

IFIP AICT 505

Jan-Philipp Steghöfer
Babak Esfandiari
(Eds.)



Trust Management XI

11th IFIP WG 11.11 International Conference, IFIPTM 2017
Gothenburg, Sweden, June 12–16, 2017
Proceedings

 Springer



Editor-in-Chief

Kai Rannenber, Goethe University Frankfurt, Germany

Editorial Board

TC 1 – Foundations of Computer Science

Jacques Sakarovitch, Télécom ParisTech, France

TC 2 – Software: Theory and Practice

Michael Goedicke, University of Duisburg-Essen, Germany

TC 3 – Education

Arthur Tatnall, Victoria University, Melbourne, Australia

TC 5 – Information Technology Applications

Erich J. Neuhold, University of Vienna, Austria

TC 6 – Communication Systems

Aiko Pras, University of Twente, Enschede, The Netherlands

TC 7 – System Modeling and Optimization

Fredi Tröltzsch, TU Berlin, Germany

TC 8 – Information Systems

Jan Pries-Heje, Roskilde University, Denmark

TC 9 – ICT and Society

Diane Whitehouse, The Castlegate Consultancy, Malton, UK

TC 10 – Computer Systems Technology

Ricardo Reis, Federal University of Rio Grande do Sul, Porto Alegre, Brazil

TC 11 – Security and Privacy Protection in Information Processing Systems

Steven Furnell, Plymouth University, UK

TC 12 – Artificial Intelligence

Ulrich Furbach, University of Koblenz-Landau, Germany

TC 13 – Human-Computer Interaction

Marco Winckler, University Paul Sabatier, Toulouse, France

TC 14 – Entertainment Computing

Matthias Rauterberg, Eindhoven University of Technology, The Netherlands

IFIP – The International Federation for Information Processing

IFIP was founded in 1960 under the auspices of UNESCO, following the first World Computer Congress held in Paris the previous year. A federation for societies working in information processing, IFIP's aim is two-fold: to support information processing in the countries of its members and to encourage technology transfer to developing nations. As its mission statement clearly states:

IFIP is the global non-profit federation of societies of ICT professionals that aims at achieving a worldwide professional and socially responsible development and application of information and communication technologies.

IFIP is a non-profit-making organization, run almost solely by 2500 volunteers. It operates through a number of technical committees and working groups, which organize events and publications. IFIP's events range from large international open conferences to working conferences and local seminars.

The flagship event is the IFIP World Computer Congress, at which both invited and contributed papers are presented. Contributed papers are rigorously refereed and the rejection rate is high.

As with the Congress, participation in the open conferences is open to all and papers may be invited or submitted. Again, submitted papers are stringently refereed.

The working conferences are structured differently. They are usually run by a working group and attendance is generally smaller and occasionally by invitation only. Their purpose is to create an atmosphere conducive to innovation and development. Refereeing is also rigorous and papers are subjected to extensive group discussion.

Publications arising from IFIP events vary. The papers presented at the IFIP World Computer Congress and at open conferences are published as conference proceedings, while the results of the working conferences are often published as collections of selected and edited papers.

IFIP distinguishes three types of institutional membership: Country Representative Members, Members at Large, and Associate Members. The type of organization that can apply for membership is a wide variety and includes national or international societies of individual computer scientists/ICT professionals, associations or federations of such societies, government institutions/government related organizations, national or international research institutes or consortia, universities, academies of sciences, companies, national or international associations or federations of companies.


More information about this series at <http://www.springer.com/series/6102>

Jan-Philipp Steghöfer · Babak Esfandiari (Eds.)

Trust Management XI

11th IFIP WG 11.11 International Conference, IFIPTM 2017
Gothenburg, Sweden, June 12–16, 2017
Proceedings

Editors

Jan-Philipp Steghöfer 
Chalmers University of Technology
Gothenburg
Sweden

Babak Esfandiari
Carleton University
Ottawa, ON
Canada

ISSN 1868-4238

ISSN 1868-422X (electronic)

IFIP Advances in Information and Communication Technology

ISBN 978-3-319-59170-4

ISBN 978-3-319-59171-1 (eBook)

DOI 10.1007/978-3-319-59171-1

Library of Congress Control Number: 2017941491

© IFIP International Federation for Information Processing 2017

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Printed on acid-free paper

This Springer imprint is published by Springer Nature

The registered company is Springer International Publishing AG

The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

The 11th edition of IFIPTM — the IFIP WG11.11 International Conference on Trust Management — held in Gothenburg, Sweden, continued the tradition of a venue that focuses on trust, the glue of any society. Underlying the different papers and the many discussions at the conference, was the insight that trust is not automatic and not self-evident, but that it must be nurtured and managed, regardless of whether the society we speak of is natural or artificial. Computational techniques, such as the ones addressed by IFIPTM and its community, enable services for brokering, certification, recommendation, legal enforcement, identity, and reputation management. Such services become all the more useful and important given the increasing scale and virtual nature of societies, in which the human and the artificial agent mix and interact without prejudice.

IFIPTM is a well-established conference in this field and we observe a steady number of submissions that is suitable for a small venue such as this. This year, we received 29 submissions and were able to accept eight full papers and six short papers. Some of these papers were shepherded, i.e., the reviewers offered their feedback and authors were able to address it by revising their papers before the final decision. This process was well received by both reviewers and authors and allowed us to include additional, promising work. Our 32 Program Committee members produced a total of 95 reviews and dozens of comments in a very lively and engaging decision process.

The selected papers represent the broad topical areas of the call for papers. They are structured in five thematical sessions. The papers in the area of “Information Sharing and Personal Data” address different topics of the information economy and how trust management techniques can help ensure the privacy of personal data in domains such as the Internet of Things. In “Novel Sources of Trust and Trust Information,” the authors explore where the data to base trust decisions on can come from and investigate sources as diverse as behavioral experiments, social networks, and service-level agreements. The papers in the section of “Applications of Trust” reveal novel ways to use existing trust values — including to decide when to uninstall software, detect intrusion into computer systems, or how self-trust and self-efficacy are connected in education. Interesting ways to calculate trust values are the subject of the “Trust Metrics” area. Behavioral profiling and flow models are the basis of two of the metrics, while the third focuses on mathematical properties of specific trust metrics that allow for composition of trust. Finally, the two contributions in the area “Reputation Systems” focus on how entities can report their own reputation without tampering and how reputation can improve recommender systems.

Another thematical focus of the conference was provided by the special session “Trust on the Road” that focused on trust management in vehicular networks, including vehicle-to-vehicle and vehicle-to-infrastructure communication. The special session was spearheaded by Mathias Widman of Volvo Group Telematics/WirelessCar, who spoke about the difficulties of extending vehicle trust and security. A panel discussion

including participants from Volvo Cars, Chalmers University of Technology, and the trust management community facilitated the further exploration of this up-and-coming topic.

The dichotomy of trust and security was the topic of Max Mühlhauser's keynote. Instead of following the attempts of his predecessors to separate the fields, Max showed that they are indeed synergetic.

In addition, we are happy to include the paper accompanying the keynote by Siani Pearson, holder of the William Winsborough Commemorative Address and Award 2017, in the proceedings. The objective of the award is to publicly recognize an individual who has significantly contributed to the development of computational trust or trust management, especially achievements with an international perspective. The award is given out in memory of Professor William Winsborough, who taught at the University of Texas at San Antonio, in recognition of his leadership in the field of trust and trust management. Siani was honored for her own leading role in the area and received 2017's award for her outstanding track record and her long-standing engagement in the community. Her paper and keynote illuminated how the fast-changing use of information technology challenges traditional notions of accountability and how the concept of accountability relates to trust.

Last but not least, we were happy to have received three contributions from young researchers who participated in the IFIPTM Graduate Symposium. There, renowned researchers and students at any stage of their graduate career discussed the research, open issues, and state of the art in the field of computational trust and trust management. The symposium featured lectures by experts in the field, exploring the theory, philosophy, and practice of trust and trust management and its application to society and science. There was ample opportunity to network with presenters and other students. Participants worked on small projects together to apply skills and knowledge and learn from each other. Tim Schürmann's paper explored the human decision processes in socio-technical systems and how much they are guided by trust. Tosan Atele-Williams discussed how much we can trust information and what the social and cognitive foundations for information trust are. Finally, Vida Ahmadi Mehri explored requirements for trust and privacy for cloud-based marketplaces and how they compare to current definitions.

To conclude, we would like to express our thanks to everyone who contributed to the organization of IFIPTM this year. We are, of course, indebted to the entire Program Committee for their commitment and enthusiasm in all phases of the reviewing process, and for the quality and insight of their reviews. We also thank the chairs of previous IFIPTM editions for their feedback on past experiences and general advice along the way, which was extremely helpful. We also benefited from working closely with the other chairs on the committee, Simone Fischer-Hübner, Stephen Marsh, Musard Balliu, Sheikh Mahbub-Habib, and Tomas Olovsson, who provided continual and unstinting support during the entire endeavor.

April 2017

Babak Esfandiari
Jan-Philipp Steghöfer

| | |
|---------------------|---|
| Jie Zhang | Nanyang Technological University, Singapore |
| Natasha Dwyer | Victoria University, Australia |
| Zeinab Noorian | Ryerson University, Canada |
| Stephen Marsh | UOIT, Canada |
| Tim Muller | University of Oxford, UK |
| Roslan Ismail | Tenaga National University |
| Alan Davoust | Carleton University, Canada |
| Weizhi Meng | Technical University of Denmark, Denmark |
| Anirban Basu | KDDI Research, Japan |
| Jesus Luna Garcia | TU Darmstadt, Germany |
| Tanja Pavleska | Jozef Stefan Institute, Slovenia |
| Sheikh Mahbub Habib | Technische Universität Darmstadt, Germany |
| David Chadwick | University of Kent, UK |
| Yuecel Karabulut | Oracle, USA |
| Tim Storer | University of Glasgow, UK |
| Sara Foresti | University of Milan, Italy |
| Nurit Gal-Oz | Sapir Academic College, Israel |

Contents

Information Sharing and Personal Data

| | |
|--|----|
| Partial Commitment – “Try Before You Buy” and “Buyer’s Remorse” for Personal Data in Big Data & Machine Learning. | 3 |
| <i>Lothar Fritsch</i> | |
| VIGraph – A Framework for Verifiable Information | 12 |
| <i>Anirban Basu, Mohammad Shahriar Rahman, Rui Xu, Kazuhide Fukushima, and Shinsaku Kiyomoto</i> | |
| A Flexible Privacy-Preserving Framework for Singular Value Decomposition Under Internet of Things Environment. | 21 |
| <i>Shuo Chen, Rongxing Lu, and Jie Zhang</i> | |

Novel Sources of Trust and Trust Information

| | |
|--|----|
| The Game of Trust: Using Behavioral Experiment as a Tool to Assess and Collect Trust-Related Data | 41 |
| <i>Diego de Siqueira Braga, Marco Niemann, Bernd Hellingrath, and Fernando Buarque de Lima Neto</i> | |
| Social Network Analysis for Trust Prediction | 49 |
| <i>Davide Ceolin and Simone Potenza</i> | |
| Investigating Security Capabilities in Service Level Agreements as Trust-Enhancing Instruments. | 57 |
| <i>Yudhistira Nugraha and Andrew Martin</i> | |

Applications of Trust

| | |
|---|-----|
| Managing Software Uninstall with Negative Trust. | 79 |
| <i>Giuseppe Primiero and Jaap Boender</i> | |
| Towards Trust-Aware Collaborative Intrusion Detection: Challenges and Solutions | 94 |
| <i>Emmanouil Vasilomanolakis, Sheikh Mahbub Habib, Pavlos Milaszewicz, Rabee Sohail Malik, and Max Mühlhäuser</i> | |
| Self-trust, Self-efficacy and Digital Learning. | 110 |
| <i>Natasha Dwyer and Stephen Marsh</i> | |

Trust Metrics

Advanced Flow Models for Computing the Reputation of Internet Domains . . . 119
Hussien Othman, Ehud Gudes, and Nurit Gal-Oz

Trust Trust Me (The Additivity) 135
Ken Mano, Hideki Sakurada, and Yasuyuki Tsukada

Towards Statistical Trust Computation for Medical Smartphone Networks
Based on Behavioral Profiling 152
Weizhi Meng and Man Ho Au

Reputation Systems

Reputation-Enhanced Recommender Systems 163
Christian Richthammer, Michael Weber, and Günther Pernul

Self-reported Verifiable Reputation with Rater Privacy. 180
Rémi Bazin, Alexander Schaub, Omar Hasan, and Lionel Brunie

William Winsborough Commemorative Address and Award 2017

Strong Accountability and Its Contribution to Trustworthy Data Handling
in the Information Society 199
Siani Pearson

IFIPTM 2017 Graduate Symposium

Information Trust 221
Tosan Atele-Williams and Stephen Marsh

Privacy and Trust in Cloud-Based Marketplaces for AI and Data Resources . . . 223
Vida Ahmadi Mehri and Kurt Tutschku

Psychological Evaluation of Human Choice Behavior in Socio-Technical
Systems: A Rational Process Model Approach 226
Tim Schürmann

Author Index 229

Information Sharing and Personal Data

Partial Commitment – “Try Before You Buy” and “Buyer’s Remorse” for Personal Data in Big Data & Machine Learning

Lothar Fritsch^(✉)

Karlstad University, Karlstad, Sweden
lothar.fritsch@kau.se

Abstract. The concept of partial commitment is discussed in the context of personal privacy management in data science. Uncommitted, promiscuous or partially committed user’s data may either have a negative impact on model or data quality, or it may impose higher privacy compliance cost on data service providers. Many Big Data (BD) and Machine Learning (ML) scenarios involve the collection and processing of large volumes of person-related data. Data is gathered about many individuals as well as about many parameters in individuals. ML and BD both spend considerable resources on model building, learning, and data handling. It is therefore important to any BD/ML system that the input data trained and processed is of high quality, represents the use case, and is legally processed in the system. Additional cost is imposed by data protection regulation with transparency, revocation and correction rights for data subjects. Data subjects may, for several reasons, only partially accept a privacy policy, and chose to opt out, request data deletion or revoke their consent for data processing. This article discusses the concept of partial commitment and its possible applications from both the data subject and the data controller perspective in Big Data and Machine Learning.

Keywords: Big Data · Machine Learning · Data sharing · Personal information · Information privacy · Commitment · Consent · Data processing · User interface · Interaction

1 Introduction

Collection and processing of personal data is an important component of contemporary IT services. Many contemporary services are free of financial charge for end users, however they demand collection of personal data and the provisioning of advertising services as compensation. A new emerging business model for free-of-charge services is the accumulation, elaboration, analysis and selling of data provided by the users. The handling of personal data is regulated according to data protection legislation. In Europe’s General Data Protection regulation (GDPR) [1], data processors shall collect legally valid informed consent from the data subjects before they collect and process their personal data. Such informed consent should specify the scope of data collection, provide details about storage and processing, specify the purpose of data use, and indicate other parties that will get access to the data. Users are usually presented with a

privacy policy text in prose which they will have to accept and confirm as it is. Privacy policies are known to misinform [2], and to impose a high burden of responsibility on the data subjects [3]. Automatic negotiation of privacy policy references has been explored with P3P and EPAL, however is rarely found in existing systems [4]. The provision of consent is therefore, in practice, YES-NO binary decision. Service providers fulfill their legal obligation, while data subjects usually skip reading the privacy policy on their way to access the free-of-charge service. Many reasons for such behavior are found – lack of time, lack of legal understanding, pseudonymous use of services with fake identities, and non-commitment, for example for the purpose of testing the service. Data subjects might, therefore, be unaware of or ignorant about the nature of data collection and processing the service relies upon. They might accept a privacy policy with a “maybe” intention, just to proceed into using the service.

The collection of data from non-committed data subjects may, however, pose a risk to the intentions of the service provider. Dependent on the purpose of data collection, the provisioning of fake identities, incomplete or fabricated data or data patterns created through playful testing of a service may reduce the quality of the collected data. In addition, the accumulation of non-committed data subject’s data into a sample that shall represent the user population may misrepresent users upon opt-out of the uncommitted users. Non-commitment poses therefore a hazard for data quality, may endanger training data sets, statistical norm data sets, and may cause long-stranding data protection compliance obligations with respect to data protection enquiries and transparency rights.

As a solution to this problem, we suggest the introduction of partial commitment into the handling of data processing consent. We propose to extend the YES-NO choice offered today by a MAYBE option that expresses partial commitment. The remainder of the article will elaborate the background of partial commitment, discuss particular benefits both data subjects and data processors might receive from partial commitment, and drafts a research agenda for the further investigation of partial commitment to personal data processing.

1.1 Background

Commitment, or the lack thereof, has been the subject of research in many disciplines. This section reviews the results of literature research for the concept of partial commitment, delayed commitment, non-commitment and promiscuous commitment. Examples from the technology domain are the reachability manager for mobile communications which contains numerous options for policies for personal reachability for direct communications [5]. Another variant is a customer self-care interface for location services in mobile networks where customers can control fine-grained opt-in and opt-out functions against any third-party service provider [6]. One base technology for partial commitment is a reference storage for various policies which can then be, under the commitment process, referenced by the negotiating stakeholders [7]. Commitment has been discussed in the areas of risk acceptance, choice and decision-making. In psychology, a known phenomenon is a preference for the status quo. Human beings seem, when confronted with decision-making, show a preference for the status quo [8].

Reasons for this are uncertainty, incomplete information, loss aversion, complexity of the alternatives and many other aspects discussed in literature. Recent research on choice architecture deepens insight into how information presentation supports decision-making [9]. Another influential aspect of commitment is fairness in interaction. Procedural justice may improve user cooperation and data quality, as found in [10]. In addition, procedural fairness is found to increase trust in on-line applications [11]. From a trust management perspective, trust partial commitment can be assumed an integral part of pessimistic and investigative trust-building strategies [12]. A connection between privacy policies and the level of customer loyalty has been observed in recurring consumer studies on web portals [13]. Consequently, giving consent to the processing of personal data can be seen as a dialogue, not a monologue over the particularities of releasing personal data and engaging into a contract with a service provider [14]. Lack of information may cause decision procrastination in search for more information [15]. From this perspective, the usability of privacy policies can be decisive for data subject commitment, as they are part of end-user decision making [16]. There is evidence about a tight binding between good stakeholder relationship and commitment. Customer relationship management is concerned strongly with customer commitment. The importance of commitment in relationship marketing was described in [17] as: “Commitment is an important variable in the relationship marketing goal system. It is a prerequisite for the customer to proactively seek relationship maintenance whereas uncommitted customers can only be kept in relationships through instruments such as use of power, long-term contracts or in monopoly situations.”

1.2 Challenges

Many users of internet services who accept service terms & conditions and the related privacy policies are not committed at the time they sign up. They test the service, and may resign or opt out a short time in the future. Such leaving customers’ data may cause a number of issues in BD/ML systems:

- According to upcoming European data protection legislation [1], data subjects will have extensive rights concerning data protection inquiries, data export and data deletion requests from 2018 on. A BD/ML operator will have to prepare all data processing systems to comply with such requests, even for uncommitted short-term users of the services. This will cause major liabilities and compliance efforts.
- Machine Learning models trained with data gathered from non-committed data subjects may not make as good decisions as those trained with committed data subjects’ data. Service providers may be interested in separation of data acquired from committed and non-committed users. Uncommitted data subjects may “pollute” the data pool and the models.
- “Roll-back” of learning models or data collections that collect aggregated data in the case of data subject opt-out may be difficult performed on simple data bases. A roll-back mechanism for ML and for various forms for BD data aggregation should support opt-out of data subjects, including their contribution to the models and databases. Roll-back may prove useful when trying to fight pollution of models and data sets by uncommitted data subjects.

- Resulting models and databases should provide sufficient audit information about personal data processed into them, and how it contributed to model building and decision-making. Quality insurance and demonstrability of correct data processing might be essential once analysis results are questioned.

The handling of the aforementioned challenges requires strategies and techniques to handle them in an application processing data from uncommitted data subjects. In the following section, we suggest and investigate the concept of partial commitment, and how its conceptualization as a classification tools could be used to solve the challenges above.

2 Partial Commitment as a Concept: The MAYBE Button

In this section, the concept of partial commitment into processing of personal data is presented. The concept of partial commitment was suggested by Elena Barrantes for the rump session on the 11th IFIP Summer School on Privacy and Identity Management in Karlstad, Sweden, in August 2015. Lothar Fritsch moderated the discussion following the presentation. The participants – researchers, industry participants and PhD students – brainstormed about the concept, its interpretation and its uses.

The suggestion starting the brainstorming was the question whether there should be a “MAYBE button” next to the accept/decline choices when providing consent to a privacy policy (see Fig. 1). In the following sections, we will discuss the stakeholder perspectives on partial commitment. We focus on the two stakeholders “data subjects” (delivering data, expected to accept a privacy policy to access a service) and “service provider” (a personal data consuming service that expects a data subject to give some form of consent to data processing. On the rump session workshop at the 11th IFIP Summer School on Privacy and Identity Management, the participants were asked to brainstorm possible beneficial uses and implications of a “Maybe” option on privacy policies, both for data subjects and for service providers. The results were collected, analyzed and used to formulate benefits from both stakeholder groups’ perspectives, which are summarized in the following two sections.

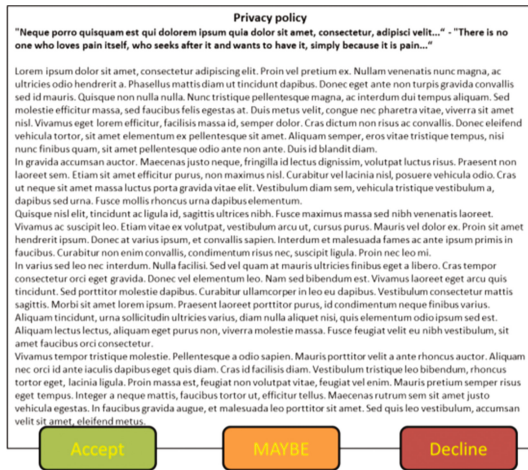


Fig. 1. Partial commitment through the MAYBE option.

2.1 Data Subjects’ Perspective

On the rump session workshop, the participants produced four different data subject perspectives on partial commitment.

First: Why should one commit at all? Concerns were raised about how realistic a policy reflects actual data processing, how much a – yet unknown – service is worth the commitment, and about how little trust information is known about the service provider. Participants partially voiced a strong wish of ownership over their data, and voiced concerns about granting too many privileges to service providers. It was stated that there is no time to read and comprehend privacy policies, which should get compensated by possibly committing later.

Second: Inappropriateness of the privacy policy. Participants expressed concern over the appropriateness, fairness, or truthfulness of the presented privacy policy. They voiced usefulness of delayed or partial commitment where confronted with policies that are either incomprehensible (too complicated, too long, poorly written), unfair (too general, one-sided, too much power transferred to the service provider), poorly specified (written for another legal system) or technically unusable (display on devices not suitable for reading).

Third: Promiscuity - Exploration and experimentation. Participants expressed the usefulness of unconditional, playful trial options and exploration of new services. In addition, they stated that they want to be able to use several services without much consideration about the implications of their privacy policies in intersection.

Fourth: Counteraction and retaliation when faced with no choices. Participants expressed that they, in cases where they find privacy policies unacceptable, but where they have to use the services for some reason, chose obfuscation or sabotage strategies such as entering fake identities, fake data, and the intentional provocation of false profiles. The possibility of partial commitment could reduce the need for such strategies.

From the data subject’s perspective, a partial commitment can implement three different modes of interaction with a data-consuming service:

- **Promiscuity against yet unknown services or providers.** In this mode, the data subject has principal objections against commitment to a service provider. Why give exclusive rights over data and possible profits generated with it to a single stakeholder one has not yet established a relationship to, or built up trust in? Data subjects may wish to “sell” their data to several stakeholders, and chose how their data gets used freely. Depending on choices they get offered, they may delay commitment as they are not yet convinced that they have found the one service provider that suits best for their needs and requirements.
- **Test-before-commitment.** In this mode, a data subject executes the “try before you buy” philosophy. Reasons may be the satisfaction of curiosity, simple playful exploration of new services without serious commitment intentions, or mistrust in the quality of delivered service. “Try before you buy” schemes are implemented in various areas of life. In consumer protection law, when buying at the door, via telephone or on the internet, buyers can leave the contract for a certain period. Commercial providers of subscriptions, ranging from newspapers to telecommunication services,

often offer discounted trial subscriptions for limited time periods to get customers to try out new products or services.

- **Verify realities behind privacy promises.** Often, the privacy policies and service descriptions are incomprehensible to data subjects. It is hard to evaluate the implications, consequences and accuracy of privacy policies [18] and their technical and administrative enforcement [4, 19]. Data subjects may use partial commitment for the purpose of exploring and evaluation of the reality of personal data processing in the service.

The presented modes of partial commitment may help data subjects therefore help with trust establishment, help with the playful exploration and adaption of new services, and can establish a dialogue between data subjects and service providers about privacy preferences.

