186/s13173-017-0051-1.

[FMC12] Falliere, Nicolas; Murchu, Liam O; Chien, Eric: W32.Stuxnet Dossier. Technical report, Symantec, February 2012.

[Fo19] Foundation, Free Software: , GDB: The GNU Project Debugger, September 2019. https://www.gnu.org/software/gdb/.

[Ka18] Kaboth/Spies: , Beck Online Kommentar Urheberrecht, June 2018. BeckOK UrhR/Kaboth/Spies UrhG §69d Rn. 15-17.

[Ma18] Marly, Jochen: Praxishandbuch Softwarerecht, 7. Auflage. C.H. Beck, 2018.

[Pr06] Project, The Boomerang Decompiler: , Boomerang Decompiler, June 2006. http://boomerang.sourceforge.net/FAQ.php.

[Sc14] Schweyer, Florian: Die rechtliche Bewertung des Reverse Engineering in Deutschland und den USA. Mohr Siebeck, 2014.

[VP17] Vanhoef, Mathy; Piessens, Frank: Key Reinstallation Attacks: Forcing Nonce Reuse in WPA2. In: Proceedings of the 24th ACM Conference on Computer and Communications Security (CCS). ACM, 2017.

[Wa19] Wandtke/Bullinger/Grützmacher: UrhG § 69a Rn. 38, 5. Aufl. C.H. Beck, 2019.

[WF12] Willems, Carsten; Freiling, Felix C.: Reverse Code Engineering - State of the Art and Countermeasures. it - Information Technology, 54:53–63, 2012.

Informationssicherheit für KRITIS-Betreiber: Kritische Dienstleistungen systematisch schützen

Workshop-Konzept zur Definition des Geltungsbereichs (Scope) und Ableitung geeigneter KRITIS- und IT-Schutzziele als Basis für ein erfolgreiches ISMS

Frauke Greven[1]

Abstract: Der Aufbau und Betrieb eines erfolgreichen Informations-Sicherheits-Management-Systems beginnt mit der passgenauen Festlegung des zu schützenden Geltungsbereichs und der Definition konkreter Schutzziele. Sie gelten als Ausgangspunkt für eine ganzheitliche Risikobeurteilung und die Auswahl sowie Realisierung geeigneter Schutzmaßnahmen. Betreiber Kritischer Infrastrukturen (KRITIS) sind verpflichtet, gegenüber dem Bundesamt für Sicherheit in der Informationstechnik (BSI) die Umsetzung angemessener technischer und organisatorischer Maßnahmen zum Schutz ihrer kritischen Dienstleistungen (kDL) nachzuweisen. Eine wichtige Arbeitshilfe stellen die von KRITIS-Betreibern bzw. Branchenverbänden entwickelten und vom BSI als geeignet festgestellten Branchenspezifischen Sicherheitsstandards (B3S) dar. Das BSI unterstützt im Rahmen der KRITIS-Betreuung mit einer Workshop-Reihe dabei, den Geltungsbereich auf die branchenspezifischen kDL zu fokussieren und entsprechende KRITIS- und IT-Schutzziele abzuleiten. Die Ergebnisse können KRITIS-Betreiber nutzen, um ihren individuellen Geltungsbereich für die alle zwei Jahre erforderlichen Nachweise gemäß §8a (3) BSIG festzulegen. So lassen sich notwendige Prüfungen wie Sicherheitsaudits und Zertifizierungen zielgerichtet mit möglichst wenig personellem und finanziellem Aufwand durchführen.

Keywords: Kritische Infrastruktur (KRITIS), Kritische Dienstleistungen (kDL), Managementsystem für Informationssicherheit (ISMS), Geltungsbereich (Scope), KRITIS- und IT-Schutzziele

1 Einleitung: Informationssicherheit für KRITIS-Betreiber – auf den Scope kommt es an

Betreiber Kritischer Infrastrukturen (KRITIS), die durch IT-SiG und BSI-KritisV reguliert werden, sind dazu verpflichtet, angemessene organisatorische und technische Vorkehrungen zur Vermeidung von Störungen der Verfügbarkeit, Integrität, Authentizität und Vertraulichkeit ihrer informationstechnischen Systeme, Komponenten oder Prozesse zu treffen, die für die Funktionsfähigkeit der von ihnen betriebenen Kritischen Infrastrukturen maßgeblich sind. Gemäß §8a (3) BSIG müssen KRITIS-Betreiber darüber alle zwei Jahre gegenüber dem BSI einen geeigneten Nachweis erbringen.

[1] Bundesamt für Sicherheit in der Informationstechnik, Referat WG13 - KRITIS-Sektoren Ernährung, Gesundheit, Transport und Verkehr, Godesberger Allee 185 -189, 53175 Bonn, frauke.greven@bsi.bund.de

doi:10.18420/sicherheit2020_09

Basis für ein erfolgreiches Informations-Sicherheits-Management-System (ISMS) sind eine IT-Sicherheitskonzeption (KRITIS- und IT-Schutzziele, Schutzbedarfsfeststellung, Risikoanalyse etc.) und die betreiberindividuelle Definition des Geltungsbereichs (Scope). Erfahrungen aus der Praxis zeigen, dass es besonders zielführend ist, dabei auf die kritische Dienstleistung (kDL) gemäß BSI-KritisV abzustellen. Laut Orientierungshilfe des BSI zum B3S sind in diesem Kontext alle informationstechnischen Systeme, Komponenten und Prozesse, sowohl Standard-IT als auch branchenspezifische OT, zu betrachten, die die kDL direkt oder indirekt unterstützen oder von denen die kDL abhängig ist und bei deren Ausfall es zu einer Störung der kritischen Dienstleistung kommen könnte. Auch alle Schnittstellen z. B. zu externen Dienstleistern oder anderen IT-Netzen sind zu erfassen.[2]

2 Workshop-Konzept: KRITIS-typische Aspekte im Fokus

Im Rahmen der KRITIS-Betreuung unterstützt das BSI Betreiber und Branchenverbände mit einer Workshop-Reihe dabei, KRITIS-typische Aspekte bei der Festlegung von Geltungsbereich und Schutzzielen im ISMS zu fokussieren. Ziel ist es, Anforderungen an die qualitative und quantitative Versorgung der Bevölkerung mit der kritischen Dienstleistung in Normallagen bzw. besonderen Lagen zu definieren und diese bei der Auswahl sowie Umsetzung geeigneter Schutzmaßnahmen zu erfüllen. Dabei sollen die Sichtweisen von Leitungsebenen, IT-Fachleute sowie Beschäftigten aus den Fachabteilungen der Betreiber einbezogen werden.

2.1 Mit IT-Grundschutz-Methodik zu passgenauen Ergebnissen

Das strukturierte Vorgehen in den Workshops fußt auf den BSI-Standards 200-2[3] und 100-4[4]. Es ist an das von der Allianz für Cyber-Sicherheit zusammen mit dem Grundschutzreferat des BSI erprobte interaktive Workshop-Format[5] angelehnt und folgt wie dieses einem partizipatorischen Ansatz und dem Subsidiaritätsprinzip: Die Teilnehmenden erhalten notwendige Informationen zu Grundlagen der IT-Grundschutz-Methodik, um die Arbeitsergebnisse selbstständig zu erstellen und weiterzuentwickeln.

[2] Bundesamt für Sicherheit in der Informationstechnik (Herausgeber): Orientierungshilfe zu Inhalten und Anforderungen an branchenspezifische Sicherheitsstandards (B3S) gemäß § 8a (2) BSIG, S. 14, 2017.

[3] Bundesamt für Sicherheit in der Informationstechnik (Herausgeber): BSI-Standard 200-2: IT-Grundschutz-Methodik, 2017.

[4] Bundesamt für Sicherheit in der Informationstechnik (Herausgeber): BSI-Standard 100-4: Notfallmanagement, 2008.

[5] Greven, Frauke/Klein, Birger: „Großer“ IT-Grundschutz für kleine und mittelständische Unternehmen – Workshop-Konzept für die Erstellung von IT Grundschutz-Profilen, in: Bundesamt für Sicherheit in der Informationstechnik (Herausgeber): IT-Sicherheit als Voraussetzung für eine erfolgreiche Digitalisierung, Tagungsband zum 16. Deutschen IT-Sicherheitskongress, S. 97 – 107, 2019.

2.2 Workshop-Reihe - Idealtypus

Die konkrete Ausgestaltung bezüglich Inhalte und Methoden sowie die erzielten Ergebnisse der jeweils ca. fünfstündigen Workshops können in der Praxis aufgrund des partizipatorischen und subsidiären Ansatzes variieren. Das liegt insbesondere daran, dass sich die gemeinsamen Vereinbarungen und Arbeitsergebnisse an den Kenntnissen, Bedürfnissen und Interessen der Beteiligten ausrichten.

Idealtypisch lassen sich in nur drei Workshop-Terminen die im Folgenden beschriebenen Arbeitsergebnisse erreichen:

- **Erster Workshop: kDL definieren**

  Die Kick-off-Veranstaltung richtet sich an die Leitungsebenen von KRITIS-Betreibern (ggf. mit IT-Verantwortung), Beschäftigte aus dem Bereich IT bzw. IT-Sicherheit und ggf. zuständige Fachabteilungen sowie an Multiplikatoren aus Branchenverbänden. Die Teilnehmenden erfahren Wissenswertes über KRITIS-relevante Begriffsbestimmungen sowie Grundlagen zur Business Impact Analyse. Arbeitsergebnisse: Definition der kDL (Qualität, Quantität, zugehörige Geschäftsprozesse, KRITIS- und IT-Schutzziele), Festlegung des Geltungsbereichs (Anlage; selbst und extern erbrachte Teile der kDL; Zusammenspiel mit z. B. Zulieferern oder Kunden sowie Schnittstellen und Abhängigkeiten)[6]

- **Zweiter Workshop: Prozess- und Netzplan erstellen**

  Im zweiten Workshop arbeiten insbesondere Beschäftigte aus dem Bereich IT bzw. IT-Sicherheit interaktiv in moderierten Gruppen. Bei Bedarf erhalten sie Hinweise zu den Themen Modellierung von Prozessen sowie Ableitungen für die IT- und OT-Infrastruktur. Arbeitsergebnisse: Prozess- und Netzplan, Aufstellung der Zielobjekte (informationstechnische Systeme; Komponenten; Prozesse sowie Rollen; Personen und Organisationseinheiten etc.)[7], Beginn Erstellung einer „Landkarte“ [8]

- **Dritter Workshop: Schutzziele auf Zielobjekte anwenden**

  Der letzte Workshop dient der Schutzbedarfsfeststellung gemäß der im Kick-off definierten KRITIS- und IT-Schutzziele bezogen auf die im zweiten Arbeitstreffen erfassten Zielobjekte durch insbesondere IT-Fachleute. Es empfiehlt sich, die Ergebnisse mit den Beschäftigten aus Fachabteilungen sowie der Leitungsebene zu diskutieren, um abschließend noch einmal gemeinsam prüfend den KRITIS-Blick auf die „Landkarte“ zu richten. Arbeitsergebnisse: Fertigstellung der „Landkarte“

[6] Wie empfohlen in: Bundesamt für Sicherheit in der Informationstechnik (Herausgeber): Orientierungshilfe zu Nachweisen gemäß §8a Absatz 3 BSIG“, Bundesamt für Sicherheit in der Informationstechnik, S. 8, 2019.
[7] Ebd., S.8.
[8] Die „Landkarten“ werden je kDL/Branche erarbeitet und zeigen eine Übersicht zu allen wesentlichen Erkenntnissen aus Prozess- und Netzplanerstellung sowie der Schutzbedarfsfeststellung. Details zu dem Konzept der Landkarten: Greven, Frauke/Klein, Birger, ebd.

3 Ausblick

Die Betreiber kritischer Infrastrukturen, auch zunehmend jene, die unterhalb der Schwellenwerte nach BSI-KritisV liegen, erkennen ihre Verantwortung bezüglich ihres Versorgungsauftrags an. Mit der hier vorgestellten Workshop-Reihe lädt das BSI Betreiber und ihre Branchenverbände dazu ein, den KRITIS-Blick auf ihre relevanten Geschäftsprozesse weiter zu schärfen. Perspektivisch kann es sinnvoll sein, auch externe Dienstleister wie Hersteller oder Integratoren von KRITIS-relevanten Komponenten, Zulieferer sowie andere Akteure an den Schnittstellen zur kDL einzubeziehen.

Die Teilnahme an den vom BSI moderierten Workshops bietet allen Beteiligten vielfältige Vorteile – sei es der Ausbau von Branchen- und IT-Sicherheits-Fachkenntnissen, die Möglichkeit der Vernetzung mit anderen Akteuren für einen nachhaltigen und vertiefenden Erfahrungsaustausch oder die Nutzung von Synergien im Rahmen des (neu) entstandenen Netzwerks. Die Arbeitsergebnisse können für die Erstellung und Weiterentwicklung von Branchenspezifischen Sicherheitsstandards (B3S) genutzt werden. Sie erleichtern es KRITIS-Betreibern, ihren individuellen Geltungsbereich für die alle zwei Jahre erforderlichen Nachweise gemäß §8a (3) BSIG festzulegen. Die von dem jeweiligen Betreiber konkretisierte Variante könnte z. B. als erläuternde Anlage zu dem Nachweis dienen. Mit dem Fokus auf die kDL ließen sich notwendige Prüfungen wie Sicherheitsaudits und Zertifizierungen zielgerichtet mit möglichst wenig personellem bzw. finanziellem Aufwand durchführen.

4 Literaturverzeichnis

[Bs08] Bundesamt für Sicherheit in der Informationstechnik (Herausgeber): BSI-Standard 100-4: Notfallmanagement, 2008.

[Bs17] Bundesamt für Sicherheit in der Informationstechnik (Herausgeber): BSI-Standard 200-2: IT-Grundschutz-Methodik, 2017.

[GK19] Greven, Frauke/Klein, Birger: „Großer“ IT-Grundschutz für kleine und mittelständische Unternehmen – Workshop-Konzept für die Erstellung von IT Grundschutz-Profilen, in: Bundesamt für Sicherheit in der Informationstechnik (Herausgeber): IT-Sicherheit als Voraussetzung für eine erfolgreiche Digitalisierung, Tagungsband zum 16. Deutschen IT-Sicherheitskongress, S. 97 – 107, 2019.

[Or17] Bundesamt für Sicherheit in der Informationstechnik (Herausgeber): Orientierungshilfe zu Inhalten und Anforderungen an branchenspezifische Sicherheitsstandards (B3S) gemäß § 8a (2) BSIG, S. 14, 2017.

[Or19] Bundesamt für Sicherheit in der Informationstechnik (Herausgeber): Orientierungshilfe zu Nachweisen gemäß §8a Absatz 3 BSIG“, Bundesamt für Sicherheit in der Informationstechnik, S. 8, 2019.

Project OVVL – Threat Modeling Support for the entire secure development lifecycle

Andreas Schaad [1]

Abstract: OVVL (the Open Weakness and Vulnerability Modeller) is a tool and methodology to support threat modeling in the early stages of the secure software development lifecycle. We provide an overview of OVVL (https://ovvl.org), its data model and browser-based UI. We equally provide a discussion of initial experiments on how identified threats in the design phase can be aligned with later activities in the software lifecycle (issue management and security testing).

Keywords: Threat Modeling, Secure Software Development Lifecycle

1 Introduction

Threat Modeling is an effort to identify potential security design flaws and vulnerabilities as early as possible in the software development lifecycle to avoid costly fixes and re-engineering activities in the later stages. Apart from STRIDE [Sh14] and TAM2 [SB12], surprisingly little has been done so far regarding practical and lightweight "hands-on" methodologies as well as automated tool support - although it is conventional wisdom that early identification of security issues is good engineering and management practice. We thus present OVVL, a methodology and tool that integrates into the different stages of the software lifecycle to further drive technical and managerial acceptance of threat modeling. The goal is to not only identify vulnerabilities, but to track them throughout the later lifecycle stages and thus increase the quality of cybersecurity in real-world systems.

2 OVVL (the Open Weakness and Vulnerability Modeller)

2.1 Underlying Methodology & Data Model

OVVL [SR19] is based on (and extends) the commonly accepted STRIDE methodology for threat modeling [Sh14]. A complex system design can be abstracted to data flow diagrams and a first match is made with respect to possible STRIDE threats (Spoofing, Tampering, Repudiation, Information Disclosure, Denial, Elevation of Privileges). What distinguishes the OVVL approach from the known body of work is that the OVVL data model allows to enrich a design with information about the components that may be used for realising a system. For example, a software architect can describe that the underlying

[1] Hochschule Offenburg, Fakultät Medien, Badstr. 24, Offenburg, andreas.schaad@hs.offenburg.de

doi:10.18420/sicherheit2020_10

database will be a MongoDB with a certain release level. Based on that information, OVVL will query existing vulnerability databases and identify CVE entries, i.e. known and reported real-world vulnerabilities. Information about the identified threats (at design level) and known vulnerabilities (at operational level) can then be pushed to available tooling in the software development lifecycle. In the context of this paper we discuss bug tracking (3.1) and vulnerability testing (3.2).

2.2 Technical Architecture

OVVL is a pure web application enabled by Spring Boot. The javascript application logic is consumed through an Angular UI / frontend. Data is managed in a MongoDB and all OVVL services are available through Swagger-generated REST APIs. The OVVL server can be operated as a virtual machine to be used internally where required.

2.3 Case Study 1: Threat modeling of a cloud application (CloudProtect)

OVVL has been developed as part of the CloudProtect project[2] and is currently used to provide an initial security analysis of the CloudProtect architecture (Figure 1). This analysis yielded over a dozen potential design threats as well as 5 known existing major vulnerabilities (CVE entries > 7.0) which at this stage of the project serves as a good basis for technical discussions.

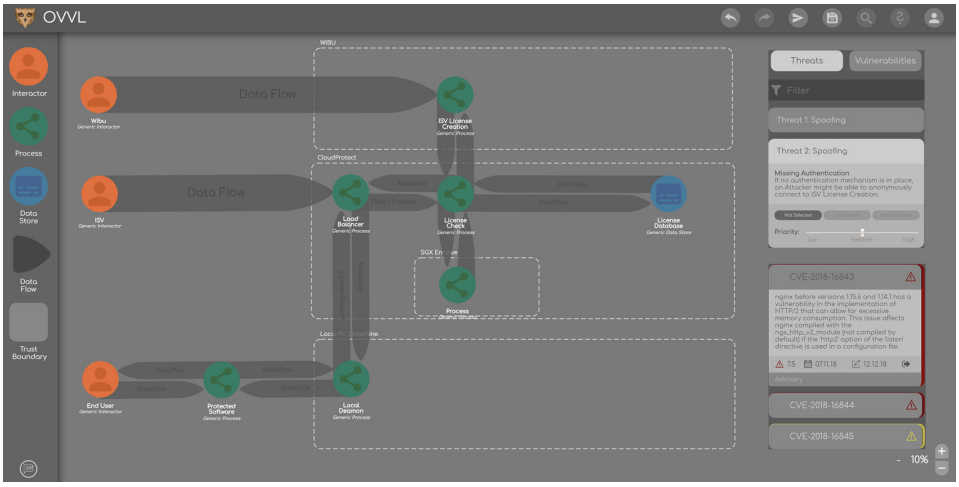


Fig. 1: OVVL Model of CloudProtect (with examples of threats and vulnerabilities)

[2] BMBF KMU-Innovativ “CloudProtect” FKZ: 16KIS0850

2.4 Case Study 2: Retrospect Threat Modeling (green&easy)

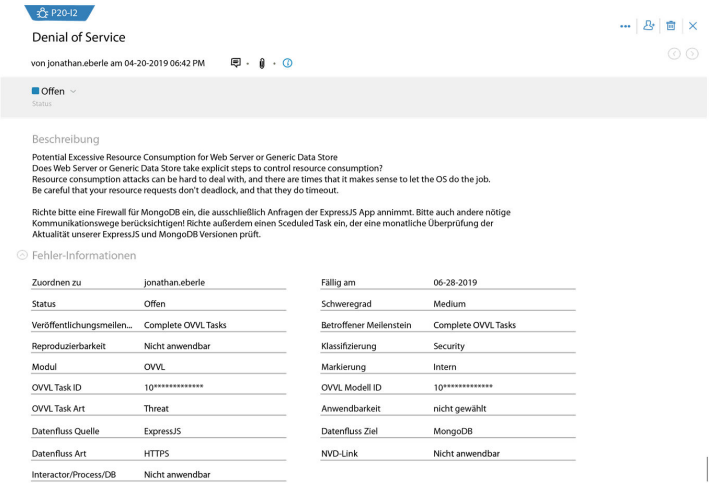
We used OVVL to model a web application called “green&easy” [Eb19]. The interesting fact here is that the responsible architect was not security trained and used OVVL in retrospect after he and his team had already provided a first working proof of concept.

The OVVL-based system design consisted of 2 interactors, 7 processes, 1 database grouped over 4 trust boundaries. 22 threats were automatically identified by OVVL out of which 20 threats were deemed as critical enough to be pushed into the projects issue tracker (Zoho). The majority of these identified threats (16) addressed possible elevation of privileges. On basis of the provided metadata, 13 immediate vulnerabilities where flagged with respect to used components such as MongoDB, ExpressJS and Redux. In fact, the analysis made in [Eb19] provides detailed estimates that the overall design effort is even reduced from 808 to 702 developer hours when using OVVL (now already including a detailed threat and vulnerability analysis).

3 OVVL Integration into the Secure Development Lifecycle

3.1 Bugtracker Integration

OVVL supports integration with the Zoho bugtracker. This allows to automatically push identified threats and associated metadata into an off-the-shelf project management tool (Figure 2). Based on this, a project manager can track whether identified threats or vulnerabilities are resolved throughout the project lifecycle. A detailed analysis of the Zoho API and corresponding data items in OVVL has been provided [Eb19].


P20-I2

Denial of Service

von jonathan.eberle am 04-20-2019 06:42 PM

Offen

Status

Beschreibung

Potential Excessive Resource Consumption for Web Server or Generic Data Store
Does Web Server or Generic Data Store take explicit steps to control resource consumption?
Resource consumption attacks can be hard to deal with, and there are times that it makes sense to let the OS do the job.
Be careful that your resource requests don't deadlock, and that they do timeout.

Richte bitte eine Firewall für MongoDB ein, die ausschließlich Anfragen der ExpressJS App annimmt. Bitte auch andere nötige Kommunikationswege berücksichtigen! Richte außerdem einen Sceduled Task ein, der eine monatliche Überprüfung der Aktualität unserer ExpressJS und MongoDB Versionen prüft.

Fehler-Informationen

Zuordnen zu	jonathan.eberle	Fällig am	06-28-2019
Status	Offen	Schweregrad	Medium
Veröffentlichungsmeilen...	Complete OVVL Tasks	Betroffener Meilenstein	Complete OVVL Tasks
Reproduzierbarkeit	Nicht anwendbar	Klassifizierung	Security
Modul	OVVL	Markierung	Intern
OVVL Task ID	10**********	OVVL Modell ID	10**********
OVVL Task Art	Threat	Anwendbarkeit	nicht gewählt
Datenfluss Quelle	ExpressJS	Datenfluss Ziel	MongoDB
Datenfluss Art	HTTPS	NVD-Link	Nicht anwendbar
Interactor/Process/DB	Nicht anwendbar		

Fig. 2: OVVL Threat managed in the Zoho Bugtracker

3.2 Vulnerability Testing

As described earlier, OVVL can help to identify design threats as well as existing vulnerabilities. We thus analysed whether reported CVEs can be used as a reference when applying automated vulnerability scanners such as Nessus (as well as openvas and nmap_vulners) in the later stages of the software lifecycle. As shown in figure 3, a test engineer can then identify whether an issue reported by Nessus had already been identified in OVVL (and thus early) or whether a new vulnerability was found. OVVL provides the needed interfaces and reporting UIs to Nessus.

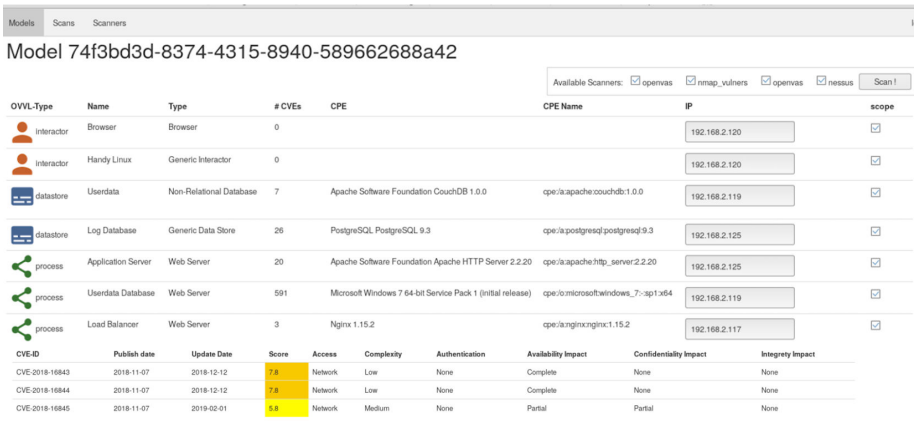


Fig. 3: OVVL data used by vulnerability scanners (Nessus, nmap-vulners, openvas)

4 Summary & Conclusion

OVVL is an open-source project available at https://github.com/OVVL-HSO. OVVL differs from existing open source tools by distinguishing between design threats and technical vulnerabilities. OVVL data can be used by other tools in the development chain such as issues trackers or vulnerability scanners.

Bibliography

[Eb19] Jonathan Eberle: „Konzeption eines agilen Software-Sicherheitsmanagements am Beispiel einer Mobile App “, Hochschule Offenburg, 2019, Nr. 1676005854

[SB12] Andreas Schaad, Mike Borozdin “TAM^2: Automated threat analysis”, ACM SAC 2012

[Sh14] Adam Shostack “Threat Modeling: Designing for Security” Wiley, 2014

[SR19] Andreas Schaad, Tobias Reski: "Open Weakness and Vulnerability Modeler" (OVVL): An Updated Approach to Threat Modeling. ICETE (2) 2019: 417-424

Lean Privacy by Design

Ein Ansatz zur Behandlung von Datenschutzanforderungen in agilen Entwicklungsprozessen

Jan Zibuschka[1], Christian Zimmermann[2]

Abstract: Durch agile Prozesse und Praktiken können Firmen in komplexen, offenen Ökosystemen flexbiler und effizienter agieren. Agile Methoden werden auch in Anwendungsfeldern verwendet, die sich durch besondere Datenschutz- und Sicherheitsanforderungen auszeichnen. Allerdings wird die gegenwärtig übliche Umsetzung solcher Anforderungen im SCRUM-Vorgehen von SCRUM-Teams als umständlich und schwer nachvollziehbar und seitens der Datenschutzverantwortlichen als kostenintensiv und überkompliziert empfunden. Um diese Probleme zu adressieren, schlagen wir einen auf lean thinking basierenden Ansatz zur Behandlung von Datenschutzanforderungen in agilen Entwicklungsprozessen vor.

Keywords: Datenschutz; lean; agil; SCRUM; PbD

1 Einleitung

Ausgehend von kleinen Software-Entwicklungsteams hat die sogenannte agile Transformation ganze Industrien und Firmen erfasst, die durch agile Prozesse und Praktiken in komplexen, offenen Ökosystemen flexbiler und effizienter agieren können [EP17]. Dies hat grundsätzliche Auswirkungen auf Datenschutzpraktiken, die sich auf verschiedenen Ebenen äußern. Generell stellen regulierte Anwendungsdomänen eine Herausforderung für agile Herangehensweisen dar [Fi13]. Dieser Beitrag beschreibt einen auf *lean UX* [Li14] aufsetzenden Ansatz zur Einbindung von Datenschutz-Anforderungen in agile Prozesse am Beispiel einer Integration in ein Referenzmodell des SCRUM-Prozesses.

2 Praktische Erfahrungen

Der *lean privacy by design*-Ansatz wurde im Rahmen mehrerer Industrie- und Forschungsprojekte, die agil organisiert waren, entwickelt. Die Industrieprojekte wurden teilweise

[1] Robert Bosch GmbH, Zentralbereich Forschung und Vorausentwicklung, Renningen, 70465 Stuttgart, Deutschland jan.zibuschka@de.bosch.com
[2] Robert Bosch GmbH, Zentralbereich Forschung und Vorausentwicklung, Renningen, 70465 Stuttgart, Deutschland christian.zimmermann3@de.bosch.com

doi:10.18420/sicherheit2020_11

von Mitarbeitern eines großen Industrieunternehmens, teilweise von Mitarbeitern eines unabhängigen Instituts in Beratungsfunktion in Organisationen durchgeführt. Wir können hier nur konkrete Beispiele für öffentlich geförderte Projekte geben: Im Rahmen der Datenschutz- und Sicherheitsarchitektur von ENTOURAGE [ZHK19] wurde auf Seiten einer teilnehmenden Organisation SCRUM eingesetzt. Gleiches gilt für BIG IoT [He16]. Dem gegenüber standen Erfahrungen von Forschungsinstituten und von Industrie gesponsorten Lehrstühlen in der Entwicklung von Datenschutz und Sicherheitskomponenten in Industrie- und öffentlich geförderten Projekten ohne agile Prozesse, etwa PRIME [Ra07]. Es wurden folgende spezifischen Probleme der Umsetzung von Datenschutz in agilen Projekten identifiziert:

- Das agile Team empfindet Datenschutz- und Sicherheits-Anforderungen als zu umständlich
- Das agile Team fühlt sich in der Umsetzungsphase der Sprints durch Unklarheiten behindert
- Die Datenschutz-Compliance-Funktion muss kostenintensiv Datenschutz-Verantwortliche Vollzeit in Projekte einbetten
- Datenschutz-Verantwortliche in Projekten sind erheblichen Komplexitäten ausgesetzt, die nicht in direkter Verbindung zu Datenschutz stehen

Diese werden durch aktuelle Ansätze, etwa Bente et al. [BEZ19], nicht adressiert. Hier ist vorgesehen, dass der Datenschutzverantwortliche jedes Software-Artefakt einzeln hinsichtlich seiner Datenschutz-Implikationen bewertet, was die Eigenständigkeit der Teams einschränkt und einen hohen Aufwand erfordert. Genau hierdurch werden die von uns beobachteten Probleme ausgelöst.