2.2 Service Providers' Perspective

On the rump session workshop, the participants produced four different service provider perspectives on partial commitment.

First: Measurement of privacy policy reception by data subjects. Delayed commitment could be used as a signal for poor readability or unacceptable privacy policies. Various forms of signals could help to understand customer objections. As a hypothesis, the measurement of frequencies of partial commitment was suggested: The more “maybe” commitments, the more confused or hesitant are the data subjects.

Second: Isolation of data from committed and little/not committed users. Using partial commitment, data processing services can manage separate pools of data, dependent on levels of commitment. Participants suggested that varying levels of data quality, service usage intensity and motivation of providing personal data will have a measureable impact on data quality and service quality.

Third: Focus on data consumption for Big Data applications and training sets for ML. Participants voiced concern over the accuracy of forecasting applications, ML based decisions and BD analytics when based on a data set that contains data from uncommitted or partially committed data sets. Separate data sets and models were suggested.

Fourth: Provision commitment metadata that enable rollback end reduces data management cost. Participants expected that, through available metadata on commitment levels, all forms of data management obligations (quality insurance, privacy transparency request handling, proof of foundations of automated decision-making) could be supported effectively.

From the service provider perspective, partial commitment can implement therefore three different benefits:

- **Measure the quality of privacy policies.** By assessing frequencies and detail aspects of various offered forms of partial commitment, service providers can assess the end user perspective on their privacy policies. A measurement resulting in low acceptance could then initiate a process with the aim to remove the problem.

This can be seen as the start of a communication and negotiation process for a more acceptable, and hence more customer-friendly service.

- **Separate data into classes of commitment.** Partial commitment can help with data separation along several dimensions. It can help keeping committed and uncommitted data pools separate, and may thereby improve the quality of data analysis, machine learning data sets, and decision-making. Commitment metadata may help with the deployment of services with better target population match, and may help improving the overall quality of data sets.
- **Prevent future separation and management cost.** Through suitable data classification, separation and labeling, the assessment of BD/ML decisions can better get planned, investigated, rolled back, or proven to 3rd parties. Compliance issues such as transparency and data deletion (data protection) and fairness (consumer law) can get managed better, with higher precision, and improved audibility. Systematic documentation and consideration of commitment levels may therefore prevent future cost.

In summary, partial commitment can be a tool for service providers to assess the acceptance of their privacy policies. It can be used as a tool for data separation and quality insurance, and it could, in addition, get deployed as a strategy for cost reduction, service quality improvement, and better transparency in analytics and automated decision-making.

3 Research Opportunities

From the above observation, I propose the scientific examination of the value of partial commitment in research activities. We propose to:

- Develop interaction patterns and architecture patterns for partial commitment;
- Map stakeholder needs and priorities;
- Perform usability research on user interface for partial commitment;
- Build a model for dynamic privacy management and data management with changing user commitment;
- Evaluate a prototypical implementation.

Additional interdisciplinary research opportunities can be included with:

- Research on the legal foundations, constraints and opportunities of partial commitment, e.g. through the construction of an analog to remorse periods in e-commerce or test subscriptions in telecommunications and Pay TV;
- Research on psychological aspects of usability and trust establishment between data collectors and data subjects;
- Information systems research on the influence of partial commitment on technology acceptance, diffusion, business model alignment, customer satisfaction, customer engagement, data crowdsourcing, and ad-hoc consent to data processing.

Both theoretical and applied research opportunities can be realized. In particular industry partners in the areas of Big Data, Machine Learning, Smart and autonomous

networks cars, mobile telecommunications, Internet of Things, electronic health services and marketing and customer management services should be interested in the opportunities provided by partial commitment.

4 Conclusion

I introduced the concept of partial commitment to the collection and processing of personal data. We analyzed the data subject and data processor perspective on partial commitment, followed by an identification of stakeholder benefits, including possible acceptance and trust increasing effects on the customer relationship in business models based on personal data. We showed the foundations of the concept in scientific literature, and identified a research agenda that will investigate the concept of partial commitment in the context of information privacy and data protection further, both in theory and in applied research.

References

1. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation), European Union L 119/1 (2016)
2. Antón, A.I., Earp, J.B., Reese, A.: Analyzing website privacy requirements using a privacy goal taxonomy. Presented at the Proceedings of the 10th Anniversary IEEE Joint International Conference on Requirements Engineering (2002)
3. Vila, T., Greenstadt, R., Molnar, D.: Why we can't be bothered to read privacy policies: models of privacy economics as a lemons market, pp. 403–407. ACM Press, Pittsburgh (2003)
4. Stufflebeam, W.H., Antón, A.I., He, Q., Jain, N.: Specifying privacy policies with P3P and EPAL: lessons learned. Presented at the Proceedings of the 2004 ACM workshop on Privacy in the electronic society, Washington DC, USA (2004)
5. Reichenbach, M., Damker, H., Federrath, H., Rannenberg, K.: Individual management of personal reachability in mobile communication. In: Yngström, L., Carlsen, J. (eds.) *Information Security in Research and Business*. IFIP, pp. 164–174. Springer, Boston (1997). doi:[10.1007/978-0-387-35259-6_14](https://doi.org/10.1007/978-0-387-35259-6_14)
6. Zibuschka, J., Fritsch, L., Radmacher, M., Scherner, T., Rannenberg, K.: Privacy-friendly LBS: a prototype-supported case study. In: 13th Americas Conference on Information Systems (AMCIS), Keystone, Colorado, USA (2007)
7. Jøsang, A., Fritsch, L., Mahler, T.: Privacy policy referencing. In: Katsikas, S., Lopez, J., Soriano, M. (eds.) *TrustBus 2010*. LNCS, vol. 6264, pp. 129–140. Springer, Heidelberg (2010). doi:[10.1007/978-3-642-15152-1_12](https://doi.org/10.1007/978-3-642-15152-1_12)
8. Samuelson, W., Zeckhauser, R.: Status quo bias in decision making. *J. Risk Uncertainty* **1**, 7–59 (1988)
9. Münscher, R., Vetter, M., Scheuerle, T.: A review and taxonomy of choice architecture techniques. *J. Behav. Decis. Making* **29**, 511–524 (2016)

10. Muthoo, A.: A bargaining model based on the commitment tactic. *J. Econ. Theor.* **69**, 134–152 (1996)
11. Lauer, T.W., Deng, X.: Building online trust through privacy practices. *Int. J. Inf. Secur.* **6**, 323–331 (2007)
12. Fritsch, L., Groven, A.-K., Schulz, T.: On the internet of things, trust is relative. In: Wichert, R., Laerhoven, K., Gelissen, J. (eds.) *AMI 2011. CCIS*, vol. 277, pp. 267–273. Springer, Heidelberg (2012). doi:[10.1007/978-3-642-31479-7_46](https://doi.org/10.1007/978-3-642-31479-7_46)
13. Flavián, C., Guinalú, M.: Consumer trust, perceived security and privacy policy: three basic elements of loyalty to a web site. *Ind. Manag. Data Syst.* **106**, 601–620 (2006)
14. Coles-Kemp, L., Kani-Zabihi, E.: On-line privacy and consent: a dialogue, not a monologue. Presented at the Proceedings of the 2010 workshop on New security paradigms, Concord, Massachusetts, USA (2010)
15. Ferrari, J.R., Dovidio, J.F.: Examining behavioral processes in indecision: decisional procrastination and decision-making style. *J. Res. Pers.* **34**, 127–137 (2000)
16. Jensen, C., Potts, C.: Privacy policies as decision-making tools: an evaluation of online privacy notices. Presented at the Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Vienna, Austria (2004)
17. Ivens, B.S., Pardo, C.: Are key account relationships different? Empirical results on supplier strategies and customer reactions. *Ind. Mark. Manag.* **36**, 470–482 (2007)
18. Sunyaev, A., Dehling, T., Taylor, P.L., Mandl, K.D.: Availability and quality of mobile health app privacy policies. *J. Am. Med. Inf. Assoc.* **22**, 1–4 (2014)
19. Earp, J.B., Antón, A.I., Aiman-Smith, L., Stufflebeam, W.H.: Examining internet privacy policies within the context of user privacy values. *IEEE Trans. Eng. Manag.* **52**, 227–237 (2005)

VIGraph – A Framework for Verifiable Information

Anirban Basu^(✉), Mohammad Shahriar Rahman, Rui Xu,
Kazuhide Fukushima, and Shinsaku Kiyomoto

KDDI Research, Fujimino, Japan

{basu,mohammad,ru-xu,ka-fukushima,kiyomoto}@kddi-research.jp

Abstract. In order to avail of some service, a user may need to share with a service provider her personal chronological information, e.g., identity, financial record, health information and so on. In the context of financial organisations, a process often referred to as the *know your customer* (KYC) is carried out by financial organisations to collect information about their customers. Sharing this information with multiple service providers duplicates the data making it difficult to keep it up-to-date as well as verify. Furthermore, the user has limited to no control over the, mostly sensitive, data that is released to such organisations. In this *preliminary work*, we propose an efficient framework – Verifiable Information Graph or VIGraph – based on generalised hash trees, which can be used for verification of data with selective release of sensitive information. Throughout the paper, we use personal profile information as the running example to which our proposed framework is applied.

Keywords: Privacy · Hash tree · Verifiability · Selective disclosure

1 Introduction

Verifiable user identity information has always been a cornerstone in security and authentication. Most services, both online and offline, require a user to provide information to prove the user’s identity. Chronological non-identity related information is also used in other scenarios. For instance, in Japan, users may maintain their medicine prescriptions in a specific logbook that is checked before future consultations or prescriptions to avoid conflicting medications. Financial organisations use a formal process called the *know your customer* or KYC to obtain and maintain information about their customers. Traditionally, each organisational entity asking for such data has to obtain and verify the data independently, and keep copies of it for future references and legal obligations. The users, on the other hand, do not have a mechanism to selectively control the release of such sensitive information. A centralised or decentralised registry of such information

A. Basu is also a Visiting Research Fellow at the University of Sussex, UK and Rutgers University, USA.

helps with speeding up user identity verification, but it raises questions about privacy of the data in the hands of this third-party registry.

Our Contribution: In order to simplify the process for a service provider to collect, verify and maintain user information, a centralised (or distributed) solution works where the user is in charge of providing the information and a trusted third party maintains irreversible proofs of the components of such information. In this *position paper*, we propose a framework – the **Verifiable Information Graph** or the VIGraph – based on a generalised hash tree as a data structure. A hash tree enables verifiability of individual sub-trees without the full knowledge of the data contents. From the user’s perspective, this very property of the hash tree enables releasing, for verification, parts of the information for specific checks without compromising the privacy of the rest. For the sake of brevity, we focus on user profile data to describe our proposed framework. The framework can also be applied to other types of information with similar verifiability requirements; we leave this as an avenue for future work.

Paper Organisation: The rest of the paper is organised as follows. In Sect. 2, we propose the structure of the VIGraph and explain the operations on it. This is followed by a brief discussion and a description of the relevant state-of-the-art in Sect. 3, before concluding with future directions in Sect. 4.

2 VIGraph: Signed Hash Tree with Optional Dependency Overlay

Hash trees, and the specialised Merkle Trees [1] have a specific property: each sub-tree of the tree can be verified independently of the other. This facilitates obfuscation on the actual contents of the data in sub-trees, which the verifier is not interested in. Our proposed data structure, the VIGraph, is four-levels deep, as illustrated in Fig. 1. Starting at the top level (root), the actual data is on the fourth level down in the leaf nodes. VIGraph is based on a generalised hash tree, which means that a non-leaf node can have more than two children, as opposed to a binary hash tree. The structure of the VIGraph closely resembles a tree but is technically a graph due to the optional dependency overlay. However, throughout the paper, we will use terminology akin to trees, such as leaf nodes and root. The purpose of the VIGraph is verifiability and selective information release; and not a fast search from the root. Thus, VIGraph is not to be confused or compared with hash tries or hash array mapped tries (HAMT).

The entities involved with the VIGraph are: (a) the user u whose information is being maintained, (b) organisational entities (each represented as o) responsible for endorsing (through digital signatures) various components of the user’s information, e.g., the driving license identifier of a user may be signed by the driving license issuing authority, (c) the metadata provider organisation(s) (each represented as p) responsible for maintaining and facilitating verification using

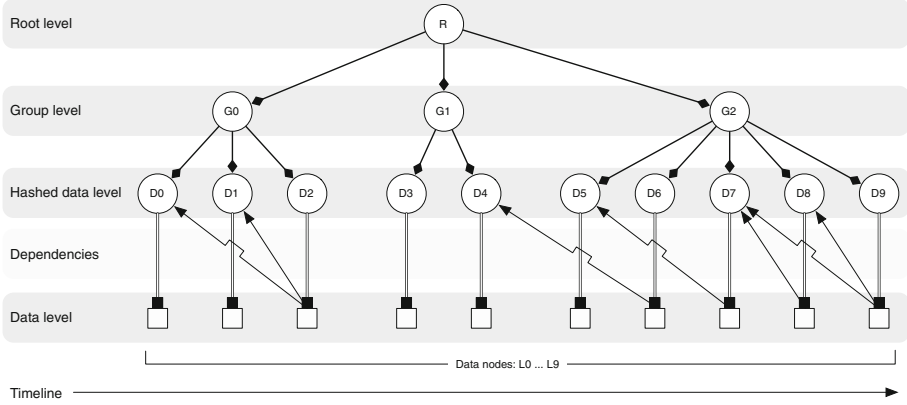


Fig. 1. Structure of a Verifiable Information Tree (VIGraph). The dependencies are optional. One-to-one, one-to-many, many-to-one and cross-group dependencies are illustrated.

the VIGraph; and (d) the service provider or a verifier organisation (represented as v) responsible for verifying information against that stored in the VIGraph.

In short, each user has her VIGraph representing information about her, parts of which are maintained by one or more metadata providers; and the information maintained by the graph may or may not be endorsed by relevant information issuing authorities.

2.1 Levels and Dependencies

Figure 2 illustrate the contents of the nodes at various levels of the VIGraph.

Data Level: The leaf nodes $L_0 \dots L_9$ (refer to Fig. 1) are the actual data, L_i , such as (using the personal profile example) national identity information, residential addresses, driving license and so on. Each node may also contain optional dependency pointers, described later.

Hashed Data Level: The hashed data nodes, $D_0 \dots D_9$, on the *hashed data level*, contain the hashes of the data node contents, such that $D_i = h(L_i, r_u)$ where $h(L_i, r_u)$ is some cryptographically secure one-way hash function that computes the hash on data L_i along with some random number r_u . The random value, r_u is different for different users. This ensures that the hash of the same information, e.g., a certain residential address is different for different users, thus thwarting linkability of information if it belongs to two different users. In addition to the hashes, each hashed data node also contains up to two signatures for the corresponding hash value. The mandatory signature is from the user, $sig_u(h(L_i, r_u))$, while the other optional signature is from the respective organisation, $sig_o(h(L_i, r_u))$, in charge of issuing the data as shown in Fig. 2.

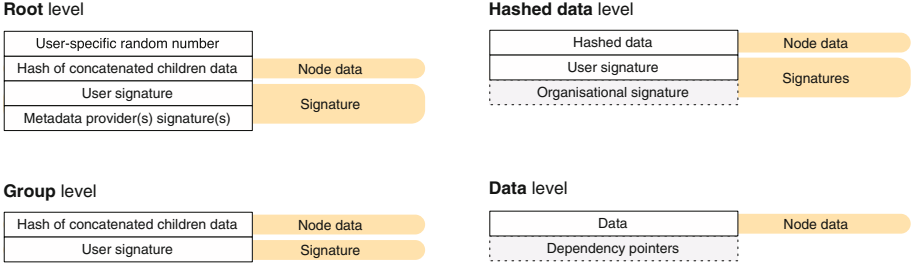


Fig. 2. Contents of the nodes at various levels of the VIGraph.

For example, in Japan, if L5 denotes the user’s current residential address then D5 will contain the hash of L5, the signature from the user and the signature from the respective city or ward office endorsing the information.

Group Level: In the *group level*, the group nodes are defined by the user and are useful for semantic grouping as well as privacy during verification. For instance, national identity information may be in a group with visa information while residential addresses could be in a different group. Each group contains the hash of the concatenation of the hash values (of the actual data) stored in its children data nodes. The newest child in the group appears rightmost in the concatenation. Each group also contains the user’s signature of its contents.

Root Level: The *root level* node contains a similar concatenation of its children group nodes. The root node also contains the user-specific random number, r_u , a signature from the user and another from the metadata provider maintaining the graph.

Dependency Overlay: The dependencies can exist between the data level nodes and the hashed data level nodes, with one-way pointers from the data level to the hashed data level. Dependencies can help retrieve related data, are defined when adding new data and are immutable after that. For example, if a user holding the nationality of country C_1 acquires a work-permit in country C_2 to live and work there, then the data node for the work permit may have an optional pointer to the signed data for the citizenship. This signifies the semantic dependency that the visas depend on a specific nationality. While declaring the optional dependency is the user’s choice, the signing authority, o , for the corresponding hashed data level node may reject a certain declared dependency if it is deemed irrelevant, or conversely require the declaration of a specific dependency. The dependencies can be one-to-one, one-to-many, many-to-one, many-to-many and cross-group. All dependencies are backward links, in terms of time.

2.2 Storage of the VIGraph

Parts of the VIGraph will be stored by the user as well as the metadata provider(s). There can be more than one metadata provider, in which case the VIGraph may be stored across multiple metadata providers with erasure coding to ensure reliability. The user will store everything but the root whereas the metadata provider will store everything but any node from the data level – thus, the user is responsible for storing the actual data. This allows the graph to be verified in part or full against the version stored by the metadata provider(s) without having them to also maintain a copy of the actual data, which can contain sensitive information.

2.3 Data Operations on the VIGraph

Add Operation: The add operation enables adding leaf nodes at the data level to an empty graph or an existing one. In both cases, we assume that the user has already generated a public key, private key pair.

Figure 3(a) illustrates the steps the user needs to take to add a new data node to an empty, i.e., non-existent graph. The states: *Obtain organisational signature* and *Publication* depend on validation from the organisational signatory, o and the metadata provider, p , respectively. The entity o may not provide the optional signature. Figure 3(b) illustrates the steps the user needs to take to add a new data node to an existing graph. Depending on whether the new data is being added to a new group or an existing group, the add operation will require the validation, by the metadata provider, of the optional group hash update and the root hash update. The user must prove to the metadata provider the knowledge of the existing group and root hashes, and the fact that those existing hashes indeed update

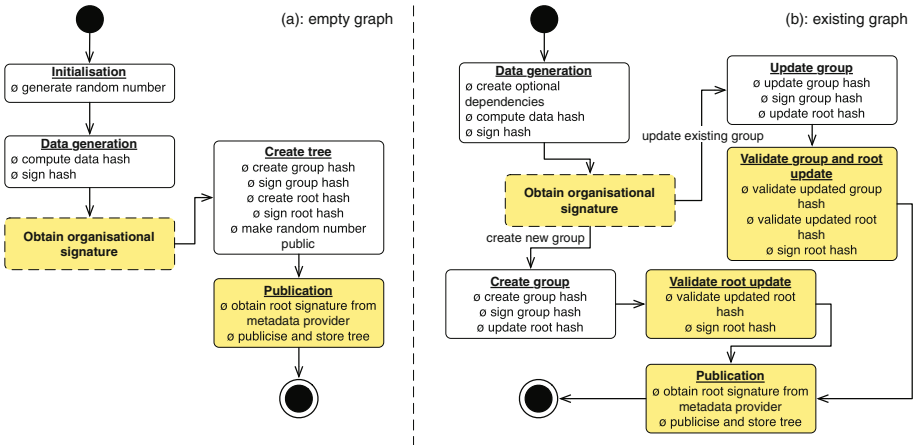


Fig. 3. The user-side state diagrams for the add data operation to: (a) an empty graph, and (b) an existing graph.

to the new ones when the new data is added. In both cases of adding a node to a new graph or an existing graph, the metadata provider may not sign the root hash of the graph if it fails to check, including offline mechanisms beyond the scope of this paper, the validity of the information being added.

Delete Operation: The user can only ‘remove’ a data leaf node from local storage. This does not delete the corresponding hashed data node but it will ensure that the original data node is no longer recoverable. The hashed data node can still be used in verification of other data nodes without compromising the privacy of the actual data that it encapsulates. Removing the organisational signature during the verification process further ensures that the verifier is unable to guess what the original data deleted node was, given its hashed value only.

Group Restructuring: Group structuring allows one data leaf node and its corresponding hashed data node to be moved from one group to another. The move operation is effectively a delete operation followed by an add operation. Since the node being moved is not removed from the graph, moving groups may result in inter-group dependencies. Suppose D_x denotes the concatenated hashes of all the other sibling data hash nodes (in the hashed data level), i.e., the concatenations of some $D_x = h(L_i, r_u) || \dots || h(L_{k-1}, r_u)$. To delete an existing node L_k from group G_k , the user needs to prove, to the metadata provider, that the hash contained in the existing G_k is $h(D_x || h(L_k, r_u))$. Once validated, the hash contained in G_k that has been updated to $h(D_x)$ is signed by the user. The rest of the move operation is just the same as an add operation to an existing graph, except that $h(L_k, r_u)$ will already have the necessary signatures on it.

2.4 Data Verification

The verification process is illustrated in Fig. 4. The verifier, e.g., a service provider, requests two sets of data; one from the user and the other from the metadata provider(s). Note that only the signed root hash and the user-specific random number are the required information from the metadata provider and the

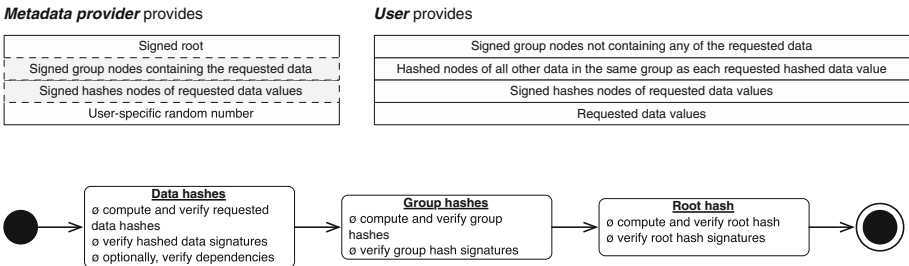


Fig. 4. The required information and the process for verification.

rest are optional. The verification process has three stages. Firstly, the verifier computes the hashes on the data provided by the user and cross-checks if those hashes match the ones obtained from the user and the metadata provider(s), and that their signatures are correct. The verifier can also check the optional dependencies at this stage. Secondly, the verifier uses information of all other provided data hashes to compute group hashes, and check if these and their signatures match those obtained from the user and the metadata provider. Finally, it computes and verifies the root hash in the same way and checks against the one obtained from the metadata provider(s). It is evident that in order to verify an information component (in the data level), it is necessary for the user to provide information of the hashed data level of all the other children of the group. The larger the number of children, the more inefficient this is. This shows, as indicated before, that having one or few groups to represent all data is inefficient although the framework will still work.

3 Discussion and the State-of-the-art

Blockchains [2,3] have been used [4] to store anonymised but verifiable information for identity and access control, including commercial work from KPMG¹ and Deloitte². Blockchains are considered as the so-called ‘trustless’ systems that remove the need to trust any entity in particular because a decentralised consensus mechanism provides majority opinion. The idea of a trustless system is questionable because the verifier has to trust the accuracy of the consensus mechanism and the underlying hash function instead of any specific entities. With a public blockchain involving a very large number of participants, the probability of collusion and hence alteration of majority opinion is low but not impossible. With a private blockchain involving a limited number of participants, the risk of collusion is higher. Furthermore, public blockchains are not scalable since all participants have to maintain a very large hash chain, and agree on a consensus. In addition, maintaining the information of every user in a single blockchain may prove wasteful in certain situations. While the fault tolerance of blockchains is often attributed to their decentralised nature, it is to be noted that other decentralised means of storage of information, e.g., erasure coding, also achieves fault tolerance without the downsides of the consensus algorithm. In our work, we show that hash trees alone can be sufficient for verifiability of selectively released information.

Selective release of personal information is a well-studied subject. Kiyomoto et. al’s work on privacy policy manager [5] discusses a framework that enables interpreting privacy policies easier for users, which has been standardised by the oneM2M initiative [6]. Sanitizable signatures schemes [7–9] allow a semi-trusted third party called sanitizer to sign on a modified message using just the public

¹ See: <https://goo.gl/oaECro>.