3 Ansatz

In der Produktion existiert eine etablierte Methode, die eingesetzt wird, wenn interdisziplinäre, autonome Teams nutzerzentrisch Produkte mit hoher Variabilität entwickeln, die in bestimmten Qualitätsdimensionen nahezu perfekt sein müssen, die also unsere spezifischen Probleme adressieren würde: *lean thinking* [Me05]. Im Kontext von SCRUM existiert für das Qualitätsattribut Benutzererlebnis *lean UX* [Li14], in dem agile Teams während der Sprints Annahmen über die Anforderungen der Nutzer machen und so ihre Agilität behalten, diese Anforderungen und auf ihnen basierende Entscheidungen jedoch dokumentieren, sodass UX-Experten diese später validieren können.

Dieses Konzept übertragen wir direkt auf den Datenschutz. Wir gehen davon aus, dass es für das Projekt oder die umgebende Organisation eine Datenschutz-Compliance-Funktion gibt, welche letzlich die Verantwortung für Datenschutzentscheidungen trägt, etwa ein Datenschutzbeauftragter. Wir gehen weiter von einem agilen Team aus, das in einem SCRUM-Prozess arbeitet. Wir integrieren *lean privacy by design* dann wie folgt (siehe Abb. 1):

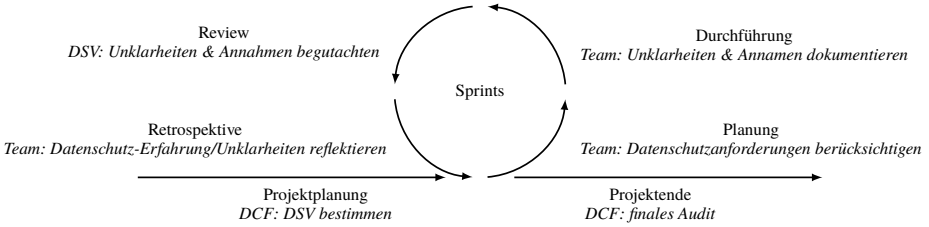


Abb. 1: *Lean Privacy by Design* im SCRUM-Entwicklungs-Prozess

Projektplanung Die Datenschutz-Compliance-Funktion (DCF) ernennt einen Datenschutzverantwortlichen (DSV) für das Projekt. Der Verantwortliche identifiziert initiale Datenschutzanforderungen.

Sprint-Planung Das agile Team führt seine Sprint-Planung unter Berücksichtigung der bekannten Datenschutzanforderungen aus. Falls Annahmen außer Kraft gesetzt worden sind, sollten die resultierenden Änderungen an Komponenten in der Regel hoch priorisiert werden.

Sprint-Durchführung Während der Sprints dokumentiert das Team Unklarheiten, die sich ergeben, macht Annahmen, was diese Unklarheiten angeht und dokumentiert Unklarheiten, Annahmen, und betroffene Komponenten.

Sprint-Review Der nur hier involvierte Datenschutz-Verantwortliche begutachtet die neu hinzugekommenen Unklarheiten/Annahmen und bestätigt diese oder setzt sie außer Kraft

Sprint-Retrospektive Das Team reflektiert Datenschutz-Erfahrungen, wodurch Datenschutz-Kompetenz und -Bewusstsein gestärkt wird.

Externes Release Die Datenschutz-Compliance-Funktion veranlasst ein finales Datenschutz-Audit.

Die jeweiligen Prozessschritte sind auch in variierten SCRUM-Prozessen üblicherweise vorzufinden [Di15]. Anders als bei Bente et al. [BEZ19] behält das Team seine Agilität, der Datenschutzverantwortliche muss sich nicht mit Komponenten auseinandersetzen, und die Kosten sinken. Die in der Retrospektive vorgesehene Reflektion spricht zusätzlich das wichtige Ziel, Datenschutz-Bewusstsein in Organisationen zu schaffen [Wr13], an.

4 Nächste Schritte

Der Ansatz wird in aktuellen Projekten weiter verfolgt und verfeinert. So haben wir beispielsweise festgestellt, dass eine Eskalation seitens des Datenschutzverantwortlichen im Projekt hin zur Datenschutz-Compliance-Funktion erfolgen kann, wenn es Unklarheiten gibt, die der Verantwortliche selber nicht klären kann. So könnten auch Auswirkungen einer eventuell auf Projektebene nicht ausreichend vorhandene Fachkunde abgemildert werden. Das agile Team kann unter unserem Ansatz dann mit dokumentierten Annahmen

weiterarbeiten. Gegebenenfalls kann eine Freigabe der Annahmen durch ein Projekt- oder Produkt-Management erfolgen, da in einem solchen Fall die sich aus Unklarheiten ergebenden Risiken über einen Sprint hinaus wachsen. Auch kann dokumentiert werden, wenn das Projektmanagement Einschätzungen des Datenschutz-Verantwortlichen oder der Compliance-Funktion nicht teilt, und wie letztlich entschieden wurde. Solche komplexeren Begutachtungsfälle sowie eine Integration mit weiteren Datenschutz-Prozessen, etwa Datenschutz-Risikobewertungen [Wr13], halten wir für vielversprechende Ansätze.

Literatur

[BEZ19] Bente, S.; Egelhaaf, B.; Zorn, I.: ELSI-Scrum als integrierte ELSI-by-Design-Entwicklungsmethode für Technologieprojekte. In: Mensch und Computer 2019 - Workshopband. Gesellschaft für Informatik e.V., 2019.

[Di15] Diebold, P.; Ostberg, J.-P.; Wagner, S.; Zendler, U.: What Do Practitioners Vary in Using Scrum? In: Agile Processes in Software Engineering and Extreme Programming. Springer International Publishing, Cham, S. 40–51, 2015.

[EP17] Ebert, C.; Paasivaara, M.: Scaling Agile. IEEE Software 34/6, S. 98–103, 2017.

[Fi13] Fitzgerald, B.; Stol, K.-J.; O'Sullivan, R.; O'Brien, D.: Scaling Agile Methods to Regulated Environments: An Industry Case Study. In: Proceedings of the 2013 International Conference on Software Engineering. ICSE '13, IEEE Press, Piscataway, NJ, USA, S. 863–872, 2013, ISBN: 978-1-4673-3076-3.

[He16] Hernández-Serrano, J.; Muñoz, J. L.; Bröring, A.; Esparza, O.; Mikkelsen, L.; Schwarzott, W.; León, O.; Zibuschka, J.: On the Road to Secure and Privacy-Preserving IoT Ecosystems. In: Interoperability and Open-Source Solutions for the Internet of Things. Springer, Cham, S. 107–122, Nov. 2016.

[Li14] Liikkanen, L. A.; Kilpiö, H.; Svan, L.; Hiltunen, M.: Lean UX: the next generation of user-centered agile development? In: Proceedings of the 8th Nordic Conference on Human-Computer Interaction: Fun, Fast, Foundational. NordiCHI '14, ACM Press, S. 1095–1100, 2014.

[Me05] Melton, T.: The Benefits of Lean Manufacturing. Chemical Engineering Research and Design 83/6, S. 662–673, 2005.

[Ra07] Radmacher, M.; Zibuschka, J.; Scherner, T.; Fritsch, L.; Rannenberg, K.: Privatsphärenfreundliche topozentrische Dienste unter Berücksichtigung rechtlicher, technischer und wirtschaftlicher Restriktionen. In: 8. Internationale Tagung Wirtschaftsinformatik 2007 - Band 1. S. 237–254, Feb. 2007.

[Wr13] Wright, D.: Making Privacy Impact Assessment More Effective. The Information Society 29/5, S. 307–315, 2013.

[ZHK19] Zibuschka, J.; Horsch, M.; Kubach, M.: The ENTOURAGE Privacy and Security Reference Architecture for Internet of Things Ecosystems. In: Open Identity Summit 2019. Gesellschaft für Informatik, Bonn, S. 119–130, 2019.

D. Reinhardt, H. Langweg, B.C. Witt, M. Fischer et al. (Hrsg.): Sicherheit 2020,
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Göttingen 2020 129

Usability, Sicherheit und Privatsphäre von risikobasierter Authentifizierung

Stephan Wiefling[1]

Abstract: Risikobasierte Authentifizierung (RBA) ist eine adaptive Sicherheitsmaßnahme zur Stärkung passwortbasierter Authentifizierung. Sie zeichnet Merkmale während des Logins auf und fordert zusätzliche Authentifizierung an, wenn sich Ausprägungen dieser Merkmale signifikant von den bisher bekannten unterscheiden. RBA bietet das Potenzial für gebrauchstauglichere Sicherheit. Bisher jedoch wurde RBA noch nicht ausreichend im Bezug auf Usability, Sicherheit und Privatsphäre untersucht. Dieser Extended Abstract legt das geplante Dissertationsvorhaben zur Erforschung von RBA dar. Innerhalb des Vorhabens konnte bereits eine Grundlagenstudie und eine darauf aufbauende Laborstudie durchgeführt werden. Wir präsentieren erste Ergebnisse dieser Studien und geben einen Ausblick auf weitere Schritte.

Keywords: Passwort; Authentifizierung; Risikobasierte Authentifizierung

1 Einleitung

Passwortbasierte Authentifizierung hat eine lange Tradition [MT79] und ihre Schwachstellen sind seit langer Zeit bekannt. Diese Schwachstellen reichen von Wiederverwendung gleicher Passwörter [FH07] über kurze Längen [DMR10] bis hin zu Phishing [DTH06]. In letzter Zeit stellen Diebstähle großer Benutzernamen-Passwort-Datenbanken und die damit verbundene automatisierte Eingabe dieser Daten bei anderen Onlinediensten (Credential Stuffing) eine immer größere Gefahr für passwortbasierte Authentifizierung dar [Ak19, WKB14]. Ebenso tragen neue intelligente Passwortratemethoden zu immer effizienteren Angriffen auf Online-Passwörter bei [Wa16].

Trotzdem bleiben Passwörter weiterhin die vorherrschende Authentifizierungsmethode bei vielen Onlinediensten. Aus diesem Grund sind Inhaber von Onlinediensten angehalten, ihre Nutzer mit zusätzlichen Maßnahmen vor Angriffen zu schützen. Eine dabei häufig angebotene Maßnahme ist die Zwei-Faktor-Authentifizierung (2FA) [QHD18]. Diese stellt sich unter Nutzern jedoch als unbeliebt heraus. Weniger als 10% aller aktiven Nutzer des Onlinedienstes Google hatten 2FA im Januar 2018 aktiviert [Mi18], obwohl Google diese Technologie seit 2011 aktiv bei ihren Nutzern bewirbt [Sh11]. Ein Grund für die niedrige Akzeptanz könnte darin liegen, dass Nutzer die kontinuierliche Abfrage zweier unterschiedlicher

[1] TH Köln, Data & Application Security Group, Betzdorfer Straße 2, 50679 Köln, Deutschland / Ruhr-Universität Bochum, Mobile Security Group, Universitätsstrasse 150, ID 2 / 127, 44780 Bochum, Deutschland
stephan.wiefling@th-koeln.de

doi:10.18420/sicherheit2020_12

Authentifizierungsfaktoren als zusätzliche Bürde empfinden [Kr15]. Ein alternativer Ansatz zur Stärkung passwortbasierter Authentifizierung ist risikobasierte Authentifizierung (RBA), welche passwortbasierte Authentifikation mit geringer Nutzerinteraktion stärken kann.

2 Risikobasierte Authentifizierung (RBA)

RBA wird häufig in Kombination mit passwortbasierter Authentifizierung eingesetzt [Fr16]. Diese Technologie soll vor Angreifern schützen, welche in Kenntnis der Zugangsdaten (Benutzernamen und Passwort) sind oder diese mit wenigen Versuchen erraten können. Beispiele dafür sind Credential Stuffing [WKB14], Phishing [DTH06] oder Machine-Learning-basiertes gezieltes automatisiertes Erraten von Passwörtern [Wa16].

Bei der Passworteingabe beobachtet RBA Merkmale, die in dem Kontext verfügbar sind. Mögliche Merkmale reichen von Netzwerkinformationen (z.B. IP-Adresse oder IP-basierte Geolocation) über Geräteinformationen (z.B. verwendeter Browser) bis zu biometrischen Eigenschaften (z.B. Tippverhalten). Basierend auf diesen Merkmalen berechnet RBA eine Risikobewertung, welche typischerweise einer der drei Kategorien *niedriges*, *mittleres* und *hohes Risiko* zugeordnet wird. Bei einem niedrigen Risiko (z.B. gleiches Gerät und gleicher Ort wie immer) wird nach dem Absenden der Zugangsdaten der Zugriff gewährt. Bei einem mittleren Risiko (z.B. ungewöhnliches Gerät an einem anderen Ort) wird der Nutzer nach zusätzlicher Authentifizierung gefragt. Eine solche Authentifizierung kann beispielsweise die Verifikation der hinterlegten E-Mail-Adresse sein [WLID19a]. Wird die zusätzliche Authentifizierung erfolgreich erbracht, so wird der Zugriff gewährt. Bei einem hohen Risiko (z.B. IP-Adresse ist für Spam bekannt) wird hingegen der Zugriff verweigert. Diesen Fall sollten Onlinedienste allerdings nur in sehr seltenen Fällen durchführen, um legitime Nutzer nicht ungewollt auszuschließen.

3 Forschungsstand

Die aktuell publizierte Forschung im Bereich RBA beschränkt sich größtenteils auf die Konzepte und Vorstellung neuer RBA-Systeme. Die von den Systemen zur Risikobewertung herangezogenen Merkmale unterscheiden sich dabei grundsätzlich voneinander. So verwendet beispielsweise das RBA-Verfahren von Freeman et al. die IP-Adresse und den User-Agent-String [Fr16]. Bei Steinegger et al. sind es hingegen der Browser-Fingerprint, fehlerhafte Login-Versuche sowie die IP-basierte Geolocation [St16]. Tabelle 1 zeigt eine Auflistung der vielfältigen in der Literatur erwähnten Merkmale, die im Kontext von RBA vorstellbar sind.

Der Einsatz von RBA beschränkt sich aktuell auf große Onlinedienste [WLID19a]. Darüber hinaus empfiehlt das National Institute of Standards and Technology (NIST) in ihren Digital Identity Guidelines RBA als Maßnahme gegen onlinebasiertes Passwortraten [Gr17].

Merkmal	**RBA Referenzen (mit Ausnahme von [AvO16])**	**Informationstiefe zur Unterscheidbarkeit** [AvO16]
IP-Adresse	[Fr16, GOB13, CM12, AuH11, HH14, St16]	Hoch
User-Agent-String	[Fr16, HH14, SPJ15]	Hoch
Sprache	[Fr16, HH14, CM12]	Hoch
Displayauflösung	[DHO17, SPJ15]	Hoch
Login-Zeit	[Fr16, HH14, GOB13, SPJ15, CM12]	Niedrig
Evercookies	[HH14]	Sehr hoch
Canvas-Fingerprinting	[DHO17, Mo12, St16]	Medium
Maus-/Tastenanschlag Dynamik	[Tr12, So19]	- (*Niedrig bei Scrollrad-Fingerprinting*)
Fehlgeschlagene Loginversuche	[HH14, St16]	-
Protokoll-Fingerprinting	-	Hoch
Audioverarbeitung	-	Medium
WebRTC	-	Medium
Zählen von Rechnern hinter NAT	-	Niedrig
Batterie-Information	-	Niedrig
Ad-Blocker-Erkennung	-	Sehr niedrig

Tab. 1: Auswahl möglicher Merkmale, die zur RBA-Risikobewertung herangezogen werden können (angelehnt an [WLID19a]). Alle Merkmale können durch die Benutzung eines Webbrowsers ohne Erweiterungen ermittelt werden.

Folglich nimmt die Bedeutung dieser Technologie immer weiter zu. Allerdings halten bislang die Unternehmen, die RBA einsetzen, ihre Erfahrungen mit RBA vor der Öffentlichkeit zurück. Dies erschwert eine flächendeckende Einführung von RBA. Dabei besitzt diese Technologie auch für kleine und mittelgroße Webseiten das Potenzial, die Sicherheit für ihre Nutzer zu erhöhen.

Die Usability einer neuen Technologie hat einen großen Einfluss auf die Akzeptanz und Verbreitung dieser Technologie [AA18]. Allerdings wurden die Usability-Aspekte von RBA bislang noch nicht erforscht. In gleicher Weise trifft dies auf die Privatsphärenaspekte zu.

4 Forschungsvorhaben

Ziel des Dissertationsvorhabens ist das weitreichende Verständnis von RBA in den dort noch unerforschten Bereichen der Usability, Sicherheit und Privatsphäre. Mit der Forschung an RBA soll eine großflächige Anwendung der Technologie, auch bei kleinen und mittelgroßen Onlinediensten, ermöglicht werden. Zusammengefasst widmet sich das geplante Dissertationsvorhaben den folgenden Fragestellungen:

- Wie muss RBA umgesetzt sein, damit diese Authentifizierung von Nutzern *effektiv*, *effizient* und *zufriedenstellend* [IS17] durchgeführt werden kann?
- Welche Merkmale müssen gesammelt werden, damit RBA die Nutzer in zufriedenstellendem Maße vor Angriffen schützen kann?

- Wie können diese Merkmale gesammelt und ausgewertet werden, um eine möglichst große Privatsphäre bei gleichbleibender Sicherheit von RBA zu erreichen?

5 Erste Ergebnisse

Als Grundlage für weitere Studien wurde der Einsatz von RBA bei acht großen Onlinediensten mithilfe von Black-Box-Tests erforscht [WLID19a]. Dafür wurden 28 virtuelle Identitäten und 224 Nutzeraccounts auf diesen Onlinediensten erstellt. Danach haben wir Nutzerverhalten mithilfe eines Automatisierungsframeworks in menschenähnlicher Weise erzeugt und damit die Onlinedienste auf normales Verhalten trainiert [WGLI19]. Nach 20 Sitzungen auf den Onlinediensten wurden anschließend einzelne Merkmale der virtuellen Identitäten variiert und die Reaktionen der Dienste auf diese Veränderungen getestet. Basierend darauf konnte auf Gestaltung und Funktionsweisen der getesteten RBA-Varianten geschlossen werden und somit der Stand der Technik in Bezug auf RBA ermittelt werden.

Aufbauend auf den Ergebnissen wurde eine Between-Group Laborstudie konzipiert, in welcher die Usability- und Sicherheitswahrnehmungen von RBA untersucht wurden. Diese Wahrnehmungen wurden mit 2FA und rein passwortbasierter Authentifikation (Nur-Passwort) verglichen. Die Ergebnisse der im Einreichungsprozess befindlichen Studie zeigen unter Anderem mit signifikanten Ergebnissen, dass RBA gebrauchstauglicher als 2FA und sicherer als Nur-Passwort angesehen wird.

6 Zusammenfassung

RBA wird in Zukunft immer wichtiger für Webseitenbetreiber werden. Gründe dafür liegen, neben der NIST-Empfehlung, in erhöhten Sicherheitsrisiken durch neue Angriffsmethoden wie Credential Stuffing oder verbesserte Passwortratemethoden. Aus diesem Grund ist die Erforschung für ein ganzumfassendes Verständnis dieser Technologie von großer Bedeutung.

Die ersten Ergebnisse des Dissertationsvorhabens heben bereits die Notwendigkeit für RBA-Forschung hervor. So wurde im Rahmen der Grundlagenstudie eine für lange Zeit existente Sicherheitslücke im RBA-Verfahren von Facebook entdeckt, welche in einem Responsible-Disclosure-Prozess geschlossen wurde [WLID19a, WLID19b]. Auch die im Einreichungsprozess befindlichen Studienergebnisse zeigen Probleme von RBA auf, die die Usability dieser Technologie negativ beeinflussen könnten, sofern diese Probleme nicht entsprechend angegangen wurden.

Trotz allem zeigen die ersten Ergebnisse bereits, dass RBA stärkere Sicherheit bei erhöhter Usability bereitstellen kann. Inwiefern das auch unter Wahrung der Privatsphäre und DSGVO-konform passieren kann, wird der weitere Verlauf des Dissertationsvorhabens zeigen.

Amazon, Facebook, GOG.com, Google, iCloud, LinkedIn, Steam und Twitch

Danksagung Das Projekt URIA (Usability of Risk-based Implicit Authentication) wird durch das Graduiertenkolleg „Human Centered Systems Security“ (NERD.NRW) vom Land Nordrhein-Westfalen gefördert.

Literaturverzeichnis

[AA18] AlHogail, Areej; AlShahrani, Mona: Building Consumer Trust to Improve Internet of Things (IoT) Technology Adoption. In: Advances in Neuroergonomics and Cognitive Engineering, Jgg. 775. Springer International Publishing, Cham, Juni 2018.

[Ak19] Akamai: Credential Stuffing: Attacks and Economies. [state of the internet] / security, 5(Special Media Edition), April 2019.

[AuH11] Akhtar, Nadeem; ul Haq, Farid: Real time online banking fraud detection using location information. In: Computational Intelligence and Information Technology. Springer, 2011.

[AvO16] Alaca, Furkan; van Oorschot, P. C.: Device fingerprinting for augmenting web authentication: classification and analysis of methods. In: ACSAC ’16. ACM Press, 2016.

[CM12] Cser, Andras; Maler, Eve: The Forrester Wave: Risk-Based Authentication, Q1 2012. 2012.

[DHO17] Daud, Nor Izyani; Haron, Galoh Rashidah; Othman, Siti Suriyati Syd: Adaptive authentication: Implementing random canvas fingerprinting as user attributes factor. In: ISCAIE ’17. IEEE, 2017.

[DMR10] Dell’Amico, Matteo; Michiardi, Pietro; Roudier, Yves: Password strength: An empirical analysis. In: INFOCOM ’10. IEEE, 2010.

[DTH06] Dhamija, Rachna; Tygar, J. D.; Hearst, Marti: Why Phishing Works. In: CHI ’06. ACM, April 2006.

[FH07] Florencio, Dinei; Herley, Cormac: A Large-scale Study of Web Password Habits. In: WWW ’07. ACM, New York, NY, USA, Mai 2007.

[Fr16] Freeman, David; Jain, Sakshi; Dürmuth, Markus; Biggio, Battista; Giacinto, Giorgio: Who Are You? A Statistical Approach to Measuring User Authenticity. In: NDSS ’16. Internet Society, Februar 2016.

[GOB13] Golan, Lior; Orad, Amir; Bennett, Naftali: System and method for risk based authentication. Oktober 2013. US Patent 8,572,391.

[Gr17] Grassi, Paul A; Fenton, James L; Newton, Elaine M; Perlner, Ray A; Regenscheid, Andrew R; Burr, William E; Richer, Justin P; Lefkovitz, Naomi B; Danker, Jamie M; Choong, Yee-Yin; Greene, Kristen K; Theofanos, Mary F: Digital identity guidelines: authentication and lifecycle management. Bericht NIST SP 800-63b, National Institute of Standards and Technology, Gaithersburg, MD, Juni 2017.

[HH14] Hurkala, Adam; Hurkala, Jaroslaw: Architecture of Context-Risk-Aware Authentication System for Web Environments. In: ICIEIS ’14. September 2014.

[IS17] ISO/TC 159/SC 4: Ergonomics of human-system interaction. ISO 9241-420:2011, 2017.

[Kr15] Krol, Kat; Philippou, Eleni; De Cristofaro, Emiliano; Sasse, M. Angela: "They brought in the horrible key ring thing!" Analysing the Usability of Two-Factor Authentication in UK Online Banking. USEC '15. Internet Society, Februar 2015.

[Mi18] Milka, Grzergor: Anatomy of Account Takeover. In: Enigma 2018. USENIX Association, Januar 2018.

[Mo12] Molloy, Ian; Dickens, Luke; Morisset, Charles; Cheng, Pau-Chen; Lobo, Jorge; Russo, Alessandra: Risk-based Security Decisions Under Uncertainty. In: CODASPY '12. ACM, Februar 2012.

[MT79] Morris, Robert; Thompson, Ken: Password security: A case history. Communications of the ACM, 22(11), November 1979.

[QHD18] Quermann, Nils; Harbach, Marian; Dürmuth, Markus: The State of User Authentication in the Wild. In: WAY '18. August 2018.

[Sh11] Shah, Nishit: Advanced sign-in security for your Google account. Official Google Blog, Februar 2011. https://googleblog.blogspot.de/2011/02/advanced-sign-in-security-for-your.html.

[So19] Solano, Jesus; Camacho, Luis; Correa, Alejandro; Deiro, Claudio; Vargas, Javier; Ochoa, Martín: Risk-Based Static Authentication in Web Applications with Behavioral Biometrics and Session Context Analytics. In: ACNS '19. Springer International Publishing, 2019.

[SPJ15] Spooren, Jan; Preuveneers, Davy; Joosen, Wouter: Mobile device fingerprinting considered harmful for risk-based authentication. In: EuroSec '15. ACM Press, 2015.

[St16] Steinegger, Roland H.; Deckers, Daniel; Giessler, Pascal; Abeck, Sebastian: Risk-based authenticator for web applications. In: EuroPlop '16. ACM Press, 2016.

[Tr12] Traore, Issa; Woungang, Isaac; Obaidat, Mohammad S.; Nakkabi, Youssef; Lai, Iris: Combining Mouse and Keystroke Dynamics Biometrics for Risk-Based Authentication in Web Environments. In: ICDH '12. IEEE, November 2012.

[Wa16] Wang, Ding; Zhang, Zijian; Wang, Ping; Yan, Jeff; Huang, Xinyi: Targeted Online Password Guessing: An Underestimated Threat. In: CCS '16. ACM, Oktober 2016.

[WGLI19] Wiefling, Stephan; Gruschka, Nils; Lo Iacono, Luigi: Even Turing Should Sometimes Not Be Able To Tell: Mimicking Humanoid Usage Behavior for Exploratory Studies of Online Services. In: NordSec '19. Springer Nature, November 2019. Open Access: https://nbn-resolving.org/urn:nbn:de:hbz:832-epub4-14221.

[WKB14] Wang, Xinran; Kohno, Tadayoshi; Blakley, Bob: Polymorphism as a Defense for Automated Attack of Websites. In: ACNS '14. Springer International Publishing, S. 513–530, 2014.

[WLID19a] Wiefling, Stephan; Lo Iacono, Luigi; Dürmuth, Markus: Is This Really You? An Empirical Study on Risk-Based Authentication Applied in the Wild. In: IFIP SEC '19. Springer International Publishing, Juni 2019. Open Access: https://nbn-resolving.org/urn:nbn:de:hbz:832-epub4-13694.

[WLID19b] Wiefling, Stephan; Lo Iacono, Luigi; Dürmuth, Markus: Risk-based Authentication website. 2019. https://riskbasedauthentication.org.

D. Reinhardt, H. Langweg, B.C. Witt, M. Fischer et al. (Hrsg.): Sicherheit 2020,
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Göttingen 2020 135

Datenschutzgerechte und mehrseitig sichere IT-Plattformen für die medizinische Forschung

Tom Petersen[1]

Abstract:

Die medizinische Forschung ist in vielen Fällen auf die Nutzung von zu Patienten erhobenen Gesundheitsdaten angewiesen. Demgegenüber stehen jedoch die besondere Sensibilität dieser Daten und daraus resultierende Datenschutzanforderungen. Das hier vorgestellte Forschungsvorhaben beschäftigt sich mit dem Entwurf von datenschutzgerechten und sicheren IT-Plattformen für die Erhebung, Speicherung und Bereitstellung von Gesundheitsdaten zu Forschungszwecken, um diesen Zielkonflikt zu lösen. Hierzu werden rechtliche Aspekte, Sicherheitsinteressen beteiligter Akteure und mögliche Architekturen betrachtet sowie technische Maßnahmen vorgestellt, die bei der Erfüllung von Datenschutz- und Sicherheitsanforderungen genutzt werden können.

Keywords: Gesundheitsdaten; Medizinische Forschung; Privacy by Design; DSGVO; Kryptographische Schwellwertschemata; Searchable Encryption

1 Einleitung

In der medizinischen Forschung spielen datenbasierte Verfahren wie u. a. Methoden des maschinellen Lernens für die statistische Beurteilung des Erfolges von Behandlungsmethoden, -geräten oder Medikamenten eine große Rolle [Rü15, OE16]. Für die Forschung relevante medizinische Daten wie beispielsweise Risikofaktoren oder Blutwerte können innerhalb einer medizinischen Einrichtung während der Behandlung erhoben oder anschließend aus der Patientenakte bezogen werden . Die Daten einer einzigen Einrichtung sind jedoch für Forschungszwecke häufig nicht ausreichend, da insbesondere für selten auftretende Krankheiten oder zur Beurteilung regionaler Effekte eine nicht ausreichende Zahl von Fällen vorliegt. Hieraus entsteht der Bedarf, relevante Daten aus vielen Quellen für Forschungszwecke zentral nutzen zu können.