² See: <http://www.deloitte.co.uk/smartid/> and <https://github.com/SmartIdentity/smartId-contracts>.

key from the signer without interacting with her. The constraint is that the modifications lie in the predetermined parts of the original message by the signer. This kind of signatures is useful when the message owner wants to release multiple versions of a message. It is applicable in the situation where some sensitive information of the message need anonymisation. However, it can not be directly applied to the case of verification with selective release, which is the concern of our work. Interestingly, sanitizable signatures can be viewed as a work orthogonal to ours. As a future direction, we may utilise it in our VIGraph proposal to provide better privacy protection. Aggregate signatures [10] can also be utilised to tackle the requirements of KYC since it allows the addition of signatures. The signatures of n messages from n different signers can be aggregated into a single signature, which can verify the integrity of all the n messages. Attribute-based access control [11] can provide fine-grained access credentials to different entities, but is different from verifiability of fine-grained information. Leung and Mitchell [12] proposed a privacy preserving authentication protocol, which allows a service provider to successfully identify the user without breaching the user's private information. Direct Anonymous Attestation (DAA) schemes are special signatures which provide a balance between signer authentication and privacy.

4 Conclusion and Future Work

In this paper, we have presented a generalised hash tree based framework for verifying selectively-released information through a trusted third party, which stores different parts of the hash graph but not the actual data. We have discussed how our framework enables selective release of data for verification, thus helping with privacy. Throughout this paper, we have used personal profile data as an example but our framework can be applied to other types of data. Beyond what we have identified as future work in the paper, we aim to refine the deletion operation in our framework to make it compliant with right-to-forget. We also plan to implement a prototype, and run user and performance evaluations; and compare our proposal more extensively with blockchain based alternatives.

References

1. Merkle, R.C.: Method of providing digital signatures (1979). <https://www.google.com/patents/US4309569>
2. Nakamoto, S.: Bitcoin: a peer-to-peer electronic cash system (2008)
3. Swan, M.: Blockchain: Blueprint for a New Economy. O'Reilly Media, Inc., Sebastopol (2015)
4. Zyskind, G., Nathan, O., et al.: Decentralizing privacy: using blockchain to protect personal data. In: Security and Privacy Workshops (SPW), pp. 180–184. IEEE (2015)
5. Kiyomoto, S., Nakamura, T., Takasaki, H., Watanabe, R., Miyake, Y.: PPM: privacy policy manager for personalized services. In: Cuzzocrea, A., Kittl, C., Simos, D.E., Weippl, E., Xu, L. (eds.) CD-ARES 2013. LNCS, vol. 8128, pp. 377–392. Springer, Heidelberg (2013). doi:10.1007/978-3-642-40588-4_26

6. Datta, S.K., Gyrard, A., Bonnet, C., Boudaoud, K.: oneM2M architecture based user centric IoT application development. In: 2015 3rd International Conference on Future Internet of Things and Cloud (FiCloud), pp. 100–107. IEEE (2015)
7. Ateniese, G., Chou, D.H., Medeiros, B., Tsudik, G.: Sanitizable signatures. In: Vimercati, S.C., Syverson, P., Gollmann, D. (eds.) ESORICS 2005. LNCS, vol. 3679, pp. 159–177. Springer, Heidelberg (2005). doi:[10.1007/11555827_10](https://doi.org/10.1007/11555827_10)
8. Miyazaki, K., Hanaoka, G., Hideki, I.: Invisibly sanitizable digital signature scheme. IEICE Trans. Fundam. Electron. Commun. Comput. Sci. **91**(1), 392–402 (2008)
9. Brzuska, C., Fischlin, M., Freudenreich, T., Lehmann, A., Page, M., Schelbert, J., Schröder, D., Volk, F.: Security of sanitizable signatures revisited. In: Jarecki, S., Tsudik, G. (eds.) PKC 2009. LNCS, vol. 5443, pp. 317–336. Springer, Heidelberg (2009). doi:[10.1007/978-3-642-00468-1_18](https://doi.org/10.1007/978-3-642-00468-1_18)
10. Boneh, D., Gentry, C., Lynn, B., Shacham, H.: Aggregate and verifiably encrypted signatures from bilinear maps. In: Biham, E. (ed.) EUROCRYPT 2003. LNCS, vol. 2656, pp. 416–432. Springer, Heidelberg (2003). doi:[10.1007/3-540-39200-9_26](https://doi.org/10.1007/3-540-39200-9_26)
11. Goyal, V., Pandey, O., Sahai, A., Waters, B.: Attribute-based encryption for fine-grained access control of encrypted data. In: Proceedings of the 13th ACM Conference on Computer and Communications Security, pp. 89–98. ACM (2006)
12. Leung, A., Mitchell, C.J.: Ninja: non identity based, privacy preserving authentication for ubiquitous environments. In: Krumm, J., Abowd, G.D., Seneviratne, A., Strang, T. (eds.) UbiComp 2007. LNCS, vol. 4717, pp. 73–90. Springer, Heidelberg (2007). doi:[10.1007/978-3-540-74853-3_5](https://doi.org/10.1007/978-3-540-74853-3_5)

A Flexible Privacy-Preserving Framework for Singular Value Decomposition Under Internet of Things Environment

Shuo Chen¹(✉), Rongxing Lu², and Jie Zhang¹

¹ Nanyang Technological University, Singapore, Singapore
chen1087@e.ntu.edu.sg, zhangj@ntu.edu.sg

² Faculty of Computer Science, University of New Brunswick, Fredericton, Canada
rxlu@ieee.org

Abstract. The singular value decomposition (SVD) is a widely used matrix factorization tool which underlies many useful applications, e.g. recommendation system, abnormal detection and data compression. Under the environment of emerging Internet of Things (IoT), there would be an increasing demand for data analysis. Moreover, due to the large scope of IoT, most of the data analysis work should be handled by fog computing. However, the fog computing devices may not be trustable while the data privacy is the significant concern of the users. Thus, the data privacy should be preserved when performing SVD for data analysis. In this paper, we propose a privacy-preserving fog computing framework for SVD computation. The security and performance analysis shows the practicability of the proposed framework. One application of recommendation system is introduced to show the functionality of the proposed framework.

1 Introduction

With the prosperous development of communication and computation technologies, the Internet of Things (IoT) is no longer a fantasy nowadays. The big advantage of IoT is that through analyzing the huge amount of information collected from the physical world, the server is capable of making better decisions which would produce considerable benefits. It is estimated that the number of devices connected to the Internet would be about 50 billion by 2020 [1]. It could be anticipated that rather than being conducted on the cloud or inside the intranet of companies, the data analysis work would be performed everywhere and anytime in the future due to the ubiquitousness of IoT. As the amount of data analysis tasks increases in IoT, the singular value decomposition (SVD), which is widely used in different data analysis applications [2, 12, 13, 15, 21], will be performed frequently. However, the traditional way of performing SVD, i.e. calculating the SVD in central server, may not be practical in future IoT due to the vast number of IoT devices. If all the data is transmitted to a central server

for computation, it would lead to considerable computation and communication resource consumption in the server, which would further severely impact the quality of service (QoS) of IoT applications.

To ease the burden of the IoT server and guarantee the QoS, a new technique called fog computing, which is proposed by Cisco [4], is suitable to be applied. The main idea of fog computing is to provide storage, computing and networking services between environmental devices and the central server. The fog devices which are in close proximity to end devices normally possess a certain amount of storage and computation resource. With the equipped resource, the fog devices could process the collected data locally so as to loose the workload of the server. In specific, there are three tiers in the fog computing architecture: environmental tier, edge tier and central tier. In the environmental tier, there are billions of heterogeneous IoT devices collecting and uploading information of the physical world, e.g. medical sensors in eHealth and the mobile phone of each person. The data collected by IoT devices will be transmitted to the edge tier. The fog devices in the edge tier could perform the application-specific operations on received data locally and send the results to the server in the central tier. Owing to the processing of fog devices, the volume of data sent to server could be reduced to a large extent. Since the fog devices are spread in a highly distributed environment, it is impractical for an institution which owns the central server to provide and maintain all those fog devices. Therefore, it is reasonable to assume that the fog devices would be supplied by third parties.

Under the context of fog computing, one could perform the SVD operations on the fog devices. However, another problem which would appear is the privacy issue. The third parties which control the fog devices may not be trustworthy while in many IoT applications, the data collected from the environment is considered as private by the users, e.g. the vital signs in eHealth, the location of vehicles, and the power usage in a smart grid. Performing the SVD on plaintext with fog devices is infeasible if the privacy is a primary concern from the perspective of data owners. Therefore, how to take advantage of fog computing to locally process data in a privacy-preserving way is a challenging issue.

In this paper, we propose a flexible fog computing framework for performing SVD with privacy preserved. The homomorphic encryption technique called Paillier encryption [18] is applied to protect the data privacy. The framework is designed to be capable of supporting different applications based on the SVD computation. The main contributions of this paper are three-fold.

- First, we propose a fog computing framework for privacy-preserving SVD computation to ease the burden of server and protect the data privacy from the fog devices which may not be trustable.
- Second, there is only one communication round between the data providers and data processors in our work while most of the existing works require iterative communications, which brings heavy overhead.
- Third, one application is introduced in details to demonstrate the functionality of the framework. It has been shown that the proposed framework could be

easily adopted by the applications which build the trust of unknown entities based on the third-party recommendations.

The remainder of this paper is organized as follow. In Sect. 2, the preliminaries of our scheme are introduced. The system model, security requirements and design goals are described in Sect. 3. In Sect. 4, the proposed framework is presented in details. The security analysis and performance evaluation are discussed in Sects. 5 and 6. One application based on the proposed privacy-preserving SVD framework is illustrated in Sect. 7. In Sect. 8, we discuss the related work, and finally conclude our current work in Sect. 9.

2 Preliminaries

In this section, the Paillier Cryptosystem [18] and Singular Value Decomposition [9] which are the basis of the proposed framework are reviewed.

2.1 Paillier Cryptosystem

The Paillier Cryptosystem enables the addition operation on plaintext through the manipulation of ciphertext. This homomorphic property is extensively desired in many privacy-preserving applications [16, 20, 25]. In this paper, this feature allows fog devices to process the user data in encrypted form without leaking the data content. The knowledge of Paillier Cryptosystem required for this work is introduced as follow and more details could be referred to [18].

Key Generation: Given one security parameter κ , the public key $\mathbf{PK} = (n, g)$ and private key \mathbf{SK} could be generated, where the bit length of n is 2κ .

Encryption: Given a message $m \in \mathbb{Z}_n$, randomly choose a number $r \in \mathbb{Z}_n^*$, the ciphertext could be calculated as $c = E(m, r) = g^m \cdot r^n \bmod n^2$.

Decryption: Given a ciphertext $c \in \mathbb{Z}_{n^2}^*$, the plaintext $m = D(c, \mathbf{SK})$.

Homomorphic Property: $E(m_1, r_1) \cdot E(m_2, r_2) = E(m_1 + m_2, r_1 \cdot r_2)$.

2.2 Singular Value Decomposition

SVD is a powerful and popular matrix factorization tool that underlies plenty of useful applications, e.g. abnormal detection [12, 15], recommendation system [2, 21] and data compression [13]. Let \mathbb{A} be an $l \times N$ matrix, the SVD of \mathbb{A} is of the form $\mathbb{U}\Sigma\mathbb{V}^T$ where T means conjugate transpose. Σ is an $l \times N$ rectangular diagonal matrix of which diagonal entries are the singular values of \mathbb{A} . \mathbb{U} is an $l \times l$ unitary matrix and \mathbb{V} is an $N \times N$ unitary matrix. The columns of \mathbb{U} (\mathbb{V}) are the left(right)-singular vectors of \mathbb{A} .

Another widely used matrix factorization tool is the eigenvalue decomposition. It is closely related to SVD as shown below:

$$\mathbb{A} \cdot \mathbb{A}^T = \mathbb{U}\Sigma\mathbb{V}^T\mathbb{V}\Sigma^T\mathbb{U}^T = \mathbb{U}\Sigma\Sigma^T\mathbb{U}^T, \quad \mathbb{A}^T \cdot \mathbb{A} = \mathbb{V}\Sigma^T\mathbb{U}^T\mathbb{U}\Sigma\mathbb{V}^T = \mathbb{V}\Sigma^T\Sigma\mathbb{V}^T \quad (1)$$

Equation (1) shows that \mathbb{U} is the eigenvectors of $\mathbb{A} \cdot \mathbb{A}^T$, \mathbb{V} is the eigenvectors of $\mathbb{A}^T \cdot \mathbb{A}$ and the singular values in Σ are the square root of the eigenvalues of $\mathbb{A} \cdot \mathbb{A}^T$ and $\mathbb{A}^T \cdot \mathbb{A}$. We will show that the above relation could be utilized to achieve the privacy-preserving SVD in the later sections.

3 System Model, Security Requirements and Design Goals

In this section, we describe the system model, discuss the security requirements and identify the design goals on privacy-preserving SVD.

3.1 System Model

In this work, we mainly focus on how to utilize the fog computing to compute the SVD of the uploaded data with privacy preserved. Specifically, there are four categories of entity in the system model, namely server, first layer fog device, second layer fog device and environmental device as shown in Fig. 1.

Server: Server is a fully trustable entity located in the remote control tier. It is responsible for initializing the whole system and distributing key materials to others. The other operations the server may conduct are application-specific.

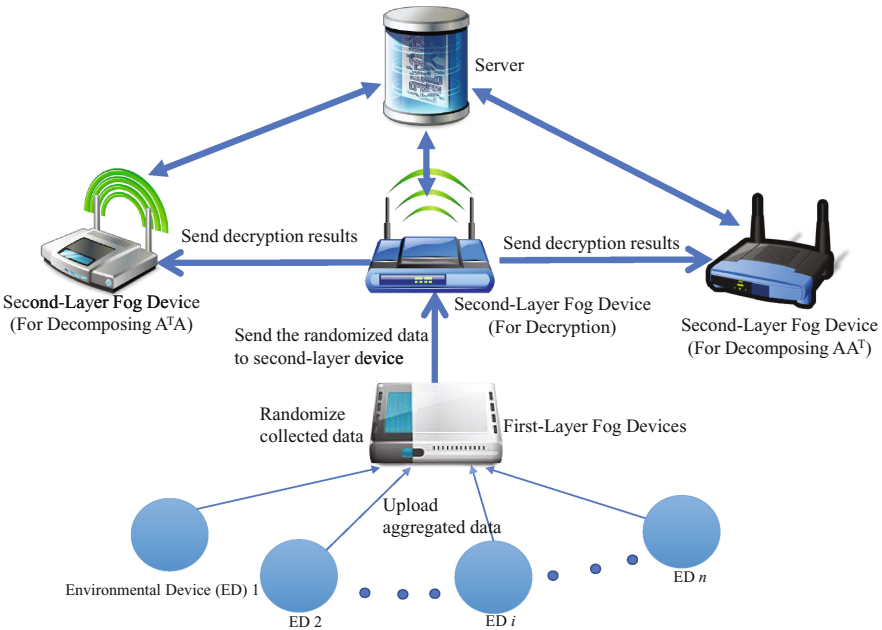


Fig. 1. System model

Environmental Device (ED): EDs are the devices distributed in the environmental layer of IoT environment. The analysis on the data uploaded by EDs could enable better decision-making.

First Layer Fog Device (FD): FDs are the fog devices which communicate with EDs directly. FDs process the collected data and upload the results to the second layer fog devices.

Second Layer Fog Device (SD): SDs are the fog devices which communicate with FDs. Compared to FDs, SDs are closer to the server and do not contact with EDs directly. In the proposed framework, there are three SDs playing different roles for SVD operation. One of them is responsible for decrypting the messages from FDs. The other two are in charge of decomposing $\mathbb{A} \cdot \mathbb{A}^T$ and $\mathbb{A}^T \cdot \mathbb{A}$. We denote the one for decryption, decomposing $\mathbb{A} \cdot \mathbb{A}^T$ and decomposing $\mathbb{A}^T \cdot \mathbb{A}$ as SD_d , SD_u and SD_v respectively.

Note that the hierarchical distribution of fog devices is a characteristic inherited from the traditional network architecture. For example, the switchers could function as the first layer fog devices and the gateways in the higher layer could serve as the second layer fog devices.

3.2 Security Requirements

Security is fundamental for the effectiveness of proposed framework. In this work, the server and EDs are assumed to be trustable. The fog devices, i.e. FDs and SDs, are assumed to be honest-but-curious [8, 23] which means they will follow the specified procedures faithfully while being curious about the uploaded data. In addition, FDs and SDs are assumed not to collude with each other. The non-collusion assumption could be realized similarly as the EigenTrust scheme [14]. Briefly speaking, for each SVD computation, the server chooses fog devices based on distributed hash table. Due to the large number of fog devices, it is infeasible for the device providers to determine whether they would be selected for the same computation and negotiate for collusion in advance.

Based on the above assumptions, the confidentiality as the security requirement should be fulfilled, i.e. even FDs and SDs process the collected data, they could not learn anything about the actual value of data. For authenticity and integrity, since there are many existing signature schemes, e.g. Boneh-Lynn-Shacham (BLS) short signature [3], this work just focuses on confidentiality.

3.3 Design Goals

According to the aforementioned system model and security requirements, the proposed framework should achieve the following objectives.

- *The confidentiality should be guaranteed in the proposed framework.* All the user data contained in the transmitted messages should be protected. The processing in fog devices should not leak data privacy.

- *The framework should be flexible enough to be adopted by different applications.* Instead of being the ultimate goal, SVD is the basis or initial step of many applications, which means the further procedures after SVD could be quite different for various scenarios. Therefore, the design of the framework should consider the flexibility such that the results of SVD could be further utilized to achieve the final purposes of different applications.

4 The Proposed Framework

In this section, the proposed framework for SVD computation is presented in details. The framework is composed of five phases: system initialization, data collection, data randomization, pre-computation and eigenvalue decomposition.

4.1 System Initialization

The server is the trustable entity which bootstraps the whole system. Assume the amount of users supported by the system is N , each user data is l -dimensional and the range for each dimension value is $[0, d]$ where d is a constant. The system parameters are $\kappa, \kappa_1, \kappa_2$ and κ_3 . Let $|\bullet|$ denote the bit length of \bullet . Given the parameter κ , the server calculates the **PK**: (n, g) , where $|n| = 2\kappa$, and the corresponding **SK**. Given the parameters κ_1, κ_2 and κ_3 , let $t = 2^{\kappa_1}$, the server randomly chooses two coprime integers W and S such that $W > \max(N, l) \cdot d^2$ and $S > \max(N, l) \cdot (d^2 + 2tWd + t^2W^2)$, where $|W| = \kappa_2$ and $|S| = \kappa_3$. Then, the server chooses one superincreasing sequence $\vec{a} = (a_1 = 1, a_2, \dots, a_l)$ such that $\sum_{j=1}^{i-1} a_j \cdot (d + tW + tS) < a_i$ for $i = 2, \dots, l$ and $\sum_{i=1}^l a_i \cdot (d + tW + tS) < n$. Finally, the server publishes $\{n, g, \vec{a}\}$ as public parameters, sends **SK** to SD_d as secret, and sends (W, S) to FDs, SD_u and SD_v as secret respectively.

4.2 Data Collection

In the environmental tier, the data uploaded from N EDs could form the data matrix \mathbb{A} . To compute the SVD of matrix \mathbb{A} is the goal of this framework. The i th column of matrix \mathbb{A} $(d_{1i}, \dots, d_{li})^T$ is from the i th device ED_i . To upload the data, ED_i performs the following steps:

- *Step-1.* Utilize the superincreasing \vec{a} to compute

$$m_i = a_1 d_{1i} + a_2 d_{2i} + \dots + a_l d_{li} \quad (2)$$

- *Step-2.* Choose a random number $r_i \in \mathbb{Z}_n^*$ and compute

$$C_i = g^{m_i} \cdot r_i^n \bmod n^2 \quad (3)$$

- *Step-3.* Send the data $C_i || ED_i$ to the FD which communicates with it.

4.3 Data Randomization

For each FD, it will perform the following steps to randomize the received data.

- *Step-1.* For the i th data C_i , FD chooses $2 \cdot l$ random numbers which are $(z_{1i}, \dots, z_{li})^T$ and $(r_{1i}, \dots, r_{li})^T$ from the range $[1, t]$. Then FD computes $rz = \sum_{k=1}^l a_k \cdot (z_{ki} \cdot W + r_{ki} \cdot S)$.
- *Step-2.* FD randomizes C_i as

$$C_i' = C_i \cdot g^{rz} \bmod n^2 = g^{\sum_{k=1}^l a_k \cdot (d_{ki} + z_{ki} \cdot W + r_{ki} \cdot S)} \cdot r_i^n \bmod n^2 \quad (4)$$

- *Step-3.* FD sends the randomized data C_i' to SD_d .

4.4 Pre-computation

Upon receiving N data from FDs, SD_d will perform the following steps to compute the randomized $\mathbb{A}\mathbb{A}^T$ and $\mathbb{A}^T\mathbb{A}$.

- *Step-1.* For each C_i' , SD_d decrypts it with \mathbf{SK} and gets the aggregated data

$$m_i' = \sum_{k=1}^l a_k \cdot (d_{ki} + z_{ki} \cdot W + r_{ki} \cdot S) \bmod n \quad (5)$$

- *Step-2.* Through the **Algorithm 1** in [6] which is the detailed version of this work, SD_d could recover the randomized value for each dimension of data i .
- *Step-3.* From each m_i' , SD_d could get an l -dimensional randomized data. In total, SD_d could get the randomized $l \times N$ data matrix \mathbb{A}' , in which the (i, j) th entry is $d'_{ij} = d_{ij} + z_{ij} \cdot W + r_{ij} \cdot S$. Then SD_d simply computes $\mathbb{A}' \cdot (\mathbb{A}')^T$ and $(\mathbb{A}')^T \cdot \mathbb{A}'$, and sends the two resulting matrices to SD_u and SD_v respectively.

4.5 Eigenvalue Decomposition

SD_u : When receiving $\mathbb{A}' \cdot (\mathbb{A}')^T$, SD_u will perform the following steps to compute the left part of the SVD for matrix \mathbb{A} , i.e. matrix \mathbb{U} and Σ .

- *Step-1.* For each entry e_u' of $\mathbb{A}' \cdot (\mathbb{A}')^T$, SD_u derandomizes the entry as follow:

$$e_u = e_u' \bmod S \bmod W \quad (6)$$

The result e_u is the corresponding entry of matrix $\mathbb{A} \cdot \mathbb{A}^T$.

- *Step-2.* After SD_u recovers the matrix $\mathbb{A} \cdot \mathbb{A}^T$, it performs eigenvalue decomposition for matrix $\mathbb{A} \cdot \mathbb{A}^T$ and gets the matrix \mathbb{U} and Σ .

SD_v : Similar as SD_u , SD_v performs eigenvalue decomposition on the recovered $\mathbb{A}^T \cdot \mathbb{A}$ to get the right part of the SVD for matrix \mathbb{A} , i.e. matrix \mathbb{V} and Σ .

By now, the SVD of matrix \mathbb{A} has been separately held by SD_u and SD_v .

The correctness of derandomization: The (i, j) th entry $e_{u'ij}$ of $\mathbb{A}' \cdot (\mathbb{A}')^T$ is implicitly formed as

$$\begin{aligned}
 e_{u'ij} &= \sum_{k=1}^N d'_{ik} \cdot d'_{jk} = \sum_{k=1}^N (d_{ik} + z_{ik} \cdot W + r_{ik} \cdot S) \cdot (d_{jk} + z_{jk} \cdot W + r_{jk} \cdot S) \\
 &= \sum_{k=1}^N d_{ik}d_{jk} + \sum_{k=1}^N [(z_{ik}d_{jk} + z_{jk}d_{ik})W + z_{ik}z_{jk}W^2] \\
 &\quad + S \sum_{k=1}^N [(r_{ik}d_{jk} + r_{jk}d_{ik}) + (z_{ik}r_{jk} + z_{jk}r_{ik})W + r_{ik}r_{jk}S]
 \end{aligned} \tag{7}$$

Since

$$\sum_{k=1}^N d_{ik}d_{jk} + \sum_{k=1}^N [(z_{ik}d_{jk} + z_{jk}d_{ik})W + z_{ik}z_{jk}W^2] < N(d^2 + 2tdW + t^2W^2) < S,$$

we have $e_{u'ij} \bmod S = \sum_{k=1}^N d_{ik}d_{jk} + \sum_{k=1}^N [(z_{ik}d_{jk} + z_{jk}d_{ik})W + z_{ik}z_{jk}W^2]$. Also, we have $\sum_{k=1}^N d_{ik}d_{jk} < Nd^2 < W$. Thus, $(e_{u'ij} \bmod S) \bmod W = \sum_{k=1}^N d_{ik}d_{jk}$ which is the (i, j) th entry of $\mathbb{A} \cdot \mathbb{A}^T$. Similarly, for the entry $e_{v'ij}$ of matrix $(\mathbb{A}')^T \cdot \mathbb{A}'$, $(e_{v'ij} \bmod S) \bmod W = \sum_{k=1}^l d_{ki}d_{kj}$ which is the (i, j) th entry of matrix $\mathbb{A}^T \cdot \mathbb{A}$.

5 Security Analysis

In this section, the privacy leakage during normal procedures and the potential attacks which could be conducted by certain participants to snoop data are analyzed. The resistance of the framework against those attacks is discussed and the principles for system configuration are demonstrated.

5.1 Privacy Leakage Under Normal Operations

In the proposed framework, each data is encrypted with Paillier Cryptosystem and SD_d is the only fog device which has the private key for decryption. Therefore, the data of each ED could not be discovered by the other EDs and FDs. For SD_d , it could only get the randomized data, i.e. $d_{ij}' = d_{ij} + z_{ij}W + r_{ij}S$, for $i = 1, \dots, l$ and $j = 1, \dots, N$ and learn nothing about the real value since SD_d does not know W and S . For SD_u , it could get \mathbb{U} and Σ during normal operations. However, it needs the correct unitary matrix \mathbb{V} to recover \mathbb{A} . Since there are infinite unitary matrices, SD_u could not learn original \mathbb{A} with only \mathbb{U} and Σ . Similarly, SD_v could not recover data matrix \mathbb{A} with only \mathbb{V} and Σ .

Based on the above analysis, the data privacy is preserved when the participants follow the defined procedures. In the following, the possible extra computations performed by participants to discover private data are considered.

5.2 Potential Attacks

Since EDs and FDs only have encrypted data, they could not gain much no matter what operations they perform on the ciphertext. Therefore, we mainly discuss the potential attacks from SD_d , SD_u and SD_v in this part.

• **SD_d :** As mentioned above, the information SD_d gets is the randomized data $d_{ij}' = d_{ij} + z_{ij}W + r_{ij}S$, for $i = 1, \dots, l$ and $j = 1, \dots, N$. What SD_d needs to do is to find the value of S and W and recover the original data as

$$d_{ij} = d_{ij}' \bmod S \bmod W \quad (8)$$

Since d_{ij} is mixed with the random combination of S and W , it is infeasible for SD_d to determine S and W without additional information. Therefore, we consider the situations in which SD_d knows some of the user data. With the knowledge of user data, the possible operations SD_d could do are as follows:

- *Step-1.* For each known data, SD_d converts the corresponding randomized data to the form $zW + rS$ by computing $d' - d$. Let LC denote the set of converted data and $LC_i = z_iW + r_iS$ denote the i th element of LC .
- *Step-2.* SD_d performs the brute force attack, i.e. tries all possible S . For each try, SD_d performs (modulo S) operation on each LC_i . Then SD_d computes the greatest common divisor (GCD) of the resulting set. If the GCD is larger than 1, it is the value of W and the currently selected S is the correct S .

The rationale behind this attack is: the probability of k randomly chosen integers being coprime is $\frac{1}{\zeta(k)}$, where $\zeta(x)$ is Riemann zeta function [17]. When k is large, the probability that they are not coprime is negligible. Thus, after the modulo operations on LC , only when the chosen S is correct, the elements of the resulting set are of the form zW and have a GCD larger than 1, which is W .

Note that, in some cases, SD_d could still form a set, in which the elements are of the form $zW + rS$ with high probability, even it does not know any user data. For example, if the data matrix is sparse, most of the randomized data is already of the desired form. Another case is that data range is not large enough compared to the amount of data, SD_d could compute $DV = d'_{ij} - d'_{i'j'}$ for all possible pairwise combinations $(d'_{ij}, d'_{i'j'})$ and some of the resulting DVs will be of the desired form. For those cases, SD_d could perform the brute force attack.

Parameter Selection. To resist the brute force attack in the possible cases, $|S|$, i.e. κ_3 , should be at least equal to 80. Moreover, SD_d could compute $LDV = LC_i - LC_j$ for all possible combinations (LC_i, LC_j) . If certain combinations have the same zW inside, those resulting $LDVs$ would be of the form $(r_i - r_j)S$. SD_d could learn S efficiently by computing the GCD of those $LDVs$ even when $|S| \geq 80$. To avoid the case, the randomly chosen z_{ij} should be different with each other with high probability. The z_{ij} is chosen from the range $[1, t]$, and the total number of z_{ij} is $l \cdot N$. According to the generalized birthday problem [22], the probability of at least two chosen z_{ij} match is $1 - \exp\left(-\frac{(lN)^2}{2t}\right)$.

Thus, the probability of no match is $\exp\frac{-(lN)^2}{2t}$ and the parameter κ_1 which determines t could be selected accordingly. Note that if FDs could cooperatively choose the set of z_{ij} such that there is no match, then the range t only needs to be larger than $l \cdot N$.

• **SD_u**: SD_u could get \mathbb{U} and Σ . To recover matrix \mathbb{A} , SD_u needs to find the unitary matrix \mathbb{V} . Let λ denote the rank of \mathbb{A} . The first λ elements of each row in \mathbb{V} correspond to a column of \mathbb{A} , so if SD_u knows the left λ columns of \mathbb{V} , it could recover the original data. Since there are infinite unitary matrices, SD_u could not determine the correct \mathbb{V} if it has no additional information. Thus, we assume that SD_u could get N' original data in some cases. Note that using one data, i.e. one column of \mathbb{A} , could form l equations for the same row of \mathbb{V} . Solving the equations from one data, SD_u could get the first λ elements of that row.

When $N' < N - 1$, since each row of \mathbb{V} is linearly independent with each other, obtaining one row does not help to learn the other rows. Thus, SD_u could not utilize the known data to learn the rest unknown data.

When $N' = N - 1$, SD_u could determine the first λ elements of $(N - 1)$ rows of \mathbb{V} , then the first λ elements of the last unknown row could be determined due to $\sum_{i=1}^N v_{ij}^2 = 1$. The last unknown user data could be recovered accordingly.

• **SD_v**: The purpose of SD_v is to find the unitary matrix \mathbb{U} . Different from the case of SD_u, the first λ elements of each row in \mathbb{U} correspond to a row of \mathbb{A} , i.e. the data value of a certain dimension from all users. If SD_v knows the left λ columns of \mathbb{U} , it could recover the original data. Similarly, we assume that SD_v knows N' linearly independent data in some cases. We have “linearly independent” here because linearly dependent data would not produce new linearly independent equations. Thus, only the number of linearly independent data matters. Each data, i.e. one column of \mathbb{A} , could form one equation for each row of \mathbb{U} .

When $\lambda < l$, if $N' = \lambda$, SD_v could form λ linearly independent equations for each row of \mathbb{U} . Then through solving equations, SD_v could determine the left λ columns of \mathbb{U} and thus recover the whole \mathbb{A} which contains the other unknown user data. On the other hand, if $N' < \lambda$, SD_v could not recover the other unknown linearly independent data due to the lack of enough linearly independent equations. However, for the data which is linearly dependent with the known data, SD_v could recover them because the columns of $\Sigma\mathbb{V}^T$ have the same linear relationships as those existing among user data.

When $\lambda = l$, there is an additional condition for solving the equations of \mathbb{U} , i.e. \mathbb{U} is a unitary matrix. Specifically, the l rows of \mathbb{U} could be regarded as the coordinate axis of l -dimensional space whose rotation degree of freedom is $l - 1$. For each linearly independent data known to SD_v, the rotation degree of freedom of the coordinate axis reduces by 1. Therefore, if $N' = l - 1$, the rotation degree of freedom reduces to 0, i.e. the coordinate axis is fixed. Moreover, since $\sum_{j=1}^l u_{ij}^2 = 1$, each row of \mathbb{U} could be seen as a point locating on the unit sphere of l -dimension. Thus, the set of intersection points between the fixed coordinate axis and the l -dimensional unit sphere is the solution of \mathbb{U} .

Based on the above analysis, the proposed framework could resist the potential attacks launched by SD_d through properly choosing z_{ij} and S . For SD_u,

only when $(N - 1)$ user data is obtained, it could learn the last unknown data. For SD_v , if $\lambda < l$, the framework could resist not more than $(\lambda - 1)$ linearly independent user data leakage and it could resist not more than $(\lambda - 2)$ linearly independent user data leakage if $\lambda = l$.

6 Performance Evaluation

In this section, we evaluate the performance of the proposed fog computing framework in terms of the capacity and efficiency. The capacity demonstrates the number of required ciphertexts for different matrix sizes while the efficiency indicates the computational complexity and communication overhead.

6.1 Capacity

In the proposed framework, the aggregated randomized data is of the form $m_i' = \sum_{k=1}^l a_k \cdot (d_{ki} + z_{ki} \cdot W + r_{ki} \cdot S)$. To guarantee the aggregated data could be decrypted correctly, m_i' should be less than n , i.e. the constraint $\sum_{k=1}^l a_k \cdot (d + tW + tS) < n$ must be fulfilled. At the same time, the superincreasing sequence $\vec{\mathbf{a}}$ has the constraint: $\sum_{j=1}^{i-1} a_j \cdot (d + tW + tS) < a_i$ for $i = 2, \dots, l$. Moreover, in order to derandomize the data, W and S need to fulfill: $W > \max(N, l) \cdot d^2$ and $S > \max(N, l) \cdot (d^2 + 2tWd + t^2W^2)$. To resist the potential attack from SD_d in special cases, t should be chosen based on N and l , and κ_3 should not be less than 80.

Let κ_N, κ_l and κ_d denote the bit length of N, l and d respectively. For simplicity, assume that FDs could cooperatively select the z_{ij} such that no match happens. Then $\kappa_1 = \kappa_N + \kappa_l + 1$ is enough. To meet $W > \max(N, l) \cdot d^2$, we have $\kappa_2 > \max(\kappa_N, \kappa_l) + 2\kappa_d$. Then due to $S > \max(N, l) \cdot (d^2 + 2tWd + t^2W^2)$, $\kappa_3 > \max(\kappa_N, \kappa_l) + 2\kappa_1 + 2\kappa_2 > 3 \cdot \max(\kappa_N, \kappa_l) + 4\kappa_d + 2\kappa_N + 2\kappa_l + 2$. For the sequence $\vec{\mathbf{a}}$, $|a_2| > \kappa_3 + \kappa_1$ and $|a_3| > 2(\kappa_3 + \kappa_1)$. It is easy to find that $|a_i| > (i - 1)(\kappa_3 + \kappa_1)$ and $|\sum_{k=1}^l a_k \cdot (d + tW + tS)| > l(\kappa_3 + \kappa_1)$. Thus, the bit length of aggregated data:

$$|m_i'| = \begin{cases} l[3\max(\kappa_N, \kappa_l) + 4\kappa_d + 3\kappa_N + 3\kappa_l + 3] , & \text{if } \max(\kappa_N, \kappa_l) + 2\kappa_1 + 2\kappa_2 > 80. \\ l(80 + \kappa_N + \kappa_l + 1) & , \text{ else.} \end{cases}$$

It is obvious that the data dimension has a great influence on the aggregated data length. Given different κ_d and l , the number of users which one ciphertext with $|n| = 1024$ could support is evaluated as shown in Fig. 2(a).

From Fig. 2(a), it could be seen that the increase of dimensionality could dramatically decrease the number of users which one ciphertext could support, while the impact of data range d is not that significant. One ciphertext could support large number of users with low dimensional data, e.g. 2^{37} users with 4-dimensional data and 2^{15} users with 8-dimensional data. To support higher dimensional data for the same amount of users, each ED needs to use multiple ciphertexts to aggregate data, e.g. to support 2^{15} users with 16-dimensional data needs 2 ciphertexts each of which aggregates 8 dimensions. Given different l and

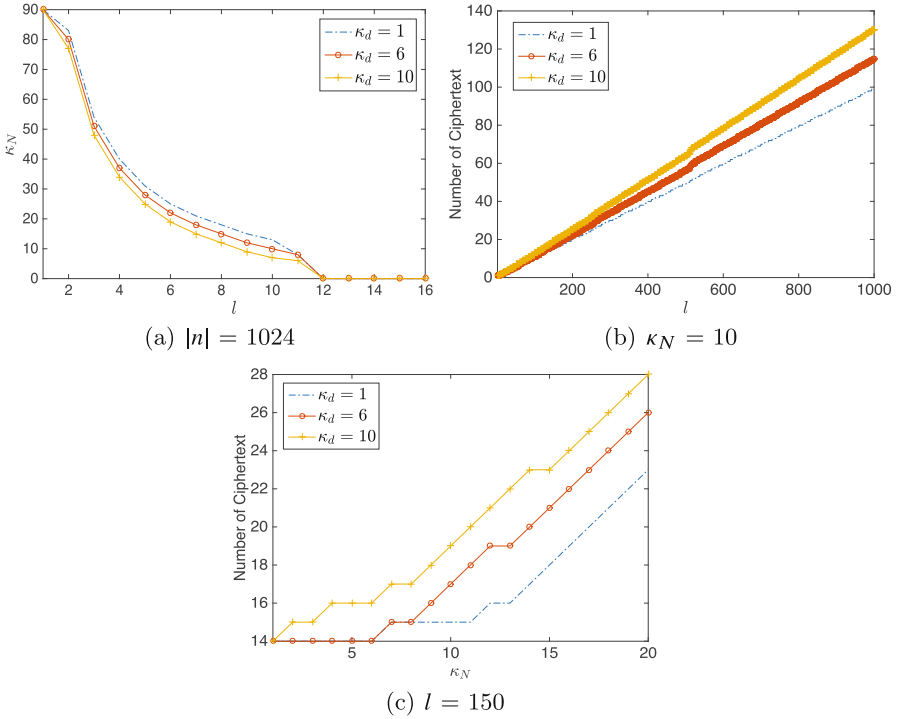


Fig. 2. Capacity of the proposed framework

κ_N , the number of required ciphertexts with $|n| = 1024$ is evaluated in Fig. 2(b) and (c) respectively. It could be seen that each ED needs to use $O(l \cdot \log(N))$ ciphertexts for uploading data.

6.2 Efficiency

As analyzed above, each ED may need more than one Paillier ciphertext for aggregating user data. Let N_C denote the number of required ciphertexts for each ED. In the following, the computational complexity and communication overhead of the proposed framework are analyzed.

Computational Complexity: Because the crypto-operations are much heavier than the computations on plaintext, the amount of crypto-operations is the main concern in this part. Since the fog computing platform in current stage possesses the resource comparable to that of a smart phone, we have implemented the Paillier Cryptosystem on an Android mobile phone. The model number of the phone is Huawei Honor 3C (H30-U10) with the system parameters as: ARM Cortex-A7 4-core CPU @1.3 GHz, 2 GB memory and 4.2.2 Android version. When $|n| = 1024$, the average running time (1000 iterations) for the exponentiation in \mathbb{Z}_{n^2} is 55.493 ms and the time for the multiplication in \mathbb{Z}_{n^2} is 0.201 ms. It is obvious

Table 1. Computational cost of the proposed framework

| Entity | Computational cost (milliseconds) |
|--------|-----------------------------------|
| ED | $2 \times 55.493 \cdot N_C$ |
| FDs | $55.493 \cdot N \cdot N_C$ |
| SD_d | $55.493 \cdot N \cdot N_C$ |

that the cost of multiplication is negligible compared to the cost of exponentiation. According to the procedures of proposed framework, the computational cost for different entities is as shown in Table 1.

Note that SD_u and SD_v only perform computations on plaintext, and server is only in charge of system initialization. Therefore, their computation cost is negligible compared to the other entities. Another notable thing is that the evaluation implicitly assumes the IoT environment devices are as powerful as a smart phone. This is true for the IoT applications which use mobile phones or vehicles to upload environmental information. However, for the applications utilizing low power sensors as EDs, the Paillier operations are still too heavy. To circumvent this issue, the sensor may transmit its data to nearby more powerful device for conducting the crypto-operations. For example, the wristband could connect with the mobile phone for processing and uploading data.

Communication Overhead: In this part, the communication overhead during SVD computation is evaluated. Note that for Paillier Cryptosystem, the ciphertext space is \mathbb{Z}_{n^2} . Thus, the bit length of one ciphertext is $2|n|$. The overhead of each communication flow is as shown in Table 2.

Table 2. Communication overhead of the proposed framework

| Communication flow | Bit length of message |
|---------------------------|---|
| ED \rightarrow FD | $N_C \cdot 2 n $ |
| FDs \rightarrow SD_d | $N \cdot N_C \cdot 2 n $ |
| $SD_d \rightarrow$ SD_u | $l^2(2\kappa_1 + 2\kappa_3 + \kappa_N)$ |
| $SD_d \rightarrow$ SD_v | $N^2(2\kappa_1 + 2\kappa_3 + \kappa_l)$ |

7 Applications

In this section, we discuss the potential IoT applications which could utilize the proposed framework. Basically, the proposed framework could be applied if the application possesses the following characteristics: (1) the application collects the environmental information for data analysis; (2) the data analysis is based on SVD; (3) the number of data analysis tasks is huge; (4) the environmental information is considered as privacy by the application users. Actually, the last two characteristics are the motivation of this work. The large amount of data

analysis tasks motivates us to analyze data on fog computing platform. The privacy concern requires the analysis being privacy-preserving. In the below, we describe a recommendation system as an example to demonstrate the functionality of the proposed framework. More applications which indicate the flexibility of the proposed framework could be found in the detailed version [6].

7.1 Localized Recommendation System

Imagine that there are tens of restaurants in the local region where you live. You have already been some of them and want to try a new one, let's say restaurant p , tonight. Before you go there, you would like to get a reputation score about p from other people in the same region. If the score is too low, you may change the plan. We call the recommendation of local resource as localized recommendation. The advantages of localized food recommendation system over the centralized food review sites could be referred in [6]. Since the personal taste is considered as privacy by many people and the recommendation tasks could appear in different regions frequently and concurrently, to get the local reputation score, we should utilize the proposed framework to conduct the SVD-based collaborative filtering as described in [21]. The detailed procedures are as follows:

- *Step-1.* The user c uploads his rating vector to FD with his mobile phone and informs FD that he is interested in restaurant p . FD collects the rating vectors from other users inside this region. Note that to remove sparsity, each user fills in the ratings of unknown restaurants with his average rating.
- *Step-2.* FD uploads randomized data to SD_d . SD_d , SD_u and SD_v conduct some extra steps for normalizing the data matrix and perform SVD to get \mathbb{U} , \mathbb{V} and Σ . Due to page limit, we omit the normalization steps here. Please refer to the detailed version [6] for the extra normalization steps.
- *Step-3.* SD_u and SD_v reduce \mathbb{U} , \mathbb{V} and Σ to k dimension, and compute $\mathbb{U}_k \Sigma_k^{\frac{1}{2}}$ and $\Sigma_k^{\frac{1}{2}} \mathbb{V}_k^T$ respectively, i.e. SD_u holds $\mathbb{U}_k \Sigma_k^{\frac{1}{2}}$ and SD_v holds $\Sigma_k^{\frac{1}{2}} \mathbb{V}_k^T$. Let $\mathbb{U}_k \Sigma_k^{\frac{1}{2}}(c)$ denote the row of $\mathbb{U}_k \Sigma_k^{\frac{1}{2}}$ which contains the information of user c and $\Sigma_k^{\frac{1}{2}} \mathbb{V}_k^T(p)$ denote the column of $\Sigma_k^{\frac{1}{2}} \mathbb{V}_k^T$ which contains the information of restaurant p . The reputation score of restaurant p for user c is computed as $\mathbb{U}_k \Sigma_k^{\frac{1}{2}}(c) \Sigma_k^{\frac{1}{2}} \mathbb{V}_k^T(p)$. Note that we use z-scores for normalization, so we do not need to add the user average back as in [21].
- *Step-4.* SD_u and SD_v send $\mathbb{U}_k \Sigma_k^{\frac{1}{2}}(c)$ and $\Sigma_k^{\frac{1}{2}} \mathbb{V}_k^T(p)$ to SD_d for reputation score computation. To prevent SD_d from inferring information, SD_u and SD_v randomize $\mathbb{U}_k \Sigma_k^{\frac{1}{2}}(c)$ and $\Sigma_k^{\frac{1}{2}} \mathbb{V}_k^T(p)$ with W and S respectively. For example, let u_i denote the i th entry of $\mathbb{U}_k \Sigma_k^{\frac{1}{2}}(c)$, SD_u randomizes it as $u_i + z_i W + r_i S$.
- *Step-5.* SD_d multiplies the two randomized vectors and sends the result $score_p'$ to FD. Since FD knows W and S , it could recover the reputation score as $score_p = score_p' \bmod S \bmod W$ and send $score_p$ to user c .

From the above description, it has been shown that the proposed framework could utilize the result of SVD operation to compute the reputation score of

unknown restaurants for a specific user. It is straightforward that other similar applications which build the trust of unknown entities based on the third-party recommendations could also adopt the framework. Also note that in the above example, the server does not participate in the process, which means the proposed framework completes all the workload in the edge tier.

8 Related Works

In literature, there are a few works which are related to privacy-preserving SVD computation. Polat et al. [19] proposed a SVD-based collaborative filtering scheme in which the data privacy is protected by randomized perturbation. However, their scheme has been proven unsecure by [24]. Note that the randomization in this work does not have the feature of the randomized perturbation in [19]. Thus, the technique in [24] is infeasible for our work. Canny et al. [5] proposed a collaborative filtering scheme which achieves the SVD computation with privacy-preserving. However, their scheme is specifically designed for the recommendation application. Han et al. [10] proposed a secure protocol for SVD computation. However, their scheme could only support the computation between two parties. Hegeds et al. [11] proposed a private SVD computation for low rank approximation in distributed P2P systems. Compared to our work, the works in [5, 10, 11] have limited applications and require considerable iterations for convergence which brings heavy overhead. Duan et al. [7] proposed a privacy-preserving framework which supports the computation of the learning algorithms which could be expressed as iterative form. Their work could support many learning algorithms while also requiring multiple rounds for the convergence of algorithms, which brings considerable overhead. For example, their scheme needs 83 min to compute the SVD for the Enron Email Data set which is a 150×150 matrix while our work would need $N_C = 15$ ciphertexts to aggregate the 150-dimensional data for each of the 150 users and only takes 499 seconds in total. Note that the evaluation in [7] sums up the computation time for all users even the computation of each user is actually performed concurrently. For fair comparison, our evaluation also accumulates the computation time of all users.

9 Conclusions

In this paper, a flexible fog computing framework for privacy-preserving SVD computation has been proposed. The framework divides the SVD calculation into two eigenvector decomposition operations and distributes the two tasks to different fog devices. The security analysis shows that the user data privacy is preserved during transmission, aggregation and eigenvector decomposition. The possible attacks from the second layer fog devices are also analyzed and the resistance of the framework is discussed. The performance analysis has indicated the capacity of the framework and shows that the data dimension is the most important factor influencing the efficiency of the system. Moreover, one application is

given as an example to demonstrate the functionality of the proposed framework. Compared with the existing works, our framework could support large scope of applications with relatively small resource consumption.