Dem Datenbedarf der medizinischen Forschung und einer zentralen Ablage der benötigten Daten steht jedoch die Sensibilität personenbezogener Gesundheitsdaten gegenüber. Die Veröffentlichung personenbezogener Gesundheitsdaten kann im schlimmsten Fall zu lebenslanger Stigmatisierung oder Benachteiligung Betroffener führen. Die Herausforderung besteht darin, Konzepte und Plattformen für die Datensammlung zu medizinischen

[1] Universität Hamburg, Arbeitsgruppe Sicherheit in Verteilten Systemen, Vogt-Kölln-Straße 30, 22307 Hamburg, Deutschland petersen@informatik.uni-hamburg.de
Die notwendige Patienteneinwilligung und Zweckbindung erhobener Daten seien an dieser Stelle nicht betrachtet.

doi:10.18420/sicherheit2020_13

Forschungszwecken zu erarbeiten, die diese gegensätzlichen Sicherheitsinteressen beachten und vermitteln [Be17].

Das hier vorgestellte Forschungsvorhaben befasst sich mit verschiedenen bei dem Entwurf von Plattformen für die medizinische Forschung relevanten Schwerpunkten, die im Folgenden vorgestellt werden: Abschnitt 2 beschäftigt sich mit rechtlichen Aspekten und Anforderungen. Abschnitt 3 beschreibt (teilweise gegensätzliche) Sicherheitsinteressen, die durch die verschiedenen beteiligten Akteure auftreten. Abschnitt 4 stellt beispielhaft einige Architekturen, die für Plattformen für die medizinische Plattformen genutzt werden können, und Bewertungskriterien für diese vor. Abschnitt 5 geht anschließend auf ergänzende Schutzmaßnahmen ein, die zur datenschutzgerechten Gestaltung bestimmter PLattformaspekte genutzt werden sollen.

2 Rechtliche Betrachtung

Das Spannungsfeld zwischen Datensensibilität und Forschungsnutzen wird bereits in der EU-Datenschutzgrundverordnung (DSGVO) deutlich. In Artikel 9 wird die Verarbeitung von medizinischen Daten ohne ausdrückliche Einwilligung der betreffenden Person grundsätzlich untersagt. Der Einschluss von Patientendaten in einem medizinischen Register kann daher im Normalfall nur durch die explizite Einwilligung des Patienten erfolgen [Po08]. Die Erwägungsgründe 52, 53 und 157 der DSGVO betonen jedoch auch die Wichtigkeit dieser Daten für die im öffentlichen Interesse liegende wissenschaftliche Forschung.

Weitere Artikel der DSGVO bilden einen rechtlichen Rahmen für das beschriebene Forschungsvorhaben, beispielsweise: Artikel 5 schreibt u.a. das Prinzip der Datenminimierung und die Sicherheit der Datenverarbeitung durch geeignete technische und organisatorische Maßnahmen vor. In den Artikeln 12 bis 23 werden die Rechte eines Betroffenen dargestellt. Artikel 25 stellt die Forderung nach Datenschutz durch Technikgestaltung und datenschutzfreundliche Voreinstellungen (Privacy by Design) auf.

Für das Forschungsvorhaben ergeben sich folgende Fragen: Wie können rechtliche Anforderungen wie die Gewährleistung von Betroffenenrechten in einer Plattform für die medizinische Forschung umgesetzt werden? Welche Prinzipien und konkreten technischen Maßnahmen können wie eingesetzt werden, um die Forderung nach Datenschutz durch Technikgestaltung zu erfüllen?

3 Mehrseitige Sicherheit

Dem Spannungsfeld gegensätzlicher Sicherheitsinteressen widmet sich aus technischer Sicht die mehrseitige Sicherheit [RPM96], indem sie versucht, technische Lösungsmöglichkeiten zum Ausgleich dieser möglicherweise gegensätzlichen Sicherheitsinteressen zu entwickeln und nutzbar zu machen.

Im Kontext von Plattformen für die medizinische Forschung müssen hier die Sicherheitsinteressen verschiedener Akteure betrachtet werden: Zu *Patienten* werden auf Basis einer Einwilligung sensible Gesundheitsdaten erfasst, deren Vertraulichkeit gegenüber anderen Akteuren soweit möglich erhalten werden muss. *Medizinisches Personal* in teilnehmenden Einrichtungen verarbeitet die Daten. Der Mitarbeiterzugriff auf Daten sollte protokolliert werden, um absichtliches oder unabsichtliches Fehlverhalten aufdecken zu können. Auf der anderen Seite können diese Daten die Überwachung von Mitarbeitern ermöglichen, was unter dem Aspekt des Mitarbeiterdatenschutzes weitestmöglich verhindert werden sollte. Der *Betreiber* einer Plattform hat ein Interesse an der Qualitätssicherung der erfassten Gesundheitsdaten, um eine hohe Reputation des Datensatzes und darauf basierenden Forschungsarbeiten zu gewährleisten. *Forscher* benötigen Zugriff auf für ihre Forschungsfragen relevante Ausschnitte der erfassten Daten. *Entwickler* und *Administratoren* der Plattform sollten abgesehen von für die Wartung und die Weiterentwicklung der Plattform unerlässlichen Daten keinen Zugriff auf erfasste Daten erhalten.

Für das Forschungsvorhaben ergeben sich folgende Fragen: Welche möglicherweise gegensätzlichen Sicherheitsinteressen haben die Beteiligten? Mithilfe welcher technischen Maßnahmen können diese gewahrt oder vermittelt werden?

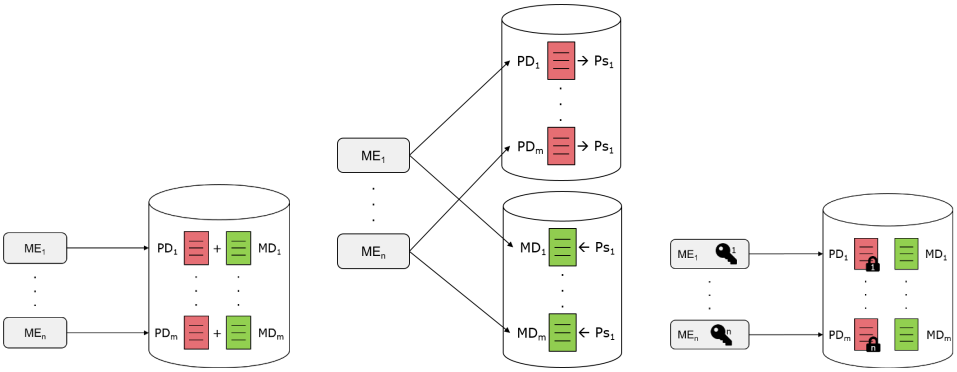
4 Architekturen für IT-Plattformen

In diesem Teil des Forschungsvorhabens sollen verschiedene Architekturen für Plattformen entworfen und evaluiert werden. Die Ausgangssituation ist dabei die folgende: In medizinischen Einrichtungen werden personenbezogene und medizinische Daten zu Patienten erfasst. Die Plattform soll es Forschern ermöglichen, zentral Zugriff auf eine für sie relevante Menge von medizinischen Daten zu erhalten. Zu diesem Zweck entworfene Architekturen beschreiben, wo welche Daten in welcher Form verarbeitet, gespeichert und bereitgestellt werden. Denkbar ist der Einsatz verschiedener Techniken, unter anderem die getrennte Verarbeitung personenbezogener und medizinischer Daten , eine Trennung der Datenhoheit (*informationelle Gewaltenteilung*), die Verwendung von Verschlüsselungsverfahren und die Nutzung von Pseudonymen zur Verknüpfung von Daten.

Die so entworfenen Architekturen können auf bestimmte kontextrelevante Eigenschaften überprüft werden, beispielsweise: Auf welche Daten erhalten beteiligte Akteure Zugriff und verhindert die Architektur nicht erforderlichen Datenzugriff? Ermöglicht die Plattform die Erfassung und Verknüpfung von medizinischen Daten eines Patienten zu mehreren Zeitpunkten (*zeitlicher Längsschnitt*) und in verschiedenen medizinischen Einrichtungen?

Beispielhaft seien hier drei mögliche Architekturen skizziert: Abbildung 1a zeigt eine Architektur, in der medizinische und personenbezogene Daten zentral an einer Stelle im Klartext gespeichert werden. Die Architektur in Abbildung 1b zeigt die getrennte

Die Reidentifikation von Patienten durch medizinische Daten muss im Rahmen des Forschungsvorhabens betrachtet werden, soll in diesem Abschnitt jedoch vorerst vernachlässigt werden.



(a) Zentrale Speicherung aller Daten

(b) Informationelle Gewaltenteilung

(c) Verschlüsselung der personenbezogenen Daten

Abb. 1: Darstellung von drei Beispielarchitekturen

Speicherung von medizinischen und personenbezogenen Daten und die Verknüpfung der Daten mittels Pseudonymen und damit eine Form der informationellen Gewaltenteilung. Abbildung 1c stellt eine Architektur dar, in der die personenbezogenen Daten zentral ausschließlich in verschlüsselter Form gespeichert werden, wobei die notwendigen Schlüssel nur in der entsprechenden medizinischen Einrichtung vorliegen.

Für das Forschungsvorhaben ergeben sich folgende Fragen: Welche sinnvollen Architekturen für Plattformen existieren und welche Eigenschaften erfüllen sie? Wie lassen sich die rechtlichen Anforderungen und Sicherheitsinteressen in entsprechenden Architekturen umsetzen (auch in Verbindung mit entsprechenden technischen Maßnahmen, siehe Abschnitt 5)?

5 Maßnahmen

Methoden aus den Bereichen der Kryptographie und der Privacy Enhancing Technologies (PETs) können dabei helfen, datenschutzgerechte und mehrseitig sichere Plattformen für die medizinische Forschung zu entwickeln. Nach aktuellem Stand sollen drei konkrete Bereiche in diesem Forschungsvorhaben untersucht werden:

5.1 Die Nutzung von kryptographischen Schwellwertschemata zur Durchsetzung des Mehraugenprinzips

Das Mehraugenprinzip ist eine Kontrollmaßnahme, bei der kritische Entscheidungen oder Handlungen nur durch mehrere Personen gemeinsam durchgeführt werden dürfen, um Missbrauch zu erschweren. Häufig wird das Prinzip nur durch organisatorische Maßnahmen umgesetzt. Kryptographische Schwellwertschemata wie Threshold Decryption [DF90] oder

Threshold Signatures [Ge96] erlauben die Durchführung kryptographischer Operationen nur durch die Kooperation mehrerer Personen, die jeweils im Besitz eines Teilschlüssels für die entsprechende Operation sind. Sie können dazu genutzt werden, das Mehraugenprinzip auch kryptographisch zu erzwingen. Für den Einsatz in Plattformen für die medizinische Forschung sind insbesondere praxisrelevante Fragen der Schlüsselverwaltung und möglicher Zugriffsstrukturen[4] zu betrachten, die in bisherigen Veröffentlichungen selten betrachtet werden.

Für das Forschungsvorhaben ergeben sich folgende Fragen: An welchen Stellen der Plattformen kann das Mehraugenprinzip zur Umsetzung von Sicherheits- oder Datenschutzanforderungen genutzt werden? Welche Probleme ergeben sich beim praktischen Einsatz kryptographischer Schwellwertschemata und wie können diese gelöst werden?

5.2 Die Anwendbarkeit von Searchable-Encryption-Schemata

Aufgrund der Sensibilität von Gesundheitsdaten können verschlüsselte Datenbanken eine sinnvolle Maßnahme zum Schutz dieser Daten darstellen (vergleiche Abbildung 1c). Dies verhindert jedoch das direkte Suchen in den Datensätzen bzw. erfordert die vorhergehende Entschlüsselung aller Daten. Ansätze aus dem Bereich der Searchable Encryption [SWP00] können hier als möglicherweise performante Alternative zu diesem Vorgehen in Betracht gezogen werden.

Für das Forschungsvorhaben ergeben sich folgende Fragen: Welche Searchable-Encryption-Schemata bieten sich für den Anwendungskontext an und welche Arten von Suchanfragen erlauben sie? Erlaubt der Einsatz dieser Schemata Rückschlüsse[5] auf die unterliegenden Daten?

5.3 Ansätze zur Anonymitätsmessung von veröffentlichen Gesundheitsdatensätzen

Sollen medizinische Daten für die Forschung verwendet und ggf. veröffentlicht werden, so muss durch Anonymisierung der Daten sichergestellt werden, dass Patienten nicht durch die Daten reidentifiziert werden können. Hierzu können Maße zur Bestimmung der Anonymität wie k-Anonymität [Sw02] oder Differential Privacy [Dw06] genutzt werden.

Im Forschungsvorhaben soll geklärt werden, welche Verfahren sich zur Messung der Anonymität im Kontext von Gesundheitsdaten eignen, welche Randbedingungen es für Ihren Einsatz gibt und wie sie praktisch umgesetzt werden können. Ein besonderer Fokus soll hier auf den Zusammenhang zwischen Anonymisierung und der dezentralen Datensammlung in verschiedenen Krankenhäusern gelegt werden.

[4] Das Mengensystem aller möglichen Mengen von Teilnehmern, die die kryptographische Operation gemeinsam durchführen können, wird Zugriffstruktur genannt.

[5] Es wurde beispielsweise gezeigt, dass statistische Betrachtungen dazu genutzt werden können, mithilfe von Range Queries verschlüsselte Daten zu rekonstruieren [Gr19].

6 Fazit

Mit dem hier vorgestellten Forschungsvorhaben soll zwischen den besonderen Schutzanforderungen von Gesundheitsdaten auf der einen Seite und der Relevanz von IT-Plattformen für den Fortschritt in der medizinischen Forschung auf der anderen Seite vermittelt werden.

Durch den Einsatz geeigneter Architekturen und Maßnahmen wird versucht, den Forderungen des Artikel 25 der DSGVO (Datenschutz durch Technikgestaltung und durch datenschutzfreundliche Voreinstellungen) im Kontext von IT-Plattformen für die medizinische Forschung nachzukommen.

Literaturverzeichnis

[Be17] Behrendt, Christian-Alexander; Pridöhl, Henning; Schaar, Katrin; Federrath, Hannes; Debus, Eike Sebastian: Klinische Register im 21. Jahrhundert. Der Chirurg, 88(11):944–949, 2017.

[DF90] Desmedt, Yvo; Frankel, Yair: Threshold cryptosystems. In (Brassard, Gilles, Hrsg.): Advances in Cryptology - CRYPTO '89. Jgg. 435 in Lecture Notes in Computer Science. Springer, S. 307–315, 1990.

[Dw06] Dwork, Cynthia; McSherry, Frank; Nissim, Kobbi; Smith, Adam: Calibrating Noise to Sensitivity in Private Data Analysis. In (Halevi, Shai; Rabin, Tal, Hrsg.): Theory of Cryptography. Springer Berlin Heidelberg, Berlin, Heidelberg, S. 265–284, 2006.

[Ge96] Gennaro, Rosario; Jarecki, Stanisław; Krawczyk, Hugo; Rabin, Tal: Robust threshold DSS signatures. In (Maurer, Ueli, Hrsg.): Advances in Cryptology - EUROCRYPT '96. Jgg. 1070 in Lecture Notes in Computer Science. Springer, S. 354–371, 1996.

[Gr19] Grubbs, P.; Lacharité, M.; Minaud, B.; Paterson, K. G.: Learning to Reconstruct: Statistical Learning Theory and Encrypted Database Attacks. In: IEEE Symposium on Security and Privacy (S&P) 2019. 2019.

[OE16] Obermeyer, Ziad; Emanuel, Ezekiel J: Predicting the future—big data, machine learning, and clinical medicine. The New England journal of medicine, 375(13):1216, 2016.

[Po08] Pommerening, K.; Debling, D.; Kaatsch, P.; Blettner, M.: Register zu seltenen Krankheiten. Bundesgesundheitsblatt - Gesundheitsforschung - Gesundheitsschutz, 51(5):491–499, May 2008.

[RPM96] Rannenberg, Kai; Pfitzmann, Andreas; Müller, Günter: Sicherheit, insbesondere mehrseitige IT-Sicherheit. it+ti - Informationstechnik und Technische Informatik, 38(4):7–10, 1996.

[Rü15] Rüping, Stefan: Big Data in Medizin und Gesundheitswesen. Bundesgesundheitsblatt - Gesundheitsforschung - Gesundheitsschutz, 58(8):794–798, Aug 2015.

[Sw02] Sweeney, Latanya: k-anonymity: A model for protecting privacy. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 10(05):557–570, 2002.

[SWP00] Song, Dawn Xiaoding; Wagner, David; Perrig, Adrian: Practical techniques for searches on encrypted data. In: Proceedings of the 2000 IEEE Symposium on Security and Privacy. IEEE, S. 44–55, 2000.

Do Privacy Concerns Prevent Employees' Acceptance of Smart Wearables and Collaborative Robots?

Alexander Richter[1]

Abstract: During the digitization of workplaces, companies are increasingly using smart wearables as well as collaborative robots. This technological progress can contribute to higher productivity and efficiency in manufacturing processes, as they assist employees in carrying out their work. This changes the way employees interact and collaborate with their working environment and robots. When companies utilize smart wearables and collaborative robots in their processes, employees are exposed to various privacy issues, which may lead to privacy concerns and may reduce the acceptance of such devices and robots. Thus, the presented PhD research project aims to understand the employees' privacy concerns and address them.

Keywords: Privacy; Employees' Acceptance; Smart Wearables; Collaborative Robots; Smart Workplaces

1 Introduction

Employees' work is changing in factories and other industries. Companies digitize manufacturing processes to optimize workplaces and operations to achieve higher productivity and greater efficiency. For this, smart wearables have the potential to contribute in increasing productivity. Therefore, an increasing number of companies are equipping their employees with smart wearables [Sc16] to improve workers' productivity [Sp13; We15], health [Go17; Li14], and safety [CHL17]. Another essential aspect in industrial processes is human-robot collaboration [Ro17; SM17], as humans and robots can collaborate in an increasing number of tasks due to a new generation of robots and sensors [Ro17; SM17]. However, to facilitate human-robot collaboration, the separation of their workspaces need to be eliminated [Ro16; SM17]. Therefore, different sensors to enhance employees' safety must be embedded in collaborative robots.

However, the potential benefits which entail from such devices and robots are always offset by risks that may affect the employees' privacy. For example, these risks arise from wearing such devices or the interaction with robots by the respective employee due to the possibilities to collect employees' data with the devices or robots. Various authors have already shown that the use of different sensors and devices enhance the possibility to endanger users' privacy [Sh15; WLR15]. Examples include reading sensor data, such as the gyroscope or the

[1] Institute of Computer Science, University of Göttingen, Goldschmidtstr. 7, 37077 Göttingen, Germany
richter@cs.uni-goettingen.de

doi:10.18420/sicherheit2020_14

acceleration sensor. These data make it possible to determine, i.e., users' physical activities. Equal potential risks are also able to arise in the corporate context. For example, this can be seen in the scandal of the Tesco company, where employees were equipped with digital wristbands that allowed managers to find out how much the employees worked [AF13].

Both technologies are components of recent and future digitization of workplaces, as they can enhance productivity and employees' efficiency. Due to the embedded sensors, employers can collect, analyze, and draw inferences about the employees, especially, when they use them for entire shifts. Moreover, combining several sensors' data might give employers more insights regarding employees' sensitive data, for instance employees' health, job performance, etc. Therefore, this PhD research project examines the discrepancies between employers and employees, address them and potentially contribute to increase the acceptance of digitized processes. Thus, the main goal of the project is to first analyze whether the employees' privacy concerns influence their acceptance of such devices. Moreover, it aims to analyze whether increasing the data collection transparency control mechanisms, can improve employees' acceptance to mitigate such employees' concerns.

2 Related Work

Previous research can be classified into three categories: (1) employees' acceptance, (2) privacy concerns, and (3) proposed solutions. The first category includes employees' acceptance surveys conducted in a corporate context, such as [CHL17; Ja19; LHZ19]. Choi et al. identified different influence factors, like perceived usefulness, social influence, and perceived privacy risks, which have an impact on the adoption of smart wearables for occupational safety and health management. Beside employee beliefs, employees' acceptance can also be affected by job position in a company or experience with such devices [CHL17; Ja19]. Existing work is mostly based on a specific use case. Jacobs et al. investigated factors that are related to the organizational settings, the individual employee, and the purpose or use case at the workplace.

In the second category, privacy concerns are identified and classified in different ways in existing works. These privacy concerns are closely related to the embedded sensors with the ability to sense, collect, and store data [MC15] and increase with a physical and temporal context [Ra11]. Furthermore, users are not able to understand potential threats about collected data about behavior disclosures and context from measurements by sensors, unless these are their own data [Ra11]. Besides, previous work mentioned general fears from employees in the context of workplace environments. These include the fear of being under surveillance or tracked by the employers [CHL17; DNC18; SSC18] or the risk that the devices record sensitive information [CHL17; DNC18]. This, especially with regard to surveillance and monitoring, may have a negative impact on job satisfaction and also in the level of employees' stress and may lead to a deterioration in productivity [Me03; TLA18].

For the latter category, different authors propose rules relevant to workplace surveillance [Sa06] or offer recommendations to maximize the positive effects of electronic performance monitoring and to minimize negative ones [TLA18]. These rules include several points such as informing the employees about the data that are collected or accessed as well as how employees can access and correct the information [Sa06; We15]. Thus, employers should be transparent about monitoring processes and use the insights for learning and developing rather than for preventing unwanted employees' behavior. Moreover, employers should monitor only work-related behaviors [TLA18]. Furthermore, employers must take reasonable measures to protect information from misuse, loss, unauthorized access, modification or disclosure [Sa06]. However, these are only general recommendations for employers. A general approach for employers to give employees the ability to gain access to gathered data to comply with the GDPR and thus to enhance employees' privacy is hence still missing.

To the best of our knowledge, there is no previous work that focuses on the following two issues: (1) analyzing employees' privacy concerns arising by the adoption of smart wearables or the collaboration with robots, based on their knowledge regarding possibilities of sensor readings as well as (2) developing an approach allowing employees to get more transparency and control over the collected data by wearables and robots especially w.r.t. the threats related to embedded sensors in smart workplaces. Therefore, our entire research work focuses on the above-mentioned issues.

3 Research Questions and Methodology

The basis of the PhD research project relies on analyzing existing research papers, which address the key topics of smart workplaces, smart wearables, collaborative robots, control, transparency, and minimization of data as well as various technology acceptance models in the context of privacy.

In this work, we will analyze the impact of employees' privacy concerns on their acceptance to use smart wearables and collaborative robots as basis for the development of an approach to protect employees' privacy. In more detail, we will consider the following research questions.

Which privacy risks prevent the use of smart wearables or the collaboration with robots? — We will start by conducting, a structured literature review. Likewise, some in-depth case studies with helpful industry partners shall be conducted to get a closer look into processes with such devices and their implementation and use in companies. Core issues are risks and threats that may arise from these technologies and thus affect individuals' privacy. For a better understanding of existing privacy risks and threats, we will analyze and identify the general threats and risks of such devices, and the included sensors. Likewise, it includes sensors or techniques that threaten individuals' privacy.

Which employees' level of knowledge regarding privacy risks result in acceptance problems? — Regarding the previous insights, privacy concerns arising from employees' level of knowledge (knowledge or ignorance), need to be examined. Knowledge implies that employees are aware of risks for their privacy, which can result in the rejection of new technologies since they can understand the technology's data processing and occurring consequences. In comparison, ignorance implies that those rejection results from the fear based on a lack of knowledge about these potential risks towards their privacy, as they are not able to grasp how data collection or processing works or which consequences results from this data, for instance. Thus, these are two antagonistic causes from which privacy concerns may arise. For this purpose, a qualitative and quantitative survey shall contribute to identifying these concerns. Therefore, a first targeted and direct semi-structured interview with employees will be held and provide the first insights on perceived privacy risks and threats. Based on the qualitative interviews, a quantitative survey will be conducted that verifies the insights of the interviews and will confirm or reject their significance. For this purpose, the participants are presented with various benefit and risk scenarios, for instance. From the conducted surveys, an analysis of the ensuing employees' acceptance problems is required about the use of those devices in the company context. For this purpose, the technology acceptance models such as Technology Acceptance Model (TAM) or Unified Theory of Acceptance and Use of Technology (UTAUT) will be analyzed and applied to the research problem.

Which information or conditions influence employees' acceptance? — By means of a survey, information and conditions shall be identified, which may positively influence the employees with respect to the adoption and use of smart wearables or the collaboration with robots. For this, it is necessary to verify, whether the control or transparency of the collected employees' data as well as the data minimization, e.g., by pseudonymization affects the deployment of these devices mentioned above. Therefore, it must be clarified to what extent to control, transparency, and data minimization are understandable in order to help the employers to respect the GDPR compliance and implement mechanisms to improve employees' privacy.

How do solutions need to be implemented in companies to ensure and enhance employees' privacy? — Based on the results obtained, an approach shall be formulated with the aim to improve employees' privacy. This can be achieved through control mechanisms and/or transparency of processes by the management but also through systems, which have already required to implement regarding the GDPR. However, companies need effective approaches to fulfill the requirements. Thereby, the employees' acceptance to use smart products could be enhanced. Afterwards, an evaluation of the proposed solution would take place by means of user studies.

4 Summary and Expected Contributions

This PhD research project aims to contribute in designing a model, e.g., an interface, that will enable employees to gain more transparency, access, and control over their personal

information generated within the company, especially in presence of smart devices and collaborative robots. For companies to benefit from the above-mentioned advantages of the digitization of workplaces, our proposed approach could contribute to reduce employee's privacy concerns and consequently improve their acceptance.

References

[AF13] Applin, S. A.; Fischer, M. D.: Watching Me, Watching You.(Process Surveillance and Agency in the Workplace). In: Proc. of the 2013 IEEE International Symposium on Technology and Society (ISTAS): Social Implications of Wearable Computing and Augmediated Reality in Everyday Life. Pp. 268–275, 2013.

[CHL17] Choi, B.; Hwang, S.; Lee, S. H.: What Drives Construction Workers' Acceptance of Wearable Technologies in the Workplace?: Indoor Localization and Wearable Health Devices for Occupational Safety and Health. Automation in Construction 84/1, pp. 31–41, 2017.

[DNC18] Datta, P.; Namin, A. S.; Chatterjee, M.: A Survey of Privacy Concerns in Wearable Devices. In: Proc. of the IEEE International Conference on Big Data (Big Data). Pp. 4549–4553, 2018.

[Go17] Gorm, N.: Personal Health Tracking Technologies in Practice. In: Companion of the 20th ACM Conference on Computer Supported Cooperative Work and Social Computing (CSCW). Pp. 69–72, 2017.

[Ja19] Jacobs, J. V.; Hettinger, L. J.; Huang, Y.-H.; Jeffries, S.; Lesch, M. F.; Simmons, L. A.; Verma, S. K.; Willetts, J. L.: Employee Acceptance of Wearable Technology in the Workplace. Applied Ergonomics 78/1, pp. 148–156, 2019.

[LHZ19] Lotz, V.; Himmel, S.; Ziefle, M.: You're My Mate–Acceptance Factors for Human-Robot Collaboration in Industry. In: Proc. of the International Conference on Competitive Manufacturing (COMA). Pp. 405–411, 2019.

[Li14] Lingg, E.; Leone, G.; Spaulding, K.; B'Far, R.: Cardea: Cloud Based Employee Health and Wellness Integrated Wellness Application with a Wearable Device and the HCM Data Store. In: Proc. of the 1st IEEE World Forum on Internet of Things (WF-IoT). Pp. 265–270, 2014.

[MC15] Motti, V. G.; Caine, K.: Users' Privacy Concerns About Wearables. In: Proc. of the 18th International Conference on Financial Cryptography and Data Security (FC). Springer, pp. 231–244, 2015.

[Me03] Meyers, N.: Employee Privacy in the Electronic Workplace: Current Issues for IT Professionals. In: Proc. of the 14th Australasian Conference on Information Systems (ACIS). Pp. 72–81, 2003.

[Ra11] Raij, A.; Ghosh, A.; Kumar, S.; Srivastava, M.: Privacy Risks Emerging from the Adoption of Innocuous Wearable Sensors in the Mobile Environment. In: Proc. of the 29th ACM Conference on Human Factors in Computing Systems (SIGCHI). Pp. 11–20, 2011.

[Ro16] Romero, D.; Stahre, J.; Wuest, T.; Noran, O.; Bernus, P.; Fasth Fast-Berglund, Å.; Gorecky, D.: Towards an Operator 4.0 Typology: A Human-Centric Perspective on the Fourth Industrial Revolution Technologies. In: Proc. of the 46th International Conference on Computers and Industrial Engineering (CIE). Pp. 1–11, 2016.

[Ro17] Robla-Gómez, S.; Becerra, V. M.; Llata, J. R.; Gonzalez-Sarabia, E.; Torre-Ferrero, C.; Perez-Oria, J.: Working Together: A Review on Safe Human-Robot Collaboration in Industrial Environments. IEEE Access 5/1, pp. 26754–26773, 2017.

[Sa06] Sandy, G. A.: Workplace Privacy and Surveillance: A Matter of Distributive Justice. In: Proc. of the 17th Australasian Conference on Information Systems (ACIS). Pp. 69–79, 2006.

[Sc16] Schellewald, V.; Weber, B.; Ellegast, R.; Friemert, D.; Hartmann, U.: Einsatz von Wearables zur Erfassung der körperlichen Aktivität am Arbeitsplatz. DGUV Forum 11/1, pp. 36–37, 2016.

[Sh15] Shoaib, M.; Bosch, S.; Scholten, H.; Havinga, P. J.; Incel, O. D.: Towards Detection of Bad Habits by Fusing Smartphone and Smartwatch Sensors. In: Proc. of the 13th IEEE International Conference on Pervasive Computing and Communication Workshops (PerCom Workshops). Pp. 591–596, 2015.