References

1. Cisco delivers vision of fog computing to accelerate value from billions of connected devices (January 2014) (press release). <http://newsroom.cisco.com/release/1334100/Cisco-Delivers-Vision-of-Fog-Computing-to-Accelerate-Value-from-Billionsof-Connected-Devices-utm-medium-rss>
2. Billsus, D., Pazzani, M.J.: Learning collaborative information filters. In: ICML 1998, pp. 46–54 (1998)
3. Boneh, D., Lynn, B., Shacham, H.: Short signatures from the weil pairing. *J. Cryptology* **17**(4), 297–319 (2004)
4. Bonomi, F., Milito, R., Zhu, J., Addepalli, S.: Fog computing and its role in the internet of things. In: Proceedings of the MCC Workshop on Mobile Cloud Computing, 1st edn., pp. 13–16. ACM (2012)
5. Canny, J.: Collaborative filtering with privacy. In: Proceedings of the 2002 IEEE Symposium on Security and Privacy, pp. 45–57. IEEE (2002)
6. Chen, S., Lu, R., Zhang, J.: A flexible privacy-preserving framework for singular value decomposition under internet of things environment. arXiv preprint [arXiv:1703.06659](https://arxiv.org/abs/1703.06659) (2017)
7. Duan, Y., Canny, J., Zhan, J.: P4P: practical large-scale privacy-preserving distributed computation robust against malicious users. In: Proceedings of the 19th USENIX Conference on Security, USENIX Security 2010, p. 14. USENIX Association, Berkeley (2010)
8. Goethals, B., Laur, S., Lipmaa, H., Mielikäinen, T.: On private scalar product computation for privacy-preserving data mining. In: Park, C., Chee, S. (eds.) ICISC 2004. LNCS, vol. 3506, pp. 104–120. Springer, Heidelberg (2005). doi:[10.1007/11496618_9](https://doi.org/10.1007/11496618_9)
9. Golub, G.H., Van Loan, C.F.: Matrix Computations, vol. 3. JHU Press (2012)
10. Han, S., Ng, W.K., Philip, S.Y.: Privacy-preserving singular value decomposition. In: 2009 IEEE 25th International Conference on Data Engineering, pp. 1267–1270. IEEE (2009)
11. Hegedűs, I., Jelasity, M., Kocsis, L., Benczúr, A.A.: Fully distributed robust singular value decomposition. In: 14th IEEE International Conference on Peer-to-Peer Computing (P2P), pp. 1–9. IEEE (2014)
12. Idé, T., Kashima, H.: Eigenspace-based anomaly detection in computer systems. In: Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 440–449. ACM (2004)
13. Kalman, D.: A singularly valuable decomposition: the SVD of a matrix. *Coll. Math. J.* **27**(1), 2–23 (1996)
14. Kamvar, S.D., Schlosser, M.T., Garcia-Molina, H.: The eigentrust algorithm for reputation management in P2P networks. In: Proceedings of the 12th International Conference on World Wide Web, pp. 640–651. ACM (2003)
15. Lee, Y.J., Yeh, Y.R., Wang, Y.C.F.: Anomaly detection via online oversampling principal component analysis. *IEEE Trans. Knowl. Data Eng.* **25**(7), 1460–1470 (2013)

16. Lu, R., Liang, X., Li, X., Lin, X., Shen, X.S.: Eppa: an efficient and privacy-preserving aggregation scheme for secure smart grid communications. *IEEE Trans. Parallel and Distrib. Syst.* **23**(9), 1621–1631 (2012)
17. Nymann, J.: On the probability that k positive integers are relatively prime. *J. Number Theory* **4**(5), 469–473 (1972)
18. Paillier, P.: Public-key cryptosystems based on composite degree residuosity classes. In: Stern, J. (ed.) *EUROCRYPT 1999*. LNCS, vol. 1592, pp. 223–238. Springer, Heidelberg (1999). doi:[10.1007/3-540-48910-X_16](https://doi.org/10.1007/3-540-48910-X_16)
19. Polat, H., Du, W.: SVD-based collaborative filtering with privacy. In: *Proceedings of the 2005 ACM Symposium on Applied Computing*, pp. 791–795. ACM (2005)
20. Sang, Y., Shen, H., Tian, H.: Privacy-preserving tuple matching in distributed databases. *IEEE Trans. Knowl. Data Eng.* **21**(12), 1767–1782 (2009)
21. Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Application of dimensionality reduction in recommender system—a case study. Technical report, DTIC Document (2000)
22. Wagner, D.: A generalized birthday problem. In: Yung, M. (ed.) *CRYPTO 2002*. LNCS, vol. 2442, pp. 288–304. Springer, Heidelberg (2002). doi:[10.1007/3-540-45708-9_19](https://doi.org/10.1007/3-540-45708-9_19)
23. Lindell, Y., Pinkas, B.: Privacy preserving data mining. In: Bellare, M. (ed.) *CRYPTO 2000*. LNCS, vol. 1880, pp. 36–54. Springer, Heidelberg (2000). doi:[10.1007/3-540-44598-6_3](https://doi.org/10.1007/3-540-44598-6_3)
24. Zhang, S., Ford, J., Makedon, F.: Deriving private information from randomly perturbed ratings. In: *Proceedings of the 2006 SIAM International Conference on Data Mining*, pp. 59–69. SIAM (2006)
25. Zhong, S.: Privacy-preserving algorithms for distributed mining of frequent itemsets. *Inf. Sci.* **177**(2), 490–503 (2007)

Novel Sources of Trust and Trust Information

The Game of Trust: Using Behavioral Experiment as a Tool to Assess and Collect Trust-Related Data

Diego de Siqueira Braga¹(✉), Marco Niemann¹, Bernd Hellingrath¹,
and Fernando Buarque de Lima Neto²

¹ Westfälische Wilhelms-Universität Münster, Münster, Germany
{diego.siqueira,marco.niemann,bernd.hellingrath}@uni-muenster.de

² University of Pernambuco, Recife, Pernambuco, Brazil
fbln@ecomp.poli.br

Abstract. Trust is one of the most important dimensions in developing and maintaining business relationships. However, due to the difficult to collect trust-related data from industry, given its concerns surrounding privacy and trade secret protection, it still very problematic to investigate it. Motivated by the growing interest in behavioral research in the field of operations and supply chain management, and by the lack of supply chain trust-related datasets, the authors of this paper proposed and designed a novel trust behavioral experiment. Utilizing concepts of gamification and serious games, the experiment is capable of gathering information regarding individuals' behavior during procurement, information exchange, and ordering decisions considering trust relations in the context of supply chains.

Keywords: Trust · Behavioral experiment · Supply chains · Gamification

1 Introduction

It is the overall aim of this ongoing research project to create a novel behavioral experiment (i.e. The Game of Trust) to assess the influence of trust relationships in B2B supply chains. While this specific approach is unprecedented, serious games already have quite a history in the area of supply chain research and management. Some notable examples include the *Beer Game* [1], the *Mango Game* [2] and the *Trust and Trace Game* [3].

The goal of the experiment is to expose the participants to situations where they do not only have to trust another participant and take risks in order to achieve profit but they can also distrust a certain player and diminish their interactions to that specific participant. The game creates a negotiation environment where players interact with all players of neighboring tiers in the supply chain with the objective of distributing the products along the supply chain in order to achieve profit.

2 The Game of Trust: Initial Concept

Considering all the existing material and research in this area the decision has been made to set up the GAME OF TRUST based on these known and established concepts. The promise of this approach is twofold: First, it avoids redoing work already done by others. Second, making use of known concepts will ensure a flatter learning curve for users.

As the baseline model for the proposed game the *Beer Game* has been selected. Since it has been developed at the MIT in the 1960s, it has been improved [1] and become one of the most known serious games in the SC domain. However, since the *Beer Game* is typically used to visualize the Bullwhip effect (BWE), it lacks mechanisms to enforce or observe the trusting behavior. These concepts were thus extracted from respectively inspired by the lesser known *Trust and Trace Game* [3] and *Mango Game* [2]. In these games e.g. delivering parties can deliver low-quality items as high-quality ones with the receiving parties being enabled to check the actual quality or to ‘trust’.

One component of interest in the initial phase was the supply chain. The *Beer Game* uses a four tier supply chain, where each tier is assigned exactly one player. Considering the intention to include and measure the trusting behavior of participants, such a simple supply chain construction has been identified as a severe limitation. The underlying reason is that the player at each tier is forced to interact with his/her direct neighbors. While this is sufficient for interaction, the degree of risk and uncertainty - which are required properties for trust [4] - can assumed to be low or non-existent. To sanitize this issue and in order to create a market place closer to the real world [5], the decision was made to change the original *Beer Game* supply chain structure. Accordingly, the GAME OF TRUST will allow multiple players at each supply chain tier (see Fig. 1). Furthermore, the *Distributor* tier has been removed. While this tier helps to increase the BWE within the *Beer Game*, the interactions (which are analyzed for trusting behavior) were found to be very similar to the *Distributor* tier so that keeping one of the two tiers simplifies the game without restricting its research potentials.

A sample scenario and design were created to conduct an offline test execution. In the course of the conducted test run the participants had to deal with a very simple and abstract supply chain scenario (*buying and selling products*). It was conducted with members of the development team and several Ph.D. students working at the Department of Information Systems in Münster.

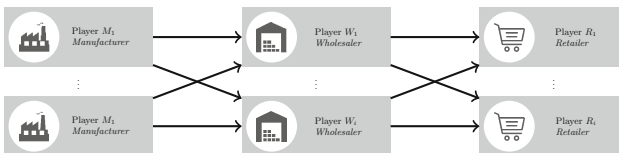


Fig. 1. Game of trust supply chain

Selecting participants working in Information Systems research being familiar with the concept of the supply chain was a purposeful decisions: First, it allowed to conduct the trial without time-consuming introductions into the topic. Second, it enabled the participants to focus on the game mechanics instead of struggling to understand new supply chain concepts. Finally, people with a background in scientific research were supposed to be more capable of providing critical advise regarding methodological or conceptual issues the game version at test might still exhibit.

The subsequent analysis of the trial revealed that the initial GAME OF TRUST had some severe design misconceptions. One of them was the fact that the participants were required to record every action and transaction manually on a set of sheets. Some of the recordings required simple calculations (e.g. *computation of sales volume*), which further intensified the time issues. Aside from the duration issue, the experiment further revealed that the number of interactions was too high. Based on the learnings, the initial analysis step has been reopened to achieve a more desirable solution.

2.1 Game Dynamics

This section focuses on explaining the current game dynamics, player roles, and rules in a thorough manner while avoiding adjustable features such as the specific price for a product at the top of the supply chain.

The game is based on rounds with four phases being executed at each round: Negotiation, Delivery, Financial Closure and Questionnaire. The negotiation phase is based on the Double Auction Mechanism proposed by [6], where a match of the offer and the demand of two negotiation partners is performed in order to allocate the availability of the supplying partner. The matching is performed in three steps, with the upper-tier partner first expressing the expected availability of products. Secondly, the lower-tier partner will make an order based on this availability and the demand of that it has to fulfill. Lastly, the initiator either accepts or rejects the order. A special case where the order matches the initial availability of products causes the order to be accepted automatically. The adaptation of the mechanism in the Trust Game assigns the role of the intermediary deciding the possible allocation of products to the supplying partner.

The delivery phase consists of a two-step sequence. All roles will receive products at the beginning of the phase, with Manufacturers receiving the production of the round and the Suppliers and Retailers receiving the order of the previous round. On the second step, the players will be able to send out the products that have been ordered out of their current updated inventory. For the Retailers, this second step is the delivery of products to the final consumer for demand fulfillment. After the first step is performed, Suppliers and Retailers have the option to execute the previously mentioned Quality Revelation. It will incur a cost for them but will avoid negative consequences when handing a product down the supply chain. If a lie is revealed, a penalty must be paid by the player who delivered the mislabeled product. If a player Alice receives a product and decides to trust the labeling without revealing the quality and sell the product,

and this product is then checked for quality and revealed to be a mislabeling, then Alice is held accountable and must pay the penalty instead of the player providing the product to Alice originally. This setup adds a new layer of risk to the trusting behavior.

The financial closure phase involves the calculation of all costs and incomes for each player. The income of each player is based on the products successfully delivered. Expenses are the sum of all costs. The game considers inventory cost and backordering cost for all roles, quality revelation cost for Suppliers and Retailers, and production costs for Manufacturers.

Finally, a subjective assessment of trust in the form of questionnaires is performed. This evaluation intends to reflect the perception of the players regarding their interaction with other participants with regards to promised quality, successful or unsuccessful negotiations, timely delivery, etc.

2.2 Trust Assessment

To assess trust within the created game a comprehensive literature review had to be conducted, to identify common trust dimensions and measures. The identified dimension and measurement/antecedent structure is visualized in Fig. 2. While literature proposes dozens of different trust antecedents, *Benevolence*, *Competence* (both see e.g. [7,8]) and *Integrity* (see e.g. [9]) were found to be the most dominant ones. As each of these antecedents is rather abstract and as such hard to compute, sets of sub-dimensions were selected to enable a formalization similar to the one conducted by [10].

The overall goal is to profile users when performing a specific game relevant decision. Optimally these set of values should correspond to the subjective perception of another participant. In how far this is actually the case a questionnaire within the game is conducted to examine the subjective perception and look for conformity. To enable their usage they had to be adapted to the data that can actually be gathered throughout the execution of the game.

The negotiation is defined between two participants $from_i$ and to_i . The $offer_i$, which has the same structure as the $order_i$, defines quantity and price for each product type. The $delivery_i$ contains in total the same amount of products as the $order$ but additionally, each of those products has two quality levels: one is referring to the actual quality and another one the quality as described by

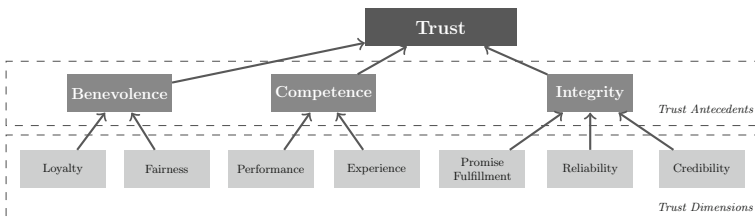


Fig. 2. Trust-dimensions and trust-antecedents

the participant *from*. Receiving the *order_i*, *from_i* can still decide to not accept it due to too high prices demanded by *to_i*. This boolean information is defined as *a_i*. One special feature of the online game is the opportunity to reveal the quality and thereby get the real quality of each product of the delivery. If the participant *to_i* actually revealed the quality of a received *delivery_i*, *q_i* is **true**, otherwise it is **false**. Lastly, a negotiation always has a promised delivery date *d_{pi}* which can either be assumed to have a predefined value of e.g. zero, as it is done in the online game, or it has to be defined in the negotiation phase. Accordingly, an actual delivery date *d_{ai}* describes the date of the delivery. *rnd_i* refers to the round of *n_i*.

This data is utilized in the formulas for each sub-dimension as shown below. Each one is used to assess and measure one negotiation *i*.

Integrity: *Promise Fulfillment* I_P is defined as the likelihood of a trustee keeping a promise to its trustor. Since the definition is based on the discrepancy between promised and actual delivery date in a number of rounds, it only makes sense to calculate this measure from a Supplier to a Manufacturer or from a Retailer to a Supplier. I_P simply takes the difference of the actual to the promised delivery date divided by the latter one. This way late delivery results in a higher actual delivery date and therefore the ratio increases. To assign late deliveries a lower score, the derived ratio is subtracted from the ideal value of one. To retain interpretability in terms mapping I_P to $\{0, 1\}$, it is set to zero if the actual delivery took longer than two times the promised delivery date (as otherwise, it would be smaller than zero).

$$I_P = \begin{cases} \left(1 - \frac{d_{ai} - d_{pi}}{d_{pi}}\right), & d_{ai} \leq 2 \cdot d_{pi} \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

The definition of *Reliability* is straightforward defined as either being one, and therefore accepted, or zero, if the negotiation has not been accepted. One decisive action that directly affects the trustworthiness is the quality revelation. Similar to *Reliability* the revelation can directly be inferred. If the quality is revealed, the *credibility* of that negotiation is zero and one vice versa.

Competence: The *Performance* measure C_P is assessing differences in total quality of delivery. It is used to calculate the performance of each negotiation. The best performance is achieved if the actual quality qty_i^{actual} of *n_i* is at least as high as the promised quality $qty_i^{promised}$. This means that the participant was able to deliver as he promised and is therefore not lying. The other case occurs if the actual quality of a delivery is worse than promised. As an additional weighting factor the price *p_i* is used, so that the *Performance* degrades faster if products are not only sold with a wrong but also for a very high price.

$$C_P = \begin{cases} 1, & \text{if } val_i \geq 0 \\ \frac{1}{|val_i|}, & \text{otherwise} \end{cases} \quad (2)$$

$$val_i = \sum_{qty_j \in qty} (p_{qty_j} \cdot qty_j^{actual}) - (p_{qty_j} \cdot qty_j^{promised})$$

Experience in this context describes the inclination to a specific product type. A participant is considered to be experienced with one type of product if the number of products of that type sold in negotiation n_i is high in comparison to the number of products of all other types in n_i . The equation is one if the number of delivered elements of a quality level qty_j e_{qty_j} ($qty_j \in qty$) is zero for all levels except one.

$$C_E = \frac{\max(e_{qty_j})}{\sum_{qty_k \in qty} e_{qty_k}}, \quad qty_j \in qty \quad (3)$$

Benevolence: *Loyalty* defines whether the two participants in a negotiation n_i were loyal to each other. A loyal participant in the GAME OF TRUST is defined as someone only interacting with one potential client (L_i^-) or source (L_i^+). So for a *Manufacturer* M_i *Loyalty* would mean to trade with only one *Wholesaler* W_i . Similarly the *Wholesaler* would only be fully loyal if he traded with exactly one M_i . The *Loyalty* in the negotiation n_i is defined as the average of the *Loyalties* of the two participants $from_i$ and to_i .

$$N_j^- = \{n_i \in N \mid from_i = from_j \wedge rnd_j = rnd_i \wedge a_i = 1\} \quad (4)$$

$$N_j^+ = \{n_i \in N \mid to_i = to_j \wedge rnd_j = rnd_i \wedge a_i = 1\} \quad (5)$$

$$L_i^- = \begin{cases} 2, & \text{if } |N_j^-| = 1 \\ 1, & \text{if } |N_j^-| > 1 \\ 0, & \text{otherwise} \end{cases} \quad (6)$$

$$L_i^+ = \begin{cases} 2, & \text{if } |N_j^+| = 1 \\ 1, & \text{if } |N_j^+| > 1 \\ 0, & \text{otherwise} \end{cases} \quad (7)$$

$$L_i = \frac{1}{2} * (L_i^- + L_i^+) \quad (8)$$

The *Fairness* of a negotiation n_i is calculated based on the price $p_{qty}(n_i)$ for each product quality qty sold in n_i in comparison to the prices demanded by participants with the same role r .

$$N_r = \{n_i \in N \mid role(from_i) = r\} \quad (9)$$

$$b_{fi} = \frac{1}{|qty|} \cdot \sum_{qty_j \in qty} \left(1 - \frac{p_{qty_j}(n_i)}{\max(p_{qty_j}(N_r))} \right), \quad r = role(from_i) \quad (10)$$

Comparing the prices to those of other negotiations n_i from the same supply chain tier is necessary to obtain meaningful *Fairness* measures. This is grounded on the assumption that the price should rise over the tiers.

3 Future Work: Digital Game

Based on the experiences with the offline based test run, the decision was made to create a digital version of the GAME OF TRUST. The digital version ensures that each player of the game will always be presented with the right forms and that he/she can not forget to enter necessary data. A second major factor for the decision to go digital was the ability to scale. The conducted offline test already revealed the need for a significant amount of moderation work. As the game is intended to help with the collection of profilable data, a lot of moderation overhead was deemed problematic since it would limit the ability to gather a large data set. Providing an electronic online version solves this issues even in two ways: It takes over the moderation part and furthermore enables the game to be played by a larger set of people.

Given the focus on data collection, a difficult trade-off had to be made for the game. On the one hand, the game had to be sufficiently appealing to attract players (and thus data), while on the contrary, it had to be created with minimal effort. Since the GAME OF TRUST is a game experiment hybrid, it was possible to make use of already existing frameworks for online studies. After evaluating the existing alternatives, JATOS [11] was selected as the framework of choice. It already represents a complete service to deliver the experiment/game to the users.

The game design for the virtual GAME OF TRUST will closely follow the design of the offline experiment. For each supply chain role, an interface tailored to the needs of the role will be offered. The data collection is organized in line with the actual implementation of the user interface. It aims at capturing as many as possible details about player interactions in a separate database. Going for more data than might be minimally needed aids to enable future more-sophisticated profiling projects without being forced to create a new dataset.

First experiments with the software prototype were already able to showcase its promise. One potential use of the collected objective and subjective trust data is e.g. the validation of the used trust measures. For example Fig. 3 shows that for some transactions the user-perceived trust nearly maps the computed objective trust (right image), whereas on other occasions the gaps are still large. Given a larger experimental dataset, the GAME OF TRUST will help to identify accurate trust measures which can then subsequently be used to generate valid, trust-based user profiles for supply chain interactions.

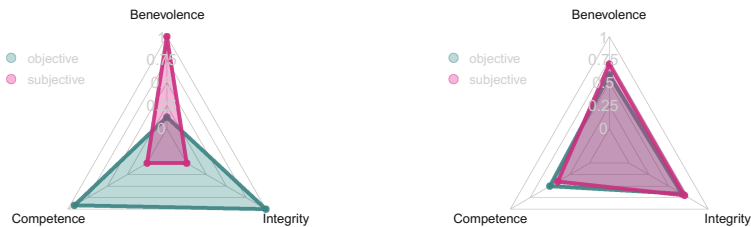


Fig. 3. Mapping between subjective and objective trust measures

References

1. Sterman, J.D.: Instructions for running the production-distribution game “The Beer Game” (1998). <http://opim.wharton.upenn.edu/sok/papers/b/BEER-GAMEINSTRUCTIONSCOMPLETE.PDF>
2. Meijer, S., Zuniga-Arias, G., Sterrenburg, S.: Experiences with the mango chain game. In: Smeds, R., Riis, J., Haho, P., Jaatinen, M. (eds.) Proceedings of the 9th International Workshop of the IFIP, Espoo, Finland, pp. 123–132 (2005). http://library.wur.nl/file/wurpubs/LUWPUBRD_00342546_A502_001.pdf
3. Meijer, S., Hofstede, G.J., Beers, G., Omta, S.W.F.: Trust and tracing game: learning about transactions and embeddedness in a trade network. *Prod. Plann. Control* **17**(6), 569–583 (2006). <http://www.tandfonline.com/doi/abs/10.1080/09537280600866629>
4. Laeequddin, M., Sahay, B., Sahay, V., Waheed, K.A.: Measuring trust in supply chain partners’ relationships. *Meas. Bus. Excellence* **14**(3), 53–69 (2010). <http://www.emeraldinsight.com/doi/abs/10.1108/13683041011074218>
5. Loewenstein, G.: Experimental economics from the vantage-point of behavioural economics. *Econ. J.* **109**(453), 25–34 (1999). <http://onlinelibrary.wiley.com/doi/10.1111/1468-0297.00400/abstract>
6. Wu, S.D.: Supply chain intermediation: a bargaining theoretic framework. In: Simchi-Levi, D., Wu, S.D., Shen, Z.-J. (eds.) *Handbook of Quantitative Supply Chain Analysis*, vol. 74, pp. 67–115. Springer, Heidelberg (2004)
7. Lui, S.S., Ngo, H.-Y.: The role of trust and contractual safeguards on cooperation in non-equity alliances. *J. Manag.* **30**(4), 471–485 (2004). <http://linkinghub.elsevier.com/retrieve/pii/S0149206304000248>
8. Ibrahim, M., Ribbers, P.M.: The impacts of competence-trust and openness-trust on interorganizational systems. *Eur. J. Inf. Syst.* **18**(3), 223–234 (2009). <http://dx.doi.org/10.1057/ejis.2009.17>
9. Haghpanah, Y., DesJardins, M.: A trust model for supply chain management. In: Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence, pp. 1933–1934 (2010)
10. Lin, F., Sung, Y., Lo, Y.: Effects of trust mechanisms on supply-chain performance: a multi-agent simulation study effects of trust mechanisms on supply-chain. *Int. J. Electron. Commer.* **9**(4), 91–112 (2005). <http://www.jstor.org/stable/27751166>
11. Lange, K., Kühn, S., Filevich, E.: Just another tool for online studies (JATOS): an easy solution for setup and management of web servers supporting online studies. *PLOS One* **10**(6), 1–14 (2015). <http://dx.plos.org/10.1371/journal.pone.0130834>

Social Network Analysis for Trust Prediction

Davide Ceolin¹(✉) and Simone Potenza²

¹ Computer Science Department, Vrije Universiteit Amsterdam,
de Boelelaan 1081a, 1081 HV Amsterdam, The Netherlands
d.ceolin@vu.nl

² Konnektid, Herengracht 182, 1016 BR Amsterdam, The Netherlands
simone@konnektid.nl

Abstract. From car rental to knowledge sharing, the connection between online and offline services is increasingly tightening. As a consequence, online trust management becomes crucial for the success of services run in the physical world. In this paper, we outline a framework for identifying social web users more inclined to trust others by looking at their profiles. We use user centrality measures as a proxy of trust, and we evaluate this framework on data from Konnektid, a knowledge sharing social Web platform. We introduce five metrics for measuring trust. Performance achieved an accuracy between 43% and 99%.

1 Introduction

From renting a taxi for cheap to attending free online academic-level courses, the sharing economy has shown an incredible expansion in the recent times. Most - if not, all - of these successful sharing economy services rely on facilities offered through the Social Web. So, besides the “sharing” aspect, these companies are also a clear example of “online-to-offline” (O2O) services. In fact, these services exploit online social interaction to achieve the ultimate goal of offline exchanges (e.g., product or service sharing). In this scenario, online trust management plays a crucial role, because trust is a necessary precondition for users to rely on these services. Often times the actual achievement of the physical engagement depends on the trust that the user places in the online platforms and in the other peers that the user gets in touch with through the platforms themselves.

In this paper, we outline a framework for estimating user trust. This framework uses user features (extracted from her demographics, knowledge and network centrality) to predict trust by means of classification algorithms and, in general, of machine learning. We provide an overview of the framework, and we analyse the effectiveness of the network centrality part, evaluating it on a proprietary dataset provided by Konnektid. Konnektid is a knowledge-sharing online platform that allows users to share knowledge with peers. In Konnektid, users declare which subjects they wish to learn and to teach. Based on their own initiative, or through recommendations from the platform, users connect with their peers and, in case “teacher” and “student” match their needs and

wishes, they meet in person. This is the ultimate goal of the platform. In this scenario, we investigate the use of five network centrality measures (degree centrality, betweenness centrality, eigenvector centrality, closeness centrality and communicability) to estimate trust, which is represented by means of five different metrics (activity count, social activity count, average activity frequency and count of accepted requests and appointments), and computation is performed by means of classification algorithms (Support Vector Machines and Naïve Bayes). This provides us with a first evaluation of the framework. The rest of this paper is structured as follows. Section 2 describes related work. Section 3 describes our approach, which evaluation is presented in Sect. 4. Section 5 concludes.