[SM17] Steil, J. J.; Maier, G. W.: Robots in the Digitalized Workplace. The Wiley Blackwell Handbook of the Psychology of the Internet at Work/, pp. 403–422, 2017.

[Sp13] Spath, D.; Weisbecker, A.; Peissner, M.; Hipp, C.: Potenziale der Mensch-Technik Interaktion für die effiziente und vernetzte Produktion von morgen. Fraunhofer-Verlag, Stuttgart, 2013, ISBN: 978-3-839-60563-9.

[SSC18] Schall, M. C. J.; Sesek, R. F.; Cavuoto, L. A.: Barriers to the Adoption of Wearable Sensors in the Workplace: A Survey of Occupational Safety and Health Professionals. Human Factors 60/3, pp. 351–362, 2018.

[TLA18] Tomczak, D. L.; Lanzo, L. A.; Aguinis, H.: Evidence-based Recommendations for Employee Performance Monitoring. Business Horizons 61/2, pp. 251–259, 2018.

[We15] Weston, M.: Wearable Surveillance – a Step Too Far? Strategic HR Review 14/6, pp. 214–219, 2015.

[WLR15] Wang, H.; Lai, T. T.-T.; Roy Choudhury, R.: Mole: Motion Leaks Through Smartwatch Sensors. In: Proc. of the 21st ACM Annual International Conference on Mobile Computing and Networking. Pp. 155–166, 2015.

Hierarchical Distributed Consensus for Smart Grids

Marius Stübs[1]

Abstract: Reaching consensus in distributed systems is a topic with a long record of suggestions and discussions. One instance of such a distributed system is the emerging Internet-of-Energy: Thousands of Smart Grid service providers orchestrate a multitude of generators, consumers and storage capacity to keep the balance between supply and demand of electrical power. Centralized decision making struggles to fulfill recent requirements regarding robustness against communication outages. In the last decades, consensus research made great leaps towards resilience, elaborating distributed approaches like Paxos and many successor concepts, but has still research gaps in the direction of scalability and dynamicality of node participation. Our work builds upon the results of this recent research while taking a more topologically-driven perspective. We discuss two important innovations towards applicability in future Micro Grids: Clusters are formed to achieve parallelized consensus, acknowledging the grid topology and inter-dependability of Nano Grids. We describe a hierarchical scheme, where aggregation of the locally achieved consensus forms a higher level consensus. This way, we achieve loose coupling combined with partial order of events, implementing a hierarchically distributed global consensus.

Keywords: Smart Grids; Distributed Consensus; Clustering

1 Introduction

Power management in Smart Grids is developing from simple distribution grids towards a decentralized system with multiple domains of responsibility. One emerging challenge is the orchestration of a multitude of generators, so-called distributed energy resources (DER) and consumers, or intelligent energy devices (IED). The scheduling of power generation and consumption can either be represented as a vector of per-resource schedules [SPK19], or in an agent-based manner by peer-to-peer agreements, to pairwise match the schedules of DERs and IEDs [SIB20]. Volatile DERs such as PV panels increase the system's complexity, while mobile consumers such as electric vehicles lead to a more dynamic system. To avoid conflicts between the participating nodes, consensus on the partial order of events can ensure that the global system state is consistent. Including locality in the sense of electrical distance between DERs and IEDs [St19] is the logical continuation towards a mature distributed Smart Grid control scheme.

Micro Grids represent independent sections of the distribution grid. The concept of a self-organizing Micro Grid, where nodes can dynamically join and leave, poses important

[1] Universität Hamburg, Sicherheit in verteilten Systemen, Vogt-Kölln-Straße 30, 22527 Hamburg, Deutschland
stuebs@informatik.uni-hamburg.de

doi:10.18420/sicherheit2020_15

research topics, especially regarding security aspects when establishing consensus in such open networks [St18]. Refraining from centralized solutions for the coordination of power grids can increase resilience, but it does not solve all existing problems and also comes with additional requirements. From the distributed systems perspective, new challenges include fairness, as well as the three properties Consistency (C), liveness or availability (A) and partition-tolerance (P) from the CAP-Theorem [Br00]. Liveness refers to the requirement, that distributed algorithms are vulnerable to deadlocks and unexpected system states. These may be hard to detect and resolve, therefore a formal verification of the liveness is essential. Consistency describes the property that all claims from the system have to be valid and up-to-date. Together with partition-tolerance, which means that the system would continue working despite disconnected nodes, these requirements build the triangle of the CAP-theorem. It states, that it is not feasible to have all three properties C, A and P: If a partition occurs, any claim from the system might possibly be incorrect, because not all necessary information is available. It follows, that any distributed system has to prioritize between liveness (or availability), consistency and partition-tolerance [Br00].
Fairness refers to a system where every actor is eventually served, even if equal options are available. Applied to the Smart Grid context this means that when considering two equally suitable consumers, it is not acceptable to always skip the same [BFT11].

Future Smart Grids will not only be more decentralized, but are also expected to be more open to small business operators and flexible contracting of prosumers to contribute to dynamic Micro Grids. The Smart Meter Gateway roll-out increases the amount of IEDs for the grid management [URZ15]. This leads to challenges regarding:
Scalability. Well proven optimization software may turn out to be too slow and static to incorporate thousands of participants.
Blurred Borders between Transmission Grid and Distribution Grid. Traditionally, power is injected into the transmission grid and then distributed in a top-down manner via the distribution grid. This is now contradicted by the ubiquitous introduction of prosumers that both inject and consume power.

To address these challenges, our research elaborates on solutions to divide the Micro Grid into so-called Nano Grids: independent clusters that need to find a consensus regarding their balance of power demand and supply. The proposed consensus protocol aims to bridge the gap between the physical grid topology and the often-times grid-agnostic application layer, as depicted in Figure 1. The overall research topic therefore is, how such a distributed consensus can be organized.

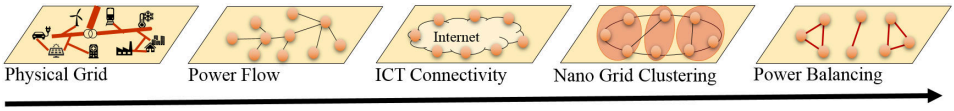


Fig. 1: The architectural layers from the physical power lines up to actual Smart Grid services like power balancing.

2 Related Work

Paxos [La98] requires all nodes to be included in the vote and a majority of them to answer and agree. On the other hand, Egalitarian Paxos (EPaxos) relaxes the guarantees, acknowledging that there are many possible orderings that meet the requirements of distributed computing: Commutative requests can be processed out-of-order [MAK13].

RAFT decouples leader-election and state-machine-replication to achieve an easy-to-understand solution for distributed consensus with a strong focus on leadership [OO14]. In Flexible Paxos (FPaxos), the interdependency between the two stages of leader election and log replication is investigated separately, allowing to further reduce quorum sizes [HMS16].

We identify a research gap when it comes to distributed consensus with the focus on topology requirements and awareness for the locality of event orderings. One important approach in this direction is Canopus [RWK17], which achieves some of the postulated requirements: Distributed Hierarchical Consensus. However in contrast to Canopus, we favor liveness, availability and failsafe partition-tolerance over global consistency [Br00], in order to maintain stable micro grids in case of grid separation.

3 Hierarchical Aggregated Consensus

We investigate consensus variants that allow us to maintain a global consensus on the partial ordering of decentralized events. Criteria are on one hand the computational, storage and message overhead. On the other hand, resilience against denial-of-service attacks, both targeted and random-target, will be investigated. The proposed scheme extends classical consensus by three concepts: Clustering, Hierarchical Aggregation and Pairwise Agreement.

Clustering We refer to local consensus regarding the classical consensus within one Nano Grid. When the participating nodes are arranged according their grid distances in the actual electrical power system, this defines the relative neighborhood between them [St19]. Nodes are dynamically organized with respect to their topological locality into a high-degree tree-structure, which we will call clusters. This way it is possible to solve critical grid states locally.

Hierarchical Aggregation Classical consensus that aims to agree on a single value or schedule for the whole system. The hierarchical distribution grid topology, as shown in Fig. 2, serves as the basis for clustering: Smart Meters aggregate IEDs, street energy management systems aggregate Smart Meters and so on up to the connection power to the transmission grid at a power transformer. In this model, we require consensus in two dimensions, namely on each hierarchy level and between the different layers.

Pairwise Agreement Announcing capabilities in a broadcast manner and then reaching pairwise agreements is an approach that can work completely without leaders and has proven to deliver reasonable results even for inter-cluster consensus [SIB20]. The application of

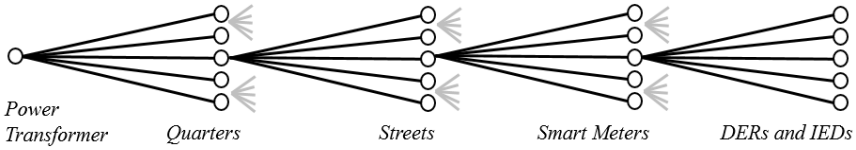


Fig. 2: Schematic hierarchy of the distribution grid and the coordinating Micro Grid.

balancing power between generators and consumers leads to the insight, that the approval of unaffected grid participants is generally not required. Analogous to EPaxos, where an a priori transactions dependency definition is required [MAK13], we are describing concurrent consensus of unrelated events from a grid-topological perspective.

Taking into account the related work of distributed consensus (cf. Section 2) and the suggested new approach, the ongoing research aims to answer the following three research questions:

RQ 1: How can pairwise agreement between nodes be parallelized into a generalized distributed hierarchical consensus protocol? Is pairwise agreement the best local consensus strategy?

RQ 2: How can the parallel agreement then be aggregated into a global state while maintaining fail-safe partition tolerance?

RQ 3: What kind of schedule description and aggregation satisfies the requirements of this Smart Grid balancing scheme, for deriving a global consensus state from concurrent sub-consensus?

When composing hierarchical layers of consensus, nonetheless it is still highly relevant to achieve local in-cluster consensus in a reliable manner. We are evaluating the suitability of the following alternatives to pairwise agreement:

Hierarchical Decision by Elected Leader: Hierarchical decision trees have the main advantage that they resemble the existing structures of power grids, both from the historical experiences as well as the topological reality of distribution grids.

Distributed Ledger: Using distributed ledger technologies has the attribute that all messages are stored for further inspection. Unlike Paxos variants, all nodes have to contribute to the message verification [SPK19]. The main disadvantages are the increased computing and storage requirements that may exceed the capabilities of small devices.

Power of Two: The concept of the Power of Two [Mi01] uses smartly randomized load balancing. The suitability of this concept for hierarchical systems is of special interest, since it leads to a convergence to close-to-optimal solutions while radically reducing the amount of messages that are required to find a consensus.

4 Outlook

In our future work we will elaborate on explicit clustering schemes. Thereby we want to describe an adaptable clustering scheme, such that the affiliation of nodes to clusters can be re-negotiated and the hierarchical structure of superordinate clusters may also shift, in order to respond to environmental influences, such as variations in wind speed, solar radiation or any other factor, which influence the capabilities of the DERs.

We also plan to look into continuous schemes that operate beyond single consensus rounds and are able to re-iterate towards changing optimal solution and clusterings. The distribution grid is structured hierarchically, but the transmission grid can be seen as a planar graph, as visualized in Fig. 3. Therefore we plan additional research towards the difference between hierarchical consensus and consensus in clusters within a flat neighborhood.

Regarding resilience against electrical Denial-of-Service attacks, we will elaborate on availability of power itself: e.g. an attack by synchronously switching on thousands of high-wattage IoT devices. We are looking into schemes that prevent the grid from experiencing cascading failures.

Machine-to-machine validation of identities and their positions in the grid topology will be a hard requirement to support the ongoing transition. In order to achieve valid consensus between authorized participants, the proposed scheme needs to provide protection against identity theft as well as Sybil attacks [NNM13].

Further, we identify two kinds of False-Data-Injection attacks that need to be addressed. First, since the aggregating nodes pose as gate-keepers between the various clusters, they are especially vulnerable to attacks on data integrity. Second, consuming or generating participants may act differently than announced. This is beyond the scope of communication integrity and thereby especially difficult to address.

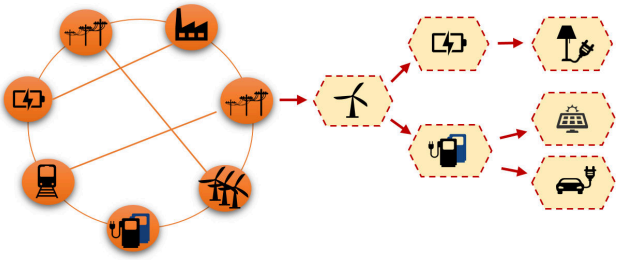


Fig. 3: The transmission grid on the left has a meshed structure whereas the distribution grid is structured hierarchically.

5 Conclusion

We elaborate on the use of hierarchical consensus schemes for Smart Grids, acknowledging both the mostly tree-like topology of the power grid, as well as the interdependency between power flow and the control infrastructure. Implementing a clustering scheme for Micro Grids, we can achieve fast consensus by directly communicating to an dynamically defined, interference-free subset of neighboring nodes and over the hierarchical composition of such clusters. We investigate which distributed consensus schemes are suitable to combine the advantages of parallelization of responsibility with strong respect to the locality of the electrical grid when achieving consensus within self-organizing Micro Grids including power suppliers and consumers.

Bibliography

[BFT11] Bertsimas, Dimitris; Farias, Vivek F; Trichakis, Nikolaos: The price of fairness. Operations research, 59(1):17–31, 2011.

[Br00] Brewer, Eric A: Towards robust distributed systems: The CAP Theorem. In: PODC. volume 7, 2000.

[HMS16] Howard, Heidi; Malkhi, Dahlia; Spiegelman, Alexander: Flexible paxos: Quorum intersection revisited. arXiv preprint arXiv:1608.06696, 2016.

[La98] Lamport, Leslie: The part-time parliament. ACM TOCS, 16(2):133–169, 1998.

[MAK13] Moraru, Iulian; Andersen, David G; Kaminsky, Michael: There is more consensus in egalitarian parliaments. In: Symposium on Operating Systems Principles. ACM, 2013.

[Mi01] Mitzenmacher, Michael: The power of two choices in randomized load balancing. IEEE Transactions on Parallel and Distributed Systems, 12(10):1094–1104, 2001.

[NNM13] Najafabadi, S. G.; Naji, H. R.; Mahani, A.: Sybil attack Detection: Improving security of WSNs for smart power grid application. In: Smart Grid Conference (SGC). IEEE, 2013.

[OO14] Ongaro, Diego; Ousterhout, John: In search of an understandable consensus algorithm. In: USENIX Annual Technical Conference. 2014.