2 Related Work

We refer to trust as ‘Firm belief in the reliability, truth, or ability of someone or something’ [13]. Sabater and Sierra [14], Artz and Gil [1], and Golbeck [6], provide extensive surveys of trust management models for trust in Computer Science, Social Web, and Web respectively. Sherchan et al. [15] present a survey of trust in social networks. Wu and Chiclana [17] make use of social network analysis to group decision-making problems. Despite the approach similarity, their ultimate goal is reaching consensus in decision-making, while we aim at identifying users that are more prone to trust.

We do not incentivize specific user behaviors, neither create any mechanism for handling and sharing user reputations, but the fact that we estimate trust based on user network centrality implicitly relates to reputation systems. Masum and Tovey [10] and of Golbeck [5] provide extensive analyses of this topic.

Kolaczek [8] proposes a method for estimating trust in social networks, but his focus is on autonomous multi-agent systems while we focus on human-based social networks. Similarly, Di Cagno and Sciubba [2] analyze the impact of trust in social networks, but they focus on lab-created networks instead of real-world data, as we do. Nepal et al. [11] consider an aspect of “social capital” built by users over time when estimating trust. We implicitly aim at fostering the creation of such capital with our work. Grabner-Kräuter and Bitter [7] propose a multi-faceted approach to trust analysis, distinguishing between individual and global aspects of trust. We make a distinction between global (e.g., network centrality) features and individual aspects (e.g., a user’s decision to accept an appointment), and we will deep this separation in the overall framework (see Sect. 3). Lastly, this work can provide the basis for advanced uses of social network information in recommender systems [16] and quality assessment [3].

3 Approach

Our goal is to identify which user features correlate with user trust. First, we must identify useful user features. Second, we need to quantify (or estimate) trust. Lastly, we need to identify reasoning algorithms to link features and trust.

3.1 User Features

We identify three classes of user features useful to this aim:

User demographics. The propensity of users to engage in socializing and in other cooperation and interaction activities might be affected by their demographic profile. For example, younger users might be more inclined to participate, or this inclination could be influenced by cultural factors which could be, in turn, correlated with the nationality of the user.

User knowledge profile. Demographics characterize the user with respect to the population she belongs to. These characteristics are often not decided by the user (e.g., age), and are either immutable or subject to slow changes. A useful user profile can be built also based on the knowledge that the user demonstrates, her tastes, and the knowledge that the user wishes to acquire, thus inducing more dynamics (e.g., user tastes need to be updated periodically). Also, this profile depends on the platform: in some, skills are more important (e.g., knowledge-sharing platforms), in others (e.g., media-sharing platforms), users tastes are more relevant.

User network centrality. Social network users interact with other peers. Their network centrality can be measured in diverse manners: degree centrality, betweenness centrality, etc. These measures provide an indication of with how many users a given user interacts, whether a given user links different parts of the whole social network that would be disjoint otherwise. Intuitively, we suppose that the higher the network centrality of a user is, the higher is her tendency to trust and interact. However, which centrality measures better indicate trust and how strong such a correlation is, needs to be investigated.

3.2 Trust Measures

Trust is a belief that somebody shows with respect to something or somebody in a given context [12]. Since we situate in the realm of Social Web apps, we identify two main subjects of trust, namely the app itself and other users (which the app allows getting in touch with).

Trust in the App. Trust in the app and in the service provider are a necessary precondition for the user to join a Social Web app. This implies trust in how user personal information is dealt with, and trust in the app behavior and its functionality. This prerequisite is the basis for building user engagement. Users do hardly engage with Social Web platforms they do not trust, especially when these are aimed at creating contacts in the real world (as in the case of O2O). We use engagement indicators like the number of user accesses as trust proxies.

Trust in Other Users. Trust in other users is the key aspect of Social Web apps. While trust in the app is a precondition for the user to utilize it, trust in other users is the requirement for users to join the app. Social Web apps

are meant to enhance and facilitate user interaction, thus relying on trust to be established. Measuring trust in users is important, for example, to identify users whose engagement needs to be fostered by means of recommendations or other actions. Depending on the platform, user trust can be estimated based on the number or frequency of interactions that a user has with others. In O2O apps, user trust can be measured by user acceptance of real-world transactions.

3.3 Reasoning Algorithms

Having identified the possibly relevant user features, and having identified proxies for trust, then we will use machine learning algorithms for identifying correlations between them. We prefer classification algorithms since we treat trust metrics as qualifying classes. So, we employ the Support Vector Machines and Naïve Bayes algorithms. Alternative approaches (e.g., to improve computational performance) will be evaluated when we will extend our framework.

4 Evaluation

4.1 Dataset Description

We perform a preliminary evaluation of our approach on a dataset of user interactions by Konnektid [9] consisting of the logs of 37,423 user actions performed between September 2012 and August 2015. The only personal information present in this dataset is anonymous user identifiers. Actions are classified as:

ProfileRegistered, ProfileUpdate. To access the platform, users register their profile (which contains both demographics and indications about what they wish to learn and to teach). Profiles can be updated by users anytime.

DirectRequest, NeighbourRequest, GroupRequest. Users can issue requests to learn particular skills. These requests can be directed to selected users, or broadcasted, also to the neighboring users (geolocated).

DirectMessageSent. Users can exchange textual messages.

AppointmentCreated, AppointmentUpdated, AppointmentAccepted.

The goal of the app is to facilitate user encounter, in person, to let them teach something each other.

Graph Description. We model the social graph of Konnektid as follows. Each node of the graph is represented by a user. Each edge represents any possible kind of interaction occurred among users. In this manner, we model user interaction, without focusing on its quality or frequency, but merely from the “social” point of view. We will consider different kinds of graphs in the future.

4.2 Network Centrality Features

On the graph described above, we calculate the following five network centrality measures to be used as features for trust prediction in this setting.

Degree Centrality. The degree centrality of a node is equal to the degree of that node, i.e., to the number of edges that connect that node.

Closeness Centrality. The closeness centrality of a node is the reciprocal of the sum of the distances between that node and all the other nodes.

Betweenness Centrality. The betweenness centrality of a node counts how many times it acts as part of the shortest path between two nodes.

Communicability Centrality. This is the sum of closed walks of all lengths starting and ending at a given node. This is defined as: $CC(i) = \sum_{j=1}^N C_{i,j} = [e^A]_{i,j}$, where i, j are nodes and A is the adjacency matrix [4].

Eigenvector centrality. This computes the centrality for a node based on the centrality of its neighbors. The eigenvector centrality for node i is $\mathbf{Ax} = \lambda \mathbf{x}$ where A is the adjacency matrix of the graph G with eigenvalue λ .

4.3 Trust in the Platform

Trust in the platform is estimated based on the user activity. Trust is necessarily tangled with other user attitudes, like user engagement, and user preferences. Even if it is not possible to discern the influence of trust on user activities, trust is necessarily their prerequisite: users interact with the platform because, consciously or not, they trust it. Trust is present in any other interaction of the users with any other platform. However, in this case, the platform is a means to interact with strangers that users will decide whether to encounter or not. Hence, trust in the platform implies trust in its ability to preserve privacy and in its ability to identify potentially interesting encounters. We propose the following measures as proxies for this type of trust:

Count of Activities. The first measure of user interaction is given by overall the count of user activities. This measure corresponds to the degree centrality computed on a graph representing all the interactions performed among users, while our graph of interest is unweighted and undirected, and represents any kind of interaction among users, regardless of their frequency or type.

Results. We run the Support Vector Machine (SVM) algorithm with Stochastic Gradient Descent (SGD) preprocessing to predict the number of activities of each user, treating this problem as a classification problem (so to predict the “1-activity users”, the “2-activities users”, etc.). We evaluated SGD-SVM with 10-fold cross validation, obtaining 43% accuracy (there are 67 different classes in total, i.e., users have 67 different numbers of actions performed). Accuracy is computed as the percentage of correctly classified items. Accuracy rises to 84% when we group actions in groups of 5 (i.e., users who performed between 0 and 4 actions fall into the same class, etc.)

Count of “Social Activities”. Users can perform different activities on the Social Web app. Besides the fact that all these activities are meant to facilitate social interaction, only some of them actually involve other users. For example,

a user might decide to update her own profile in order to be more easily contacted, but this action does not directly involve other users. This measure is equivalent to the degree centrality computed on the network reporting only the following activities: message sending, offer sending, requests sending, appointment making and updating.

Results. We employed SGD-SVM also in this case, and we evaluated it by running 10-fold cross-validation also in this case. We obtain an accuracy of 67%, which reaches 92% by grouping the counts of social actions in classes modulo 5.

Weighed Activity Frequency. The count of activities is a possible indicator of user interaction with the platform. However, this indicator does not take into account the time span of this interaction: a user might perform a high number of activities in a limited period of time, and then disappear. Or, she could demonstrate trust and engagement in the platform by participating frequently. So, as another measure of trust, we propose a weighed measure of user frequency. On the one hand, in fact, we value frequent user activities. On the other hand, we ‘penalize’ users who do not return to the platform. We define the measure in such a manner that it ranges from 0 (no trust) to ∞ (full trust). Also, we define this measure so to take a specific point of view that corresponds to a specific time instant t : to decide whether a user u ‘disappeared’ for a long period of time, we must be sure that a long period of time occurred between our observational point and her last appearance. The resulting metric is defined as:

$$weighed_freq(u, t) = e^{-\frac{t(u)_{last} - t(u)_{first}}{\#activities(u)}} * e^{-(t - t(u)_{last})}$$

Results. SVM with 10-fold cross validation reaches 94% accuracy.

4.4 Trust in Other Users

Here we define metrics for estimating the trust users express in other users. These metrics are computed from the logs of user activities in the platform.

Count of Accepted Requests. Users receive requests from other users. We count how many times each user reacts to a request with an offer. This count is affected by the user “good-will”, by the fact that she is interested in the content of the offers received, as well as by the intention of the user to trust the requester: ultimately these offers should lead to meetings in person.

Results. SGD-SVM with 10-fold cross-classification achieves 99% accuracy in this case. All the users receive requests because, besides those issued by other users, the system itself periodically sends requests, in an attempt to facilitate encounters. However, the longevity of users is likely to be linked to the number of requests received, and thus it could make sense to analyze also the ratio between the number of requests received and the number of offers made.

Count of Accepted Appointments. The second measure of trust in other users that we propose to adopt is the number of appointments a given user accepted.

Results. SGD-SVM with 10-fold cross-validation achieves 99% of accuracy in this case, but this is due also to the sparsity of appointments accepted with respect to the total counts of activities (indeed, these correspond to about 1% of the activities). More interesting, in this case, is the recall. Given the sparsity of the data targeted, we can sacrifice part of the precision of the results in order to identify a large enough set of candidate users that comprises most of the users who actually accepted an appointment. Recall of SGD-SVM is, in fact, 14%. In this case, we run also Naïve Bayes as an alternative classification algorithm. This allows us still achieving 99% but with 55% recall.

5 Discussion

This paper introduces a framework for predicting trust in Social Web apps. In particular, in this framework, we analyze the use of network centrality measures to predict trust that users show in the platform and in other users. Our analyses show that interpersonal trust is well-captured by user centrality: the more central a user is, the more prone to trust others he will be. This is useful, for instance, to identify users to recommend to newcomers, in order to increase the likelihood of positive outcomes of interactions. Also, there is a clear link between trust in the platform (and, hence, engagement), and user centrality. This link is weaker than interpersonal trust, but still identifies in the number of diverse network links one possible motivation for user engagement. In the platform that we analyze, user engagement and interpersonal trust are tightly bound because of the nature of the task performed: users interact with the platform in order to interact with other users. These results could hence be expected. However, they show that, besides the fact that users are motivated to use the platform because of already-established acquaintances, the creation of new links and their diversification are important factors to consider to foster quality interaction. In fact, diverse centrality measures focus on different aspects of connectivity, from the mere number of connection (like in the case of degree centrality) to the ability to connect diverse groups of users (like in the case of betweenness centrality).

In the future, we will develop further this framework in all its three main components: features, trust measures, and reasoning algorithms. We will expand the set of centrality measures considered and add in the computation also demographics, knowledge features (as defined in Sect. 3), and possibly other classes of features, as the current selection is heavily driven by the case study at our disposal. Also, we aim at investigating further the trust metrics proposed, in order to extend them, as well as to identify relations between them (e.g., one trust metric might be highly correlated with others; this kind of information is useful to increase computation performance). Lastly, we will consider other prediction algorithms. For example, besides the classification angle taken in this paper,

given that user actions occur sequentially, it might be useful to model them in terms of Markov chains, to predict whether the sequence of actions (rather than the set of action) performed by a user provides indications for trust.

Acknowledgements. This work has been partially funded by the Dutch national program COMMIT under the Big Data Veracity project and by a research voucher awarded by the Network Institute of the Vrije Universiteit Amsterdam.

References

1. Artz, D., Gil, Y.: A survey of trust in computer science and the semantic web. *J. Semant. Web* **5**(2), 58–71 (2007)
2. Cagno, D.D., Sciubba, E.: Trust, trustworthiness and social networks: playing a trust game when networks are formed in the lab. *J. Econ. Behav. Organ.* **75**(2), 156–167 (2010)
3. Ceolin, D., Noordegraaf, J., Aroyo, L., van Son, C.: Towards web documents quality assessment for digital humanities scholars. In: *ACM WebSci 2016* (2016)
4. Estrada, E., Hatano, N.: Communicability in complex networks. *Phys. Rev. E* **77**, 036111 (2008)
5. Golbeck, J.A.: Computing and applying trust in web-based social networks. Ph.D. thesis, AAI3178583 (2005)
6. Golbeck, J.A.: Trust on the World Wide Web: a survey. *Found. Trends Web Sci.* **1**(2), 131–197 (2006)
7. Grabner-Kräuter, S., Bitter, S.: Trust in online social networks: a multifaceted perspective. *Forum Soc. Econ.* **44**(1), 48–68 (2015)
8. Kołaczek, G.: Agent and multi-agent technology for internet and enterprise systems. In: Håkansson, A., Hartung, R., Nguyen, N.T. (eds.) *Social Network Analysis Based Approach to Trust Modeling for Autonomous Multi-agent Systems*. *SCI*, vol. 289, pp. 137–156. Springer, Heidelberg (2010)
9. Konnektid. <http://www.konnektid.com>
10. Masum, H., Tovey, M. (eds.): *The Reputation Society*. MIT Press, Boston (2012)
11. Nepal, S., Sherchan, W., Paris, C.: Strust: a trust model for social networks. In: *TrustCom*, pp. 841–846 (2011)
12. O’Hara, K.: A general definition of trust. Technical report, University of Southampton (2012)
13. Oxford English Dictionary. Trust. <https://en.oxforddictionaries.com/definition/trust>. Accessed 27 Mar 2017
14. Sabater, J., Sierra, C.: Review on computational trust and reputation models. *Artif. Intell. Rev.* **24**, 33–60 (2005)
15. Sherchan, W., Nepal, S., Paris, C.: A survey of trust in social networks. *ACM Comput. Surv.* **45**(4), 1–33 (2013)
16. Wu, J., Chen, L., Yu, Q., Han, P., Wu, Z.: Trust-aware media recommendation in heterogeneous social networks. *World Wide Web* **18**(1), 139–157 (2015)
17. Wu, J., Chiclana, F.: A social network analysis trust-consensus based approach to group decision-making problems with interval-valued fuzzy reciprocal preference relations. *Knowl. Based Syst.* **59**, 97–107 (2014)

Investigating Security Capabilities in Service Level Agreements as Trust-Enhancing Instruments

Yudhistira Nugraha^{1,2}(✉) and Andrew Martin¹

¹ Centre for Doctoral Training in Cyber Security,
Department of Computer Science, University of Oxford, Oxford, UK
{yudhistira.nugraha, andrew.martin}@cs.ox.ac.uk

² Directorate of Information Security, Ministry of ICT, Jakarta, Indonesia
yudhistira.nugraha@kominfo.go.id

Abstract. Many government agencies (**GAs**) increasingly rely on external computing, communications and storage services supplied by service providers (**SPs**) to process, store or transmit sensitive data to increase scalability and decrease the costs of maintaining services. The relationships with external **SPs** are usually established through service level agreements (**SLAs**) as trust-enhancing instruments. However, there is a concern that existing **SLAs** are mainly focused on the system availability and performance aspects, but overlook security in **SLAs**. In this paper, we investigated ‘real world’ **SLAs** in terms of security guarantees between **GAs** and external **SPs**, using Indonesia as a case study. This paper develops a grounded adaptive Delphi method to clarify the current and potential attributes of security-related **SLAs** that are common among external service offerings. To this end, we conducted a longitudinal study of the Indonesian government auctions of 59 e-procurement services from 2010–2016 to find ‘auction winners’. Further, we contacted five selected major **SPs** ($n = 15$ participants) to participate in a three-round Delphi study. Using a grounded theory analysis, we examined the Delphi study data to categorise and generalise the extracted statements in the process of developing propositions. We observed that most of the **GAs** placed significant importance on service availability, but security capabilities of the **SPs** were not explicitly expressed in **SLAs**. Additionally, the **GAs** often use the provision of service availability to demand additional security capabilities supplied by the **SPs**. We also observed that most of the **SPs** found difficulties in addressing data confidentiality and integrity in **SLAs**. Overall, our findings call for a proposition-driven analysis of the Delphi study data to establish the foundation for incorporating security capabilities into security-related **SLAs**.

Keywords: Security · **SLAs** · Trust · Security capability · Grounded Delphi method

1 Introduction

In recent years, many governments have been targets for a wide range of cyber attacks, by perpetrators ranging from unskilled individuals to foreign intelligence services. According to data from BAE Systems, 85% of the attacks have targeted high-profile organisations, such as government ministries (55%), embassies (15%) and public organisations (12%).¹ This statistical data is also supported by the Control Risks on Risk Map Report 2016, which pointed out that governments are the top sector targeted by cyber attacks (36% of total attacks). This is not surprising, as many governments generate, collect and store far more sensitive data than the private sectors, and this data is accumulated in more vulnerable systems. Consequently, some governments, notably the UK, the US and China require SPs to demonstrate compliance with government security requirements [14–16].

In fact, many government agencies (GAs) increasingly rely on external computing, communications and storage services supplied by service providers (SPs). The relationships with external SPs are usually established through service level agreements (SLAs) as trust-enhancing instruments. The concept of trust can be defined as a belief that a security capability will behave in an expected manner when demonstrating compliance with a security requirement according to particular threat. Whereas, a security capability is a combination of mutually-reinforcing security controls that are implemented by technical, physical and human elements [18]. In some cases, the level of *trust* is determined in relation to a specific *security capability* provided by external SPs [18]. For instance, an acceptable level of protection will be required depends on the trust that GAs place in external SPs [18] when using such external services. However, there is an absence of coherent approaches for preserving the confidentiality of sensitive data across GAs when using such SLAs. On top of that, most external SPs place a greater emphasis on the system availability and performance aspects, but overlook security in SLAs [3, 4, 7]. Also, they do not adequately incorporate security capabilities of the SPs into formulating security-related SLAs.

This study investigates the current and potential attributes of security-related SLAs that are common among external computing, communication and storage service offerings, using Indonesia as a case study. To this end, we conducted a longitudinal study of the government auctions of 59 e-procurement services to select major external SPs that provided Internet services, cloud-based services and data centre services across 80 GAs between 2010 and 2016. The selected SPs were then contacted to participate in a three-round Delphi study with group discussions and individual sessions to clarify security capabilities in SLAs. We analysed the Delphi study data using a grounded theory analysis [22–24], and synthesised findings, as follows: (i) perceived threats, (ii) government-specific security requirements, and (iii) service provider-specific security capabilities. We then postulate propositions for each research question.

¹ Data was gathered from the slide, <https://goo.gl/vumsm2>, (Accessed March 2017).

In this paper, we claim three contributions. Firstly, we report a longitudinal study of the government auctions in Indonesia from 2010–2016. The insight will be useful to the government and other governments who make decisions. Secondly, we discuss how these findings can be used to improve such an understanding to incorporate the interplay of threats, security requirements and security capabilities into security-related SLAs. The insight will be used to develop a framework in the formulation of security-related SLAs as trust-enhancing instruments. Finally, we propose a grounded adaptive Delphi method to clarify existing security-related SLAs in service provision.

The remainder of this paper is structured as follows: Sect. 2 presents the research methodology. Section 3 reports key findings and discusses propositions. In Sect. 4, we discuss the implications of our findings, followed by the limitations of the paper and reflection with related work. We conclude our study in Sect. 5.

2 Research Methodology

This paper attempts to investigate the current and potential attributes of security-related SLAs that are common among external computing, communication and storage service offerings. Particularly, we attempt to clarify existing ‘real world’ SLAs with external SPs in terms of security guarantees to GAs, using Indonesia as a case study. As SLAs can be established with various interacting entities (i.e. customers, end-users, SPs, suppliers, integrators, standards bodies and accreditation bodies), this study was limited to GAs as customers who increasingly rely on such external services provided by SPs.

We use Indonesia as a case study because according to Article 12 of Indonesian Government Regulation on the Operation of Electronic Systems and Transactions Number 82 of 2012, SPs have obligations to ensure agreements on minimum service level and information security when providing such external services to customers (e.g. GAs). Furthermore, *e-Government procurement systems* officially have been widely used since 2015 for procuring external information system products and services. For the purpose of this study, we aim to select representative SPs that supply external communications, computing and storage services to GAs through 59 e-procurement services in Indonesia.

Due to the inherent limitations of empirical studies of the scope of the current research, we developed a grounded adaptive Delphi method (GADM) that combines elements of the Delphi method and grounded theory (GT) (Fig. 1). Both the Delphi method and GT consist of simultaneous data collection and analysis, with each process being interrelated and iterative. The GADM varies in some respects from the two previous grounded Delphi methods [27, 28]. An important similarity between these methods is the integration of GT analysis and a group communication processes. One of the differences is that the GADM is based on a Policy Delphi approach [29] and an adaptive Wideband Delphi method [19], which aim to suit the different views of individual participants on specific matters, with greater generalisability across different participants. The GT analysis is well suited for capturing these different views from the participants.

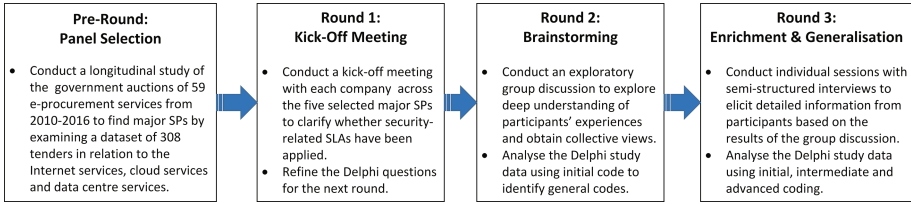


Fig. 1. The research method—a grounded adaptive Delphi method (GADM)

To this end, we conducted a longitudinal study of the government auctions in Indonesia to find “auction winners” or major external SPs, which were then contacted to do extensive face-to-face meetings as the application of GADM. In this paper, we adopted an adaptive wideband Delphi study [19] to enable the surveying of multiple panellists from major SPs through group discussions to clarify existing security-SLAs, along with individual sessions through semi-structured interviews to gather genuine knowledge and experiences in relation to the current and potential attributes of security-related SLAs. We analysed the Delphi study data using a grounded theory analysis to categorise and generalise the extracted statements.

2.1 Research Participants

Since the motivations and experiences of the participants directly affect the quality of the findings, the selection of participants is considered as an important aspect of a Delphi study. Consequently, a comprehensive selection criteria is necessary to select appropriate participants. In this study, particular attention was paid to the selection of SPs that provided external computing, communications and storage services to GAs through the government procurement system in Indonesia. To this end, we conducted a longitudinal study of the government auctions to find “auction winners” or major external SPs, which were then contacted to ask their participation in the data collection activities. We carried out the search process in the following steps.

Step 1: We created and examined a dataset of 308 government tenders in relation to the Internet services, cloud-based services and data centre services from 59 e-procurement systems (SPSE) across 80 government agencies of which some agencies engage with other procurement services from other agencies.

Step 2: We accessed the SPSE website for each government agency. Most of the SPSE website follow the general format: `lpse.[agency’s website]/eproc/lelang`. We analysed 95944 government auctions from 2010 to 2016.

Step 3: We used the automated search and applied the following *five* keywords, which were adopted from the Gartner Global IT Spending Forecast, to the site’s search engine: (1) **Data Centre**, (2) **Cloud**, (3) **Co-location**, (4) **Internet**, and (5) **Network**. We initially extracted 273 for data centre category, 31 for

cloud category, **17** for co-location category, **230** for Internet category and **236** for network category.²

Step 4: We selected the set of e-procurement services, which could be relevant by reading the title of tender as well as identifying the relevant keywords in relation to the *five* keywords. Further, we searched by looking at information about the auctions that aimed to retrieve the requirements specifically for selecting external SPs.

Step 5: Finally, we identified major external computing, communications and storage services that are widely procured across GAs. To understand the government's supply chain, we identified the SPs who were selected as auction winners.³

Further, we invited the five major selected SPs based on our longitudinal study to participate in a three-round Delphi study. We recruited our participants via an email containing an official invitation letter on behalf of the Indonesian ministry of communications and information technology. We typically corresponded with an organisational leader who then suggested potential participants according to the following the selection criteria: (1) work experience and background, (2) involvement in the government procurement auctions, and (3) a visible interest in the research topic. We also distributed the Delphi questions⁴ to all potential participants across the five selected SPs before they agreed to take part in this study. Finally, we received 15 participants confirmed [P1–P15] who were representatives from the five selected SPs.

Although there is no need to meet certain number of participants [30], divergent opinions are required with more than two participants. Okoli and Pawlowski [31] suggest 10–18 participants on a Delphi panel. Other researchers suggest that the recommended size of the panel of experts varies from 5–20 participants [32], 10–15 participants [33] and 15–20 participants [34]. In this study, we aimed for a panel size of 6–11 participants for each round. The number of participants was sufficient for providing theoretical saturation. Although saturation occurred within the first twelve interviews, basic meta-themes became apparent after only six interviews [26].

Our participants are technical and regulatory compliance experts that have been working for many years at the five SPs {SP1, SP2, SP3, SP4, SP5}, which were selected as the winners of auctions, and provided Internet services, cloud-based services and data centre services to the GAs between 2010 and 2016. We spoke with our participants across the spectrum of general technical, procurement and security expertise.⁵

2.2 Data Collection: A Three-Round Delphi Study

We collected data primarily through a three-round Delphi study with 15 experts across the five selected SPs. We use some features of Delphi, such as group

² e-Gov Procurement on IT Services, <https://goo.gl/hzcHL9>, (Accessed March 2017).

³ Government Procurement Auctions, <https://goo.gl/5LhWun>, (Accessed March 2017).

⁴ Delphi study questions, <https://goo.gl/mIrQUk>, (Accessed March 2017).

⁵ Participants information, <https://goo.gl/dBSDcn>, (Accessed March 2017).

responses with face to face meetings for eliciting collective views and individual sessions with semi-structured interviews for collecting individual views where participants may not wish to elaborate in a group discussion [21]. Unlike other Delphi studies [27, 28], this study used group discussions and interviews instead of questionnaires as the instrument for data collection because the questionnaires are impractical for the purpose of eliciting genuine views or thoughts from busy participants, such as vice president and director.

Round 1: Kick-Off Meeting. We conducted a kickoff meeting with each company across the five selected SPs. However, one company did not take part in the first round due to some technical reasons. This round was intended to clarify the service providers' understanding of their obligations to ensure agreements on service level and information security. This stage was also important to refine the Delphi questions for the next round.

Round 2: Brainstorming Phase. We conducted an exploratory group discussion with representatives of participants from five selected SPs to explore a rich understanding of participants' experiences and beliefs, as well as to generate information on collective views [20]. We invited the 15 participants who initially agreed to participate in the study. However, only nine participants ($n = 9$) from the five SPs attended the focus group.

Round 3: Enrichment and Generalisation Phase. We conducted individual sessions using semi-structured interviews to elicit detailed information from participants based on the results of the group discussion. We invited the 15 participants again to participate in the third round. However, we only conducted interviews and individual feedback with six participants ($n = 6$) from two selected SPs. The two providers are the major SPs in Indonesia, and their network infrastructures were reported to be compromised according to Edward Snowden's revelations in 2013 [19].

2.3 Data Analysis: Grounded Theory Analysis

We applied the grounded theory analysis [22–25] to examine group discussion and interview transcripts, and to categorise and generalise the extracted statements. The process of developing a proposition was established after a thorough examination of the Delphi study data by establishing conceptual relations between categories.

In this study, the main researcher performed analysis of the Delphi study data. We conducted initial coding of a group discussion transcript to identify general codes. Further, we analysed the interview transcripts including the focus group discussion transcript, using initial coding, intermediate coding and advanced coding [25].

We used initial coding to identify topic of interest 'key-point coding' in which the researcher extracted useful sentences or statements and applied codes against the Delphi study data. In intermediate coding, we began to select categories from amongst topics of interest and found relationships among the initial codes

(e.g. the most frequent or important codes) [24]. In advance coding, once categories were identified, we established the relationship between the categories to integrate them into a cohesive proposition regarding the interplay of threats, security requirement and security capabilities expressed in the formulation of security-related SLAs.

We can illustrate the grounded theory analysis with an example from this study. One participant commented that the greater threat to external SPs mostly come from DDoS attacks. We coded the following statement as ‘deny access’.

“With regard to cases that hit banks around the world, such as SWIFT attacks, we, the service providers are required to protect against DDoS attacks” (P1).

Unlike other qualitative studies where coding is performed by multiple researchers, the Delphi study data was coded only by the single researcher due to confidentiality reasons. However, the researcher discussed his findings with another researcher to receive feedback and comments on the findings.

3 Results and Analysis

In designing and analysing our research data, we will present our detailed findings for each primary research question, as follows:

1. What are the perceived threats to computing, communications and storage services as seen from the perspective of a service provider?
2. What are the government-specific security requirements when using external computing, communications and storage services supplied by service providers?
3. What are the security capabilities of the service providers used to mitigate the threats, and to demonstrate compliance with the security requirements?

We format the statements and challenges raised by participants in italics to distinguish them from our interpretations. We conclude each primary research question with propositions we derived from findings. By applying an appropriate qualitative analysis [24], we identify important codes and other observations present in the Delphi study data. We then report the raw number of participants who discussed a certain code to give an approximate indication of its prevalence amongst selected SPs.

3.1 Perceived Threats

We begin by examining specific threats that SPs are attempting to counter. Several statements have been made by participants to mitigate perceived threats to their service offerings. We noticed that consensus was obtained regarding a specific threat. For instance, many participants mentioned specific threats in relation to **Deny Access**. We highlight the perceived threats, as follows⁶:

⁶ Perceived threats, <https://goo.gl/IdNKZj>, (Accessed March 2017).

Deny Access. Many participants discussed this type of threat as the main security concern. This threat allows an adversary to prevent legitimate users from accessing the services. Thus, our participants paid much attention to mitigating the following threat:

“Our concern as a service provider is related to DDoS attacks because we can have three times the DDoS attacks in one month” (P11).

Misuse. Our participants were typically concerned with the weakest link (e.g. people). This threat allows an adversary to perform unauthorised use of assets. Some participants pointed out that authorised users could perform malicious actions to obtain sensitive data from the target. One of these participants indicated the following statement:

“We consider the highest risk is that authorised users that perform abuse or malicious stuff” (P6).

Transmit. Our participants discussed the importance of preventing unauthorised transfer of data, as this threat allows an adversary to transmit sensitive data externally. Only one participant indicated the threat (i.e. data exfiltration) in the following statement:

“An effort is needed so that data cannot be read and transferred by other people while data is in storage” (P1).

Intercept. A few participants reported that an adversary could intercept communication from the target people or devices, as indicated in the following:

“If the Internet is used by customers to send sensitive information without using a secure protocol, an attacker can intercept the communication” (P1, P3).

Based on the aforementioned perceived threats, the extracted statements demonstrate challenges for offering an opportunity to specify security capabilities in SLAs. The most striking result to emerge from the Delphi study data is that the GAs often consider service availability the highest priority because DDoS attacks are currently targeting government services. We then postulate two propositions, as follows:

Proposition 1. *Identifying [perceived threats] is correlated with the concept of formulating [security requirements].*

A strong relationship between threat models and security requirements has been reported in the literature [35]. As we learned from this study, our participants confirmed that such an understanding of the present and future perceived threats would help GAs and external SPs to formulate security requirements. In other

words, external SPs can concern about specific perceived threats and/or vulnerabilities to express security requirements, and to specify security capabilities used in the formulation of security-related SLAs, which can provide trustworthy services to GAs [17].

Proposition 2. *The current information about [perceived threats] is correlated with the concept of applying [security capabilities] to mitigate threats.*

Mitigating perceived threats plays an important role to deliver more secure products, services, or technologies. Our participants revealed that the GAs did not specify specific security capabilities for mitigating particular threats when using such external services. In most cases, the GAs are often less careful in terms of security objectives other than service availability. Our participants pointed out that although specific security objectives were not demanded by the GAs, the SPs employed minimum security capabilities, without additional cost of security services, to help ensure the services remain available based on the SLAs. Therefore, it can be assumed that the SPs will make their best effort to ensure their security posture when they provide such services to the GAs whether the agencies consider the need for security capabilities to mitigate possible threats, or not.

3.2 Government-Specific Security Requirements

Understanding the perceived threats can drive security requirements. Thus, security requirements play an important role in mitigating threats, such as unauthorised disclosure data by foreign intelligence services [19, 35]. However, our participants confirmed that understanding the government security requirements was essential in offering trustworthy services to the GAs. However, several challenges were described by participants, such as there were no specific security requirements from the GAs of what security capabilities the SPs would implement when processing, storing or transmitting sensitive data. We highlight the government-specific security requirements⁷, as follows:

Availability. All participants placed significant importance on availability and an overall guaranteed availability of approximately 99.5%. The provision of availability also addresses the reliability of the services to guarantee uninterrupted services that meet the availability requirement, as a key requirement from the GAs, as follows:

“If consumers ask for 95% availability, then we will provide a specific topology, such as dual homed gateway to meet the requirements” (P1).

“As part of the availability requirement, we also provide a 24 × 7 monitoring service, response time, and resolution time. Additional requirements are related to the availability of Firewalls, IDS, IPS and Anti-DDoS Attacks” (P1, P9).

⁷ Government Security Requirements, <https://goo.gl/eGtLRi>, (Accessed March 2017).

Access Control. Our participants typically reported relatively strong support for availability. Similarly, our participants reported that access control mechanisms were also often used to control access to networked resources and data. Several participants specifically mentioned access control mechanisms, as follows:

“How to get an access to the data centre’s room? Is there a Log Book, whether the shelf is caged, and how to get the key to the caged rack?” (P1).

“What kind of traffic is allowed in or out” (P1, P3).

Authorisation. Several participants reported that they had determined the access rights of an entity. Three participants mentioned that authorisations were used to manage who can read data at a higher security level etc. as follows:

“To access the data, the user must be registered, and the role must be permitted by the owner of the data” (P6).

“As a service provider, we can only perform certain commands based on our privileges provided by the customer” (P1, P3).

Non Repudiation. Our participants indicated that SPs were required to maintain logs for monitoring and auditing purposes, as described in the following statement:

“To take precautions against unauthorised access, non-repudiation requirements can be added to record all activity on the devices” (P1).

Confidentiality. Many participants had no idea when we asked them whether they had implemented specific security capabilities in relation to confidentiality requirements and objectives in their services. However, our participants pointed out that specific security requirements from the GAs could impose such data confidentiality, as follows:

“When it comes to confidentiality of data, data classifications are of paramount importance to define. We also need to know whom the owner of that data is to determine the authorised user” (P5).

“When encryption has been performed at the provider side, the customer should hold the key in terms of key management” (P1).

From the above discussion, several challenges were described regarding the government-specific security requirements. The participants confirmed that the GAs did not demand specific security requirements for external SPs, which supply such services to them. However, the GAs placed particular security standard, namely ISO 27001 as the key security consideration for the government procurement (see footnote no. 3). We then define the following propositions:

Proposition 3. *Service providers with a clear understanding of [security requirements] will be more likely to provide an appropriate level of trust by implementing specific [security capabilities].*

It was hypothesised that formulating security requirements plays an important role in mitigating perceived threats. However, our findings show that very little was found on the adoption of security considerations in the government procurement because of the difficulty of specifying all security requirements [2]. Despite the strong need for compliance with the security standards (e.g. ISO 27001), there is also the need for minimum security requirements in place when selecting external SPs (e.g. cloud services). Another lesson learned from this study is that existing regulations do not adequately support security procurement language for the government auctions. For instance, the Internet services, which are widely used in day to day government businesses, are still reliant on external SPs (considering ISO 27001 as a common security examination designed for government procurement). Such external services are selected annually for every year's budget. However, we identified a lack of basic technical protection to mitigate common threats when providing such external services to the GAs. This finding, while preliminary, suggests that it is necessary to classify security capabilities according to threats to establish the level of trust required between the GAs and external SPs.

Proposition 4. *Formulating [security requirements] is a fundamental part of incorporating appropriate [security capabilities] into the formulation of security-related SLAs.*

The results of this study indicate that all participants reported no specific security requirements were considered as instruments of selecting external SPs that provide such services to the GAs. Interestingly, another lesson learned from this study is that the GAs do not initially know what they want, or come up with new ideas about what and how to protect, what types of threats to mitigate, what types of security requirements that need to be defined, and which security capabilities that need to be employed. In some cases, most of the GAs rely on the ISO 27001/2 standards to form a strong security foundation. Indeed, it is not possible for the SPs to identify a complete security requirements up-front because security incidents occur many times and come later. The participants suggested that the GAs need to define the high-level security requirements up-front. Detailed security requirements are gathered as needed. It is evident that the diversity of security requirements can address unreasonable risks that were unlikely to occur.

3.3 Provider-Specific Security Capabilities

Some security capabilities are in place to demonstrate compliance with the government-specific security requirements. The statements made by participants indicate that threat-mitigation techniques have been normally conducted through technology capabilities because the GAs consider applying security

requirements for such external services by implementing security technologies. From the Delphi study data, whether or not **SPs** had experienced perceived threats, our participants reported that they had implemented some security capabilities, including technical elements, physical elements and human elements. We summarise the specifically mentioned security capabilities mentioned, and mapped each to security requirements [35] (Availability, Integrity, Non-Repudiation, Confidentiality, Authentication, and Authorisation).⁸

Technology Elements. In most cases, our participants mentioned using security technologies to protect their communication and information systems, as described in the below mentioned statements. We highlight provider's use of specific security technologies, as follows:

"We provide related requests, such as firewall, IDS, IPS and Anti-DDoS" (P5).

"For data in motion we can do encryption, using SSL, IPSec or VPN. For data at rest, we can make use of data encryption and data loss prevention, and for more advanced technologies for cloud customers, we can provide storage encryption or hardware security module" (P4).

Physical Elements. Since all participants were industrial experts; we were particularly interested in other security capabilities that they have developed to protect their information system services (e.g. computing, communications and storage services). Several participants mentioned physical security measures used, such as doors, locks and surveillance tools, to deny unauthorised access to facilities and resources. For example, several participants pointed out that some security capabilities in relation to physical elements, as follows:

"We guarantee the availability of CCTV devices, door access and visitor access management" (P2).

"We log all activity that occurs to monitor and track all user activity" (P1).

Human Elements. We also uncovered a number of human elements as mitigation strategies, such as people, process, and procedures that they have developed to protect their infrastructure. For example, most participants pointed out that people and process elements are necessary to be considered, as follows:

"A set of controls should have to comply with controls in ISO 27001, as the controls do not only discuss technology but also process and people" (P5).

"It would be great if the customer already has a security policy and user access matrix to mitigate unauthorized access" (P1, P3).

⁸ Security Capabilities, <https://goo.gl/zuCt18>, (Accessed March 2017).

Note that the above statements demonstrate challenges for classifying security capabilities according to threats. We found that most of the SPs were reliant on the ISO 27001:2013 standard for providing better security services to the GAs. Our findings is consistent with our earlier observations, which showed that the SPs were required to hold the ISO 27001 certification for the government auctions at the value above IDR 5 billion, (see footnote no. 3). Consequently, the SPs must have such security certification when they provide such external services to the GAs particularly for high-assurance services. However, such certification cannot contribute to addressing emerging threats [2]. We then derive the following propositions:

Proposition 5. *There is a need for an approach that addresses the interplay of threats, security requirements and security capabilities in the formulation of security-SLAs.*

Based on the Delphi study data, the GAs heavily rely on the experience of the external SPs in defining security requirements and implementing appropriate security capabilities to defend government data against a range of applicable threats. Our participants confirmed that certifications schemes, such as ISO 27001, were necessary for meeting agreed-upon security capabilities for protecting government data (see footnote no. 3). However, there are several issues with relying on the ISO 27001, as this certification scheme is not sufficient to address specific threat that the GAs and SPs are attempting to counter [2]. Furthermore, the SPs reported that most of the GAs had no idea how to mitigate particular threats. One unanticipated finding was that implementing basic security capabilities is part of the SPs' initiatives to ensure the services remain available to the GAs based on SLAs. It seems that there is a connection between the level of trust and security capabilities of the SPs used to demonstrate compliance with the security requirements and to mitigate the perceived threats.

Proposition 6. *Classification of [security capabilities] specified in security-related SLAs according to [perceived threats] will be more likely to asses what is being claimed and achieved by service providers.*

Concerning this issue, we have learned that it is not possible to address every threat we have found. The results of this study show that security capabilities-related defensive technologies are commonly used for the GAs to mitigate threats. The findings further support the idea of technology-level implementation of defensive strategies are the fastest and easiest way to address one or more threats [35]. In this case, the GAs often take simple ways to address threats through technology-level implementations of mitigation strategies. However, despite the strong need for technology solutions, there is also the need for a perspective on human elements, which might still be a vulnerability, as the weakest link. Also, the participants reported that technology capabilities can be a major consideration, but it is not the only method in mitigating threats. It may be the case that the formulation and classification of security capabilities provided by the SPs can help the GAs to select appropriate security capabilities according to threats.

4 Discussion

We discuss the implications of our findings for governments, service providers and researchers working on security-related SLAs, and summarise the limitations of our study. We then discuss the relationships with related work.

4.1 Implications

The interesting finding was that most of the **GAs** placed significant importance on service availability. However, other security requirements, such as data confidentiality and integrity were not demanded by the **GAs**. To help explain this, concerns over data confidentiality and integrity in the use of such external services are already seen as inhibiting the adoption of data centre services and cloud-based services in the government procurement auctions (see footnote no. 2). However, it is apparent that ISO 27001 is often the only available way to demonstrate compliance with the government security requirements to provide a degree of security assurance, particularly for the government auctions at the value above *IDR 5 billion (GBP 320 thousand)*, (see footnote no. 3). Based on our findings, specification of other **security requirements**, particularly with regards to data confidentiality and integrity, are not considered in the existing SLAs, as it brings some security challenges, such as the cost of security services associated with data confidentiality and integrity specified in security-related SLAs. Interestingly, the **SPs** have incorporated other **security requirements** in terms of the availability of security facilities, such as firewalls, intrusion detection and access management.

So far, the total cost associated with the interplay of **perceived threats**, **security requirements** and **security capabilities** in the formulation of security-related SLAs becomes a more difficult calculation since it encompasses liability and compensation. Furthermore, our findings reveals that several assumptions have been made to understand the current challenges with expressing the **security requirements** and **security capabilities** in SLAs according to specific **perceived threats**. Our propositions will be used in future research as a foundation for developing such a conceptual framework, including how the **security capabilities** can be incorporated into the formulation of security-related SLAs.

Overall, identifying the **perceived threats** can drive the **security requirements**, which can impose appropriate **security capabilities**. In other words, *level of trust* between the **GAs** and external **SPs** can be determined by using specific **security capabilities** according to specific **perceived threats**.

4.2 Limitations

This study has three main limitations. Firstly, these results may be applicable only to the domain and context being studied [24]. The results are, to some extent, dependent on the research participants selected for this study and how

participants described their experiences. Our qualitative data relies on the statements of the participants, which might be subjective. However, we limit its effects by conducting a series of data collection activities using a three-rounds Delphi study. While the demographics of our participants were representative of major SPs particularly in Indonesia, we did observe that our participants had a deficit of experiences in the formulation of security-related SLAs, particularly with regards to data confidentiality and integrity. Secondly, the internal validity of this study is determined mainly by the evidence we have used to generate our propositions. To limit these weaknesses, we recorded the audio of group discussions, transcribed the recorded audio, and sent the results to the participants before the individual sessions began. Finally, this study was subject to the paucity of participants who participated in each round (6–11 participants), as our participants were limited to those who were permitted to participate. However, the number of participants is still acceptable, as basic elements for meta-themes were present as early as six interviews [26]. We could increase the confidence in our propositions by asking more experts working at major SPs that provide external computing, communications and storage services to the GAs in Indonesia or in different countries. However, this study was not designed to be largely generalizable, but it aimed to clarify existing ‘real world’ SLAs and explore how the SPs implement security-related SLAs within service provision.

4.3 Reflection with Related Work

An SLA is a binding agreement between a service provider and a customer that is widely used in a variety of contexts to claim the obligation of external SPs to deliver services according to service requirements [1,3]. The concept of security-related SLAs was first proposed by Henning [5], who pointed out that security-related SLAs have a lack of tangible and measurable services because security is not quantifiable and has not been expressed in such concrete terms in SLAs. The authors pointed out that it is not trivial to address the cost of security service required in contracts or SLAs, as security is challenging to measure and quantify.

This view is supported by Monahan and Yearworthy [6] who argue that statistical measures need to be captured and understood by customers and SPs to develop meaningful security-related SLAs. The authors explored basic examples, such as the measurable distribution of anti-virus signatures and how the formulation of security-related SLAs can be incorporated with certain legal and contractual instruments.

Similarly, Bernsmed et al. [3] asserted that existing security mechanisms should be formalised into a contract language, such as an SLA. With emerging remote services, such as cloud-based services, the authors pointed out that the absence of security properties in SLAs makes it impractical for external SPs to offer trustworthy services to their customers, especially when external SPs along with their suppliers are involved. However, the authors found that there are still many unresolved issues associated with the formulation of security-related SLAs.

Moreover, Jaatun et al. [4] pointed out that security-related SLAs are necessary for Internet services to help ensure that customers and external SPs have a shared understanding of security considerations expressed in SLAs for which customers receive the required level of security services. In most cases, the authors found that many SPs offer QoS guarantees (e.g. service availability) as part of their contracts. However, the lack of guarantees for security properties, such as data confidentiality and integrity, is a major drawback from the customers' point of view.

Guesmi and Clemente in [7] described security-related SLAs in relation to problems arise in cloud-based services. The authors noted that external SPs should be able to describe what they can supply regarding security capabilities specified in SLAs according to security requirements, which help the providers to convince the customers regarding their security capabilities. However, the authors found that existing cloud SPs do not adequately express security requirements in cloud SLAs.

Some consortia have proposed standards to generate security-related SLAs between customers and external SPs to comply with the customer's requirements, particularly in cloud computing, such as the Secure Provisioning of Cloud Services based on SLA Management (SPECS) [9], the Multi-Cloud Secure Applications (MUSA) [12], SLA-Ready [11] and SLALOM [10]. The SPECS project aims at offering a solution for such problems, developing and implementing an open source framework to offer Security-as-a-Service, by relying on the notion of security parameters specified in SLAs. The SPECS project is linked to a further project, called MUSA, a framework for facilitating security in multi-cloud applications. Similarly, SLA-Ready is a European initiative that aims to deliver a reference model for cloud SLAs that are designed for small and medium-sized enterprises (SMEs). SLALOM is another European initiative established to develop standardised SLAs and contract terms for cloud-based services, which is built on ISO standards as a baseline with the SLALOM templates.⁹

Questions have been raised by Luna et al. in [13] about the lack of assurance and techniques to quantify security. The authors noted that it is difficult to understand what security capabilities the customers have been paying for, when considering particular services. The authors introduced techniques to assess quantitatively the security level of protection offered by cloud SPs to allow customers to compare with other SPs, based on their security-related SLAs. However, it is necessary to implement advanced security metrics expressed in SLAs to improve assurance and trustworthiness in remote services, such as cloud-based services.

So far, there is a concern that the existing SLAs are usually limited to defining guarantees and regulations in terms of service availability and quality. Consequently, many external SPs to date have tended to focus on the system availability and performance aspects rather than security aspects (e.g. data confidentiality and integrity). This study focuses on the idea of investigating 'real-world' SLAs in terms of security guarantees. In so doing, GAs can understand the service capabilities regarding security that are provided by external SPs.

⁹ More details of research gaps, <https://goo.gl/8i0ISC>, (Accessed March 2017).

5 Conclusion

This paper has investigated existing ‘real world’ SLAs in terms of security guarantees across the five major selected SPs that provided external computing, communications and storage services to the GAs between 2010 and 2016, using Indonesia as a case study. We found that most of the SPs did not incorporate the security capabilities adequately into their SLAs, except for defining guarantees and regulations in terms of service availability and quality. This study has shown that most of the GAs placed significant importance on service availability, including response time and resolution time. One of the more significant findings to emerge from this study was that there were no security considerations expressed in existing SLAs. Another major finding was that most of the GAs applied the provision of service availability to demand additional means of confirming the security services supplied by the SPs. For example, the GAs require the availability of security facilities, such as the availability of firewalls, access controls, visitor access management, intrusion detection systems (IDS), intrusion prevention systems (IPS) and closed circuit television (CCTV). Hence, the results of this study indicate that there is a need for methods supporting security capabilities addressed in security-related SLAs to enhance the level of trust in service provision, as all participants confirmed that they encountered challenges to address data confidentiality and integrity in SLAs. Also, this study provides additional evidence with respect to the lack of formulation and classification of security capabilities specified in SLAs according to particular threats. Although this study is based on a selective sample of participants, the findings can illuminate security concerns for other governments to incorporate the interplay of threats, security requirements and security capabilities into SLAs.