[RWK17] Rizvi, Sajjad; Wong, Bernard; Keshav, Srinivasan: Canopus: A scalable and massively parallel consensus protocol. In: Proceedings of the 13th NEXT Conference. ACM, 2017.

[SIB20] Stübs, Marius; Ipach, Hanko; Becker, Christian: Topology-Aware Distributed Smart Grid Control using a Clustering-Based Utility Maximization Approach. In: 35th ACM Symposium on Applied Computing. 2020.

[SPK19] Stübs, Marius; Posdorfer, Wolf; Kalinowski, Julian: Business-Driven Blockchain-Mempool Model for Cooperative Optimization in Smart Grids. In: SmartCom. Springer, 2019.

[St18] Stübs, Marius: IT-Security in Self-Organizing Decentralized Virtual Power Plants. In: 33rd ACM Symposium on Applied Computing. pp. 431–432, 2018.

[St19] Stübs, Marius: Locality-Aware Overlay Network for VPPs Derived From the Grid Topology. In: 8th DACH+ Conference on Energy Informatics. Springer, pp. 56–59, 2019.

[URZ15] Ur-Rehman, Obaid; Zivic, Natasa: Secure design patterns for security in smart metering systems. In: IEEE European Modelling Symposium (EMS). pp. 278–283, 2015.

Abusers don't get Privacy. Sensitively Logging and Blocking Tor Abuse

Matthias Marx[1]

Abstract: Tor has a significant problem with malicious traffic routed through Tor exit nodes. They create a credible reason for websites to discriminate against Tor users. The abuse also creates a strong disincentive to run exit nodes since the exit node operators have to deal with abuse messages and possible law enforcement interactions. We want to detect and mitigate the attacks that happen through Tor exit nodes without undermining Tor users' anonymity and privacy. We use a modified version of the Tor exit node to enable NIDS (Network Intrusion Detection) monitoring and termination of malicious activity on a per-circuit level. We use the Zeek IDS (formerly Bro) to detect attacks using robust mechanisms that have very low false positive rates. Initial results indicate that, using our approach, the number of abuse cases can be reduced.

Keywords: Tor; Malicious Traffic; Intrusion Detection System

1 Introduction

Abusive use of Tor, where attackers take advantage of Tor's anonymity to launch attacks, damages the Tor network in multiple ways. This includes acting as a significant disincentive to running a Tor exit, serving as a source of negative publicity, and creating a credible reason for websites to discriminate against Tor users.

Several publications describe the abusive use of Tor [Mc08, CMK10, Li15]. Over 20 % of the top Alexa websites discriminate against Tor users. Many of the website operators mention the network attacks passing through Tor as one of the main reasons for discriminating [Si17]. If we can prevent large scale attacks through Tor, we hope to remove major incentives for discriminating against Tor traffic.

The goal of this project is to develop and evaluate intrusion detection policies, guided by abuse complaints, which can run on a Tor exit node to mitigate the outbound attacks without disrupting normal user behavior and without posing a privacy risk to non-malicious users.

[1] Universität Hamburg, Sicherheit in verteilten Systemen, Vogt-Kölln-Straße 30, 22527 Hamburg, Deutschland
marx@informatik.uni-hamburg.de

doi:10.18420/sicherheit2020_16

2 Detection Schemes

Our detection schemes treat each Tor circuit as a separate user which is then evaluated with the Zeek[2] network intrusion detection system (NIDS). We modified the Tor exit source code to report to the NIDS the circuit ID associated with each TCP connection 4-tuple and each DNS lookup. Using this, the NIDS can evaluate each Tor circuit for abuse independently of all other circuits. This also enables the NIDS to optionally terminate a misbehaving circuit without affecting any other user.

We use Zeek with all default logging disabled. Only the limited logging (discussed below) occurs. We use a combination of existing Zeek rules and new rules to detect abuse. Our initial starting points for detecting and blocking abuse are:

1. **Port Scanning:** We use Threshold Random Walk (TRW) [Ju04] to detect port scanning both across hosts and within a single host.
2. **Brute Force Guessing:** We detect brute-force password guessing for HTTPS and SSH using TRW, with a short terminated connection (for SSH) and either a short terminated connection or a regular period of same-sized requests with short responses (for HTTPS) as indicative of failures for the TRW algorithm.
3. **Pattern Matching for HTTP Abuse:** We use regular expressions against the HTTP GET and POST requests which detect various attacks, such as SQL injection or cross site scripting.

It is critical that we use detectors with a very low false positive rate. TRW-type detectors are already well understood and have very low false positive while still being sensitive, and we evaluate the pattern matching for the HTTP abuse against known benign network traffic before deployment on our exit nodes. We will add subsequent detection routines as they prove useful, in particular in response to any abuse complaints we receive directly or through public IP abuse databases.

Upon detecting an attack, the NIDS enables logging for the alerted circuit and optionally terminates the Tor circuit which triggered detection to prevent further abuse after having identified this as a problematic circuit.

3 Methodology

We will operate two exit nodes running our IDS policies which analyze for abuse on a per-circuit basis: one which only logs alerts, and one which both logs alerts and terminates offending circuits. This logging only occurs for circuits which trigger our abuse detectors. For both we will evaluate the complaints received, and see if there is a substantial difference between the one which terminates offending circuits and the one which only logs.

[2] https://www.zeek.org/

Data Collection Intrusion Detection Systems normally collect a large amount of information. We perform just minimal logging for all circuits: the number of distinct hosts and the number of bytes transmitted, with both values truncated and the associated time intervals truncated. This is necessary to estimate the normal, non-malicious usage of the exit.

We only perform more detailed logs on circuits detected as malicious. Such logging includes all detected NIDS events associated with the circuit, in order. The NIDS events are in order but won't be timestamped, and circuit creation is only recorded truncated to the nearest hour. Such logging only takes place once an hour, with the logged circuits presented in randomized order, so as to ensure no correlation between them.

The logs are essential for evaluation: they enable us to determine if our detectors effectively detect abuse (by correlating abuse complaints to log entries) and if such detection is sufficiently early to prevent abuse complaints. If we can't correlate a complaint to a log entry, this suggests holes in our detection strategy.

Modifications to Tor We modified the Tor source code to generate new control events for RELAY RESOLVE and RELAY BEGIN requests which report the circuit ID and the requests' contents. For RELAY RESOLVE cells this reports both the hostname and the answer (IP address, hostname or error). For RELAY BEGIN cells we report the TCP 4-tuple.

Using Stem[3], the NIDS subscribes to a stream of RELAY RESOLVE and RELAY BEGIN events and uses these events to associate the Tor circuit ID with the network traffic to determine attacks. Furthermore, the NIDS can instruct Tor to close a circuit via Stem.

Privacy Concerns The detection algorithms we are planning to use have very low false positive rates. This will be confirmed by experiments in a test network before experimenting on the live Tor network. We limit the granularity of data and will keep the logs private. We do not collect data that we do not need for attack detection. We use Zeek with all default logging disabled and make sure that Zeek does not contact any third party services during operation. Furthermore, we use offline relay identity keys, two factor authentication for SSH, unattended upgrades and limit the number of people who have access to server and data. We are discussing our approach with the Tor Research Safety Board.

Our detection does not require manual inspection of client traffic. The logs are essential for evaluation and do not contain enough information to deanonymize a circuit. They are not interconnected accross circuits and there is only the coarsest of timing information for circuit start and duration. Events within the circuit, although ordered, add very little timing information. An adversary cannot make the system log any other users' traffic by inserting abusive circuits.

Other Concerns We plan to detect large scale attacks: port scanning, brute-force attacks and attacks with known signatures. Removing these abusive traffic will improve Tor's image

[3] https://stem.torproject.org/

and will reduce websites preemptively blocking Tor. Additionally the blocking, by being selective, should not affect normal users.

As a result of our mitigation scheme, attackers could change their behavior. Abusive users could spread their streams over more circuits. We will investigate by how much attacks slow down when we distribute them over many circuits. Furthermore, we examine our approach with the assumption that in the future all relays would mitigate malicious traffic. Therefore, we do not consider that abusive traffic could shift to relays that don't implement mitigation schemes.

4 Evaluations

Pattern Matching for HTTP Abuse We use Zeek and the CICIDS2017 dataset [SLG18] to analyze labeled PCAPs that contain realistic background traffic and common attacks. For detection of SQL injection scanners, we use Zeek's standard module. Detection of cross site scripting scanners is based on NoScript's[4] regular expressions.

We vary the monitoring interval and the threshold that determines if an attack is ongoing based on the number of requests that appear to be suspicious. We want a high threshold to achieve a low false positive rate and at the same time we want a low threshold to detect attacks early.

Figure 1 shows the influence of the threshold parameter for cross site scripting attacks. The results show for a monitoring interval of 10 seconds that no false positives occur for a threshold of 11 or more requests. Attackers who do not want to be detected must use a new circuit every 10 requests. With a monitoring interval of 10 minutes, no false positives occur for a threshold of 25 requests or more.

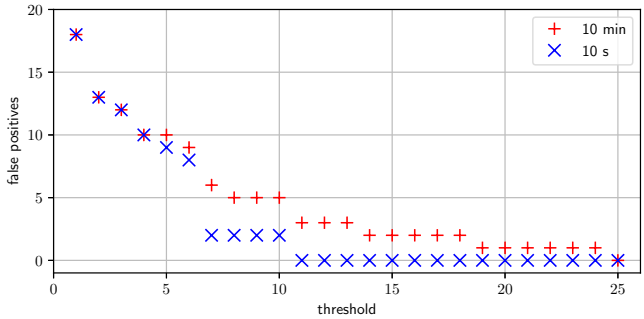


Fig. 1: Number of false positives of regular expressions used to detect cross site scripting attacks for different monitoring intervals (10 s, 10 min) and request thresholds.

[4] https://noscript.net/

Based on the results, in the following experiments we will choose a request threshold of 5 for SQL injection and a threshold of 25 for cross site scripting attacks. Later, we may change the thresholds based on findings from the abuse complaints.

Spreading Attacks Across Circuits We investigate if attacks could simply spread across many circuits to avoid detection. Figure 2 shows that it does not make much difference if one or ten circuits are used to request a website one hundred times. This means that we have to detect attacks after very few requests or circuit creation has to be made more expensive, for example through proof-of-work.

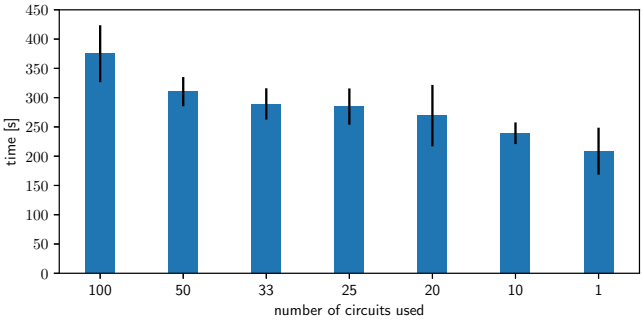


Fig. 2: Time that is needed to request a website 100 times over Tor using one or multiple circuits.

Detecting Abuse on Emulated Tor Exit Nodes We use chutney[5] for running multiple instances of Tor and NetMirage[6] to set up the emulated network. We use several Metasploit[7] modules to run the following attacks: **Port Scanning** across hosts and within a single host, **Brute Force Guessing** of HTTP Basic Auth and SSH credentials, and **SQL injection** and **cross site scripting scanning**. Our experiments in the lab show that the detectors work. We can detect and close circuits that are used for attacks.

Mitigating Abuse on the Live Tor Network We run two exit nodes with reduced exit policies that allow web browsing (ports 80 and 443), SSH (port 22) and port scanning to a limited extent. One exit will only log alerts, and the other will both log alerts and terminate offending circuits. For both we will evaluate the complaints received, and see if there is a substantial difference.

Tor Network Telescope We run a network telescope (analog to [Mo04]) to measure Tor's background noise. We monitor nine unused /8 IPv4 subnets that carry no legitimate traffic. Using ping-scans and BGP data, we have verified that the subnets are actually not used. In the logs we include the following information: circuit ID, number of distinct scanned addresses per /16 subnet, scanned ports and timestamp (hourly granularity). We omit logs for circuits with less than 25 connection attempts. The measured noise will be used to draw conclusions about (undirected) attacks carried out through Tor. On detection of port

[5] `https://gitweb.torproject.org/chutney.git/`
[6] `https://crysp.uwaterloo.ca/software/netmirage/`
[7] `https://www.metasploit.com/`

scanning, we will redirect traffic to a honeypot to provide actionable intelligence on how attackers are using Tor for exploitation.

5 Conclusion & Future Work

In this work, we proposed a new defense against abusive use of Tor. We modified Tor to enable monitoring and termination of malicious activity on a per-circuit level. Initial results indicate that, using our approach, attacks can be detected and prevented. It remains to be checked whether the number of abuse complaints can actually be reduced.

In the future, our approach could be extended to provide network-wide statistics on abuse of Tor. To further enhance privacy, the NIDS could run in an encrypted enclave that only outputs circuit IDs for circuits that are deemed malicious. Alternatives to termination of circuits, such as degradation of quality of service, should also be investigated.

Acknowledgements

The author wants to thank Sadia Afroz for the fruitful discussion and the anonymous reviewers for their useful comments and suggestions. This work was partially supported by NSF grant CNS-1518918.

Bibliography

[CMK10] Chaabane, Abdelberi; Manils, Pere; Kaafar, Mohamed Ali: Digging into anonymous traffic: A deep analysis of the Tor anonymizing network. In: International Conference on Network and System Security. IEEE, pp. 167–174, 2010.

[Ju04] Jung, Jaeyeon; Paxson, Vern; Berger, Arthur W; Balakrishnan, Hari: Fast portscan detection using sequential hypothesis testing. In: Security and Privacy. IEEE, pp. 211–225, 2004.

[Li15] Ling, Zhen; Luo, Junzhou; Wu, Kui; Yu, Wei; Fu, Xinwen: TorWard: Discovery, Blocking, and Traceback of Malicious Traffic Over Tor. IEEE Transactions on Information Forensics and Security, 10(12):2515–2530, 2015.

[Mc08] McCoy, Damon; Bauer, Kevin; Grunwald, Dirk; Kohno, Tadayoshi; Sicker, Douglas: Shining light in dark places: Understanding the Tor network. In: Privacy Enhancing Technologies Symposium. Springer, pp. 63–76, 2008.

[Mo04] Moore, David; Shannon, Colleen; Voelker, Geoffrey; Savage, Stefan et al.: Network telescopes. Technical report, Cooperative Association for Internet Data Analysis, 2004.

[Si17] Singh, Rachee; Nithyanand, Rishab; Afroz, Sadia; Pearce, Paul; Tschantz, Michael Carl; Gill, Phillipa; Paxson, Vern: Characterizing the Nature and Dynamics of Tor Exit Blocking. In: USENIX Security. 2017.

[SLG18] Sharafaldin, Iman; Lashkari, Arash Habibi; Ghorbani, Ali A: Toward: Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization. In: ICISSP. pp. 108–116, 2018.

Autorenverzeichnis