Acknowledgements. This work was supported in part by the Indonesian Ministry of Communications and Information Technology under the Directorate of Information Security, and the Indonesia Endowment Fund for Education Scholarship (LPDP).

References

1. Ferrer, A.J., i Montanera, E.P.: The role of SLAs in building a trusted cloud for europe. In: Damsgaard Jensen, C., Marsh, S., Dimitrakos, T., Murayama, Y. (eds.) IFIPTM 2015. IAICT, vol. 454, pp. 262–275. Springer, Cham (2015). doi:[10.1007/978-3-319-18491-3_22](https://doi.org/10.1007/978-3-319-18491-3_22)
2. Böhme, R.: Security audits revisited. In: Keromytis, A.D. (ed.) FC 2012. LNCS, vol. 7397, pp. 129–147. Springer, Heidelberg (2012). doi:[10.1007/978-3-642-32946-3_11](https://doi.org/10.1007/978-3-642-32946-3_11)
3. Bernsmed, et al.: Security SLAs for federated cloud services. In: International Conference on Availability, Reliability and Security, pp. 202–209. IEEE (2011)
4. Jaatun, M.G., Bernsmed, K., Undheim, A.: Security SLAs – an idea whose time has come? In: Quirchmayr, G., Basl, J., You, I., Xu, L., Weippl, E. (eds.) CDARES 2012. LNCS, vol. 7465, pp. 123–130. Springer, Heidelberg (2012). doi:[10.1007/978-3-642-32498-7_10](https://doi.org/10.1007/978-3-642-32498-7_10)
5. Henning, R.R.: Security service level agreements: quantifiable security for the enterprise? In: Proceedings of the 1999 workshop on New security paradigms, pp. 54–60. ACM (1999)

6. Monahan, B., Yearworth, M.: Meaningful security SLAs. Technical report, HP Labs (2008)
7. Guesmi, et al.: Access control and security properties requirements specification for clouds' SECLAS. In: IEEE Conference on Cloud Computing Technology and Science (2013)
8. Takahashi, T., et al.: Tailored security: building nonrepudiable security service-level agreements. *IEEE Veh. Technol. Mag.* **8**, 54–62 (2013)
9. Rak, et al.: Security as a service using an SLA-based approach via SPECS. In: IEEE Conference on Cloud Computing Technology and Science, pp. 1–6 (2013)
10. SLALOM Project. The SLALOM project website (2015)
11. SLA Ready Consortium. The SLA ready project website (2015)
12. Rios, et al.: Towards Self-Protective Multi-Cloud Applications (2015)
13. Luna, J., et al.: Quantitative reasoning about cloud security using service level agreements. *IEEE Trans. Cloud Comput.*, p. 1 (2015)
14. Cabinet Office: Procurement policy note-use of cyber essentials scheme certification (2016)
15. Hadeka, S., Scheimer, M.: DoD Amends its DFARS Safeguarding and Cyber Incident Reporting Requirements with a Second Interim Rule (2016)
16. Bird, et al.: China introduces new cybersecurity for rules for banking procurement (2016)
17. Nugraha, Y.: Security assurance requirements engineering (STARE) for trustworthy service level agreements. In: IEEE Conference on Requirements Engineering, pp. 398–399 (2015)
18. NIST 800-53: Security and privacy controls for federal information systems and organisations (2013)
19. Nugraha, Y., et al.: An adaptive wideband delphi method to study state cyber-defence requirements. *IEEE Trans. Emerg. Top. Comput.* **4**, 47–59 (2016)
20. Harrell et al.: Data Collection Methods, RAND Corporation (2009)
21. Paul, G., et al.: Methods of data collection in qualitative research. *Nature*, 291–295 (2008)
22. McGregor, et al.: Investigating the computer security practices and needs of journalists. In: 24th USENIX Security Symposium (USENIX Security 15), pp. 399–414 (2015)
23. Egelman, et al.: Are you ready to lock? In: ACM CCS, pp. 750–761 (2014)
24. Charmaz, K.: *Constructing Grounded Theory*. Sage, London (2014)
25. Birks, M., Mills, J.: *Grounded Theory: A Practical Guide*. Sage, London (2015)
26. Guest, G., et al.: How many interviews are enough? *Field Methods* **18**, 59–82 (2006)
27. Pivrinta, et al.: Grounding theory from Delphi studies. In: International Conference on Information Systems, pp. 2022–2035 (2011)
28. Howard, K.: Educating cultural heritage information professionals for Australia's galleries, libraries, archives and museums: A grounded Delphi study, Ph.D dissertation, QUT (2015)
29. Turoff, M.: The design of a policy Delphi. *Technol. Forecast. Soc. Change* **2**(2), 149–171 (1970)
30. Schmidt, R., et al.: Identifying software project risks: an international Delphi study. *J. Manag. Inf. Syst.* **17**(4), 5–36 (2001)
31. Okoli, C., et al.: The Delphi method as a research tool. *Inf. Manag.* **42**, 15–29 (2004)
32. Forsyth, D.: Delphi technique. In: Levine, J., Hogg, M. (eds.), *Encyclopedia of Group Processes & Intergroup Relations*, pp. 196–198. SAGE Publications (2010)

33. Delbecq, et al.: *Group Techniques for Program Planning*. Scott Foresman (1975)
34. Hsu, C., Sandford, B.: Delphi technique. In: Salkind, N.J. (ed.) *Encyclopedia of Research Design*, pp. 344–347. SAGE Publications (2010)
35. Shostack, A.: *Threat Modeling: Designing for Security*. Wiley, Hoboken (2014)

Applications of Trust

Managing Software Uninstall with Negative Trust

Giuseppe Primiero and Jaap Boender^(✉)

Department of Computer Science, Middlesex University, London, UK
{G.Primiero,J.Boender}@mdx.ac.uk

Abstract. A problematic aspect of software management systems in view of integrity preservation is the handling, approval, tracking and eventual execution of change requests. In the context of the relation between clients and repositories, trust can help identifying all packages required by the intended installation. Negative trust, in turn, can be used to approach the complementary problem induced by removing packages. In this paper we offer a logic for negative trust which allows to identify admissible and no-longer admissible software packages in the current installation profile in view of uninstall processes. We provide a simple working example and the system is formally verified using the Coq theorem prover.

1 Introduction

Software management configuration is among the most pervasive problems in modern personal computing, with complications caused by multiplication of users, required support for several software versions releases, increasing customization options and the need of coordination across distributed systems. One specific aspect of configuration management activities is change management, i.e. the handling, approval and tracking of change requests, with the aim of preserving the integrity of the system.

Consider the following example. A user interacts with a software package system to install or remove applications. The set of packages installed on a machine is called the installation profile of that machine. A valid installation profile is one which meets all the dependencies and conflicts clauses of all the packages installed and such that it satisfies sufficient dependencies for any desired package to be executed. Assume the current installation profile contains: two packages ϕ_1, ϕ_2 from the main repository; one package ψ_1 from the free repository; and one package ξ_1 from the non-free repository. Assume moreover that ψ_1 depends on ϕ_1 and from ϕ_2 , while ξ_1 depends on ψ_1 . Consider now the situation where the user wishes to prevent installation of a given additional package ψ_n from the free repository, while wishing to install a package ξ_2 from the non-free repository: which other packages is she safe in installing? and which ones does she need to remove in order to avoid conflicts in the new installation?

Determining these consistency relations between packages in a given installation is essential for system stability, but also to prevent the possibility of security threats in critical systems.

In [15], the problem of maintaining profile consistency and system integrity in view of uninstall processes is presented in the following terms:

Definition 1 (*Uninstall Problem*). *Given a new package ϕ to install, determine the minimal number of packages (possibly none) that must be removed from the system in order to make ϕ installable.*

This means identifying and removing all packages that are in conflict with the intended installation and its dependencies. This version of the problem can be complemented by that of identifying packages that depend on an undesired one.

In this context, trust can be used to characterize the relations between clients, software packages (including their dependencies) and repositories during the installation process. A software package in conflict with the current installation profile can not be trusted under it and hence not installed; if already installed, trust needs to be removed. Hence, dealing with such processes requires an explicit treatment of *negative trust*. Here and in the following the term *untrust* is used as neutral for ‘negative trust’ with respect to its derivatives *mistrust* and *distrust*: the former expresses trust removal, the latter trust denial. It should be noted that we refer to negative trust in the sense of being obtained through logical negation, as opposed to other quantitative approaches, where negative numbers are used. In [12] a natural deduction calculus is formulated which offers a proof-theoretical semantics for both notions. On this basis, we adapt here the Uninstall Problem from Definition 1 to the two semantics of untrust:

- A user identifies a package ϕ which generates conflict with a desired installation; to preserve profile consistency, ϕ is *distrusted* while the set of packages not depending on ϕ remain installable;
- A user identifies a package ϕ to be installed but in conflict with the current profile; to preserve profile consistency the packages ψ_i, \dots, ψ_n in the installation profile in conflict with ϕ are *mistrusted*.

The Uninstall Problem from Definition 1 can then be reformulated accordingly in the two variants:

Definition 2 (*Distrusted Uninstall Problem*). *Given a package ϕ that should not be installed, determine which other packages can be installed (i.e. that do not require ϕ).*

In this case, we are obviously interested in determining the maximal set of installable packages that do not conflict with ϕ .

Definition 3 (*Mistrusted Uninstall Problem*). *Given a package ϕ that should be installed, but which is in conflict with the current profile, determine which packages need to be uninstalled in order for ϕ to become installable.*

As in the approach from [15], we are interested here in determining the minimal set of packages inconsistent with ϕ that have to be removed from the installation profile.

In the present paper we provide a solution to these two problems in software management through their formalization in a logic for negative trust. In our model we use a trust function to allow access relations that presuppose consistency; in the current interpretation, trust (and hence of consistency checks) applies to software packages and conflicts are treated through negative trust. Note that the kind of inconsistencies we consider are not just those induced by technical requirements of the packages, but also by security issues. This formal strategy can help in offering a computable approach to trust management and in reducing risks related to installation profile inconsistency. The logic allows to reason about statements of the form:

Installation profile Γ allows consistent installation of package ϕ and prevents installation of conflicting package ψ .

This approach is in the first place novel from a conceptual point of view, because software dependency satisfaction as trust management has not yet been largely investigated. Secondly, it is novel from a technical point of view, as proof-theoretic solutions and the possibility of implementation in theorem provers for automatic inconsistency checking have been neglected so far. In comparison with existing approaches for the resolution of inconsistent installations, our underlying logic allows a finer-grained approach than, for example, SAT-solvers.

The paper is structured as follows. In Sect. 2 we offer an overview of related works in the area of computational trust and software management. In Sect. 3 we introduce the system (un)SecureND, which provides the formal machinery for our analysis. In Sect. 4 the Distrusted Uninstall Problem is reformulated within our logic and its solution illustrated. In Sect. 5 the same is done for the Mistrusted Uninstall Problem. In Sect. 6 we present a simple scenario modelled by example derivations showing both cases at work. We conclude with some general remarks and a brief overview of future work.

2 Related Work

The present work sits at the intersection of the literature on software dependency management and computational trust. In this section we briefly overview related works in both areas and compare those to our approach and results.

In [5], we have offered a trust-based version of the optimization problem from [15], known as the *minimum install problem*, determining the optimal way to install a new package, where optimality is determined by an objective function to minimize the amount of dependencies satisfied such that it results in a valid installation profile. Trust is then used to guarantee that the minimal amount of dependencies for each newly installed package is satisfied by transitively accessed repositories. The complementary problem of maintaining profile consistency and

system integrity in view of uninstall processes can be similarly developed by applying the logic from [12] to the software management context.

In the context of software management, SAT solving appears as a promising approach for the development of efficient methods of dependency graph resolution. SAT technology has been used in [9] to validate dependencies and check installability of packages of specific Linux distribution. In Sect. 1 we have illustrated our current task as resolving two variants of the *Uninstall Problem* from [15]. In that work the Opium package-management tool is introduced, also based on pseudo-boolean solvers. Opium is complete with respect to solution finding and can optimize a user-defined function, e.g. to prefer smaller packages over larger ones. An implementation of Opium is available as the 0install solver.¹ A review of state-of-the-art package managers and their ability to keep up with evolution and their dependency solving abilities is offered in [1], with a proposal to treat dependency solving as a separate concern from other upgrade aspects. The upgrade problem is also considered in [2] to justify the design of a modular package manager. While we do not have an implementation of preferential settings based on user-choices, our installation profiles are defined according to a criterion of minimality for dependency satisfaction: this means that we construct installation profiles according to an ordered criterion of dependency satisfaction and package removal from a profile always proceeds to identify the minimal number of required packages. Also, in our approach we do not explicitly distinguish cases of upgrade as separate from installation of new packages: this is clearly a simplification, but the system can deal with upgrade with the more complex tactic of removing older versions and installing newer ones. The solvability of the decision problem related to software dependency management and its optimization are also considered in [3]. In the present paper our aim is to start an investigation in a proof-theoretical and trust-based approach to software dependency management, which so far has been neglected. We also hope to facilitate the introduction of automated theorem provers in the area, which can be beneficial in the checking process of intended installations in order to anticipate possible conflicts.

An associated but distinct issue is the *co-installability problem*: to quickly identify the components that can or cannot be installed together. It is related to boolean satisfiability and it is known to be algorithmically hard. It is shown to be especially complex for cases that include optimization by user preferences, where a combination of exact and approximate solving can help, [7]. In [16] a formally certified semantic method preserving graph-theoretic transformations is developed to associate to each concrete component repository a much smaller one with a simpler structure. One aspect of co-installability is that of reciprocal dependencies [4], which as mentioned more explicitly later is abstracted from in the present formulation. The *Mistrusted Uninstall Problem* formulated below replicates the intuition of the co-installability problem in the setting for external packages (and their dependencies) which are in explicit conflict with currently

¹ See <http://0install.net/solver.html>. An OCaml implementation is also available at <http://roscidus.com/blog/blog/2014/09/17/simplifying-the-solver-with-functors/>.

installed ones (and those they depend on). As for the latter work and the work presented in [1], our system enjoys a formal translation to a library for the Coq theorem prover,² with the aim of verifying its results. Our system seems also to be the only one among those in the area of software management that relies on the explicit formulation of a natural deduction calculus.

An essential characteristic of the method implemented in our system is that integrity checking on installation profiles is guaranteed through an explicit formulation of a trust access function on packages. The logic was first introduced in [13] and extended to deal with negative trust in [12]. Recently, research has started considering the advantages, implications and formal requirements needed to deal with the various aspects of negated trust, and in particular the different meanings that can be attached to mistrust and distrust, including the extension and limits of their transitivity and propagation protocols [6, 10, 11, 17]. Most current research ignores the difference between the procedural semantics of these two terms, possibly with the exception of [10], which presents mistrust as misplaced trust, untrust as little trust and distrust as no trust. This approach abstracts, though, from the reasons behind the attribution of these evaluations, in favour of a purely quantitative approach. Propagation for negative (first-order) trust is formulated in [8]. Our contribution relies on a strict distinction between *distrust* and *mistrust*: the former is intended as trust denied to packages coming from outside of the current installation profile in view of inconsistencies with currently installed ones; the latter is understood as trust revoked to installed packages, in view of desired new packages to be installed. These two cases have not been in general treated separately. Our approach formalises them in the context of uninstall operations, which as far as we are aware are entirely missing from the literature. Moreover, treating (un)install operations in terms of (un)trust allows us to integrate a consistency check performed over profiles that satisfy dependencies for the packages involved.

3 (un)SecureND

(un)SecureND is a natural deduction calculus defining trust, mistrust and distrust protocols introduced in [13] for the positive fragment and in [12] for the negation complete extension. We offer here a slightly modified version adapted for the software management problems at hand. In particular, the present version introduces a strict partial ordering on formulas to express package dependency; this is then lifted at the level of contexts to express rules for installation profile construction and finally imported at the level of repositories where the associated packages are located. In view of this order relation the system qualifies as a substructural logic, in that Weakening is constrained by a trust function, Contraction and especially Exchange by the order relation.

We start with introducing the language of our logic:

² The repository is available at <https://github.com/gprimiero/SecureNDC>.

Definition 4 (*Syntax of (un)SecureND*)

$$\begin{aligned}
\mathcal{S}^\sim &:= \{A < B < \dots\} \\
\phi^S &:= a^S \mid \neg\phi_i^S \mid \phi_i^S \rightarrow \phi_j^S \mid \phi_i^S \wedge \phi_j^S \mid \phi_i^S \vee \phi_j^S \mid \perp \mid \text{Read}(\phi^S) \mid \text{Write}(\phi^S) \mid \text{Trust}(\phi^S) \\
\Gamma^S &:= \phi_i^S \mid \phi_i^S < \phi_j^S \mid \Gamma^S; \phi_j^S
\end{aligned}$$

3.1 Repositories, Packages and Dependencies

\mathcal{S}^\sim is the set of software repositories ordered by $<$ in view of dependencies between packages they contain, obtained below as lifting from package dependency. ϕ^S is a meta-variable for formulae, expressing software packages and their logical composition inductively defined by connectives, including operations to read (query), trust (consistency checking) and write (install). The language includes \perp to express conflicts: we formulate $\neg\phi_i^A$ as an abbreviation for $\phi_i^A \rightarrow \perp$. Packages are typed by their origin in repositories: ϕ_i^S says that package ϕ_i can be retrieved from repository $S \in \mathcal{S}$. An installation profile Γ^S is the list of all packages sufficient to an access or execution operation; a profile is internally structured to reflect the dependency of packages through the partial order $<$ in \mathcal{S}^\sim . We allow extension of profiles by packages that are not dependent on previous ones, denoted by $\Gamma^S; \Gamma^{S'} = \{\phi_i^S < \dots < \phi_n^S; \phi_{n+1}^{S'}\}$. This construction allows us to consider installation profiles that have all the sufficient conditions for the valid execution of a package, but can also be extended with additional packages. When such extension comes from the same repository, we use a comma: Γ^S, ϕ_i^S . The partial order allows for branching in the hierarchy, so that e.g. $\phi_1^S < \phi_2^S < \phi_3^S$ and $\phi_1^S < \phi_2^S < \phi_4^S$, i.e. packages ϕ_3^S, ϕ_4^S have both dependencies on ϕ_2^S and transitively on ϕ_1^S , but ϕ_3^S, ϕ_4^S could have no dependencies on each other.

Definition 5 (Judgements). An (un)SecureND-judgement $\phi_i^A \vdash \psi_j^B$ says that a package ψ_j from repository B can be validly executed under a profile containing package ϕ_i from repository A .

Definition 6 (Validity). An (un)SecureND-judgement $\vdash \phi_i^A$ says that a package ϕ_i from repository A can be executed in any profile.

We now generalise the dependency relation between packages $\phi_i^A < \psi_j^B$ at the level of repositories. A partial order relation $<$ over $\mathcal{S} \times \mathcal{S}$ intuitively expresses that dependencies are satisfied across repositories.

Definition 7. $A < B$ iff $\exists\phi_i^A, \psi_j^B$ s.t. $\phi_i^A < \psi_j^B$ and $\neg\exists\phi_k^A, \psi_l^B$ s.t. $\psi_l^B < \phi_k^A$.

By the first clause in Definition 7, $A < B$ means that some package in A satisfies a dependency for a package in B . By the second clause in Definition 7, our order relation abstracts from the issue of reciprocal dependencies. As noted in [4], two packages that mutually depend on each other will either be installed together, or not installed at all. They can therefore be considered as a single package for dependency resolution purposes. Rules from Fig. 1 define installation profiles construction from packages dependencies. Here we use the extra-theoretical typing

$$\begin{array}{c}
 \frac{}{\{\} : profile} \text{ Empty Profile} \qquad \frac{\vdash \phi_i^A}{\phi_i^A : profile} \text{ Package Insertion} \\
 \\
 \frac{\Gamma^A, \phi_i^A : profile \quad \Gamma^A, \phi_i^A \vdash \psi_j^B}{\Gamma^A, \phi_i^A < \psi_j^B : profile} \text{ Dependency Insertion} \\
 \\
 \frac{\Gamma^A : profile \quad \vdash \psi_j^B}{\Gamma^A; \psi_j^B : profile} \text{ Profile Extension}
 \end{array}$$

Fig. 1. The system (un)SecureND: profile construction rules

declaration $:profile$ to state that a formal expression can be considered a valid installation profile. By Empty Profile, an installation profile can be empty (base case); by Package Insertion, the elements in an installation profile are packages; by Dependency Insertion, a profile can be extended by satisfied dependencies; by Profile Extension, if a package can be validly executed in an empty profile, it can be added to an existing profile. Notice that unnecessary packages from any repository can still be added: this is possible for packages without dependencies through the Profile Extension rule, but more in general by an application of the Weakening Rule (see Fig. 4). The result of such a profile extension is denoted by $\Gamma^A; \phi^B$ and $\Gamma^A; \Gamma^B$. It is worth noting that Weakening will preserve profile consistency as it requires additionally an instance of the *trust* rule (see Fig. 3).

3.2 Rules for Package Execution

The operational rules in Fig. 2 formulate compositionality of package execution. A judgement of the form $\Gamma^A \vdash \phi^B$ says that package ϕ from repository B is executable without errors within an installation profile with packages coming from repository A .

The rule *Atom* establishes valid package execution within the same installation profile and across repositories with satisfied dependencies. In the present version we assume $A < B$. \perp says that if a profile is inconsistent, any package whatsoever can be executed. \wedge -I allows composition of packages from distinct profiles; by \wedge -E, each composing package can be obtained from the combined profiles (with $I = \{A, B\}$). \vee -I says that a combined profile can access any package from each of the composing profiles; by the elimination \vee -E, each package consistently inferred by each individual profile can also be executed under the extended profile. \rightarrow -Introduction expresses inference of a package from a combined profile as inference between packages (Deduction Theorem); its elimination \rightarrow -E allows to recover such inference as profile extension (Modus Ponens).

$$\begin{array}{c}
\frac{\Gamma^A; \Gamma^B : \text{profile}}{\Gamma^A; \Gamma^B \vdash \psi_i^B} \text{Atom, for any } \psi_i^B \in \Gamma^B \qquad \frac{\Gamma^A \vdash \perp}{\Gamma^A \vdash \phi^B} \perp \\
\\
\frac{\Gamma^A \vdash \phi_i^A \quad \Gamma^B \vdash \phi_j^B}{\Gamma^A; \Gamma^B \vdash \phi_i^A \wedge \phi_j^B} \wedge\text{-I} \qquad \frac{\Gamma^A; \Gamma^B \vdash \phi_i^A \wedge \phi_j^B}{\Gamma^A; \Gamma^B \vdash \phi_{i/j}^I} \wedge\text{-E} \\
\\
\frac{\Gamma^A; \Gamma^B \vdash \phi_{i/j}^I}{\Gamma^A; \Gamma^B \vdash \phi_i^A \vee \phi_j^B} \vee\text{-I} \qquad \frac{\Gamma^A; \Gamma^B \vdash \phi_i^A \vee \phi_j^B \quad \phi_{i/j}^{I \in \{A, B\}} \vdash \psi_k^C}{\Gamma^A; \Gamma^B \vdash \psi_k^C} \vee\text{-E} \\
\\
\frac{\Gamma^A; \phi_i^B \vdash \phi_j^C}{\Gamma^A \vdash \phi_i^B \rightarrow \phi_j^C} \rightarrow\text{-I} \qquad \frac{\Gamma^A \vdash \phi_i^B \rightarrow \phi_j^C \quad \Gamma^A \vdash \phi_i^B}{\Gamma^A; \phi_i^B \vdash \phi_j^C} \rightarrow\text{-E}
\end{array}$$

Fig. 2. The system (un)SecureND: operational rules

3.3 Access Rules

In Fig. 3 we present the access rules. These allow a user's installation profile to act on packages available from a distinct repository. In particular, we formulate a rule to query a package from a repository (*read*) and one to install a package within a profile (*write*). A third rule is formulated to guarantee that only packages consistent with the installation profile can be installed (*trust*).

read says that from any consistent profile Γ^A a package ϕ_i^B can be read provided its dependencies are satisfied (if any). *trust* works as an elimination rule for *read*: it says that if a package ϕ_i^B can be read and it preserves profile consistency, then it can be trusted. *write* works as an elimination rule for *trust*: it says that a readable and trustable package can be installed. *exec* says that every package that is safely installed in a consistent profile can be executed in it. The Introduction rule for distrust DTrust-I expresses the principle that a package ϕ_i^B non-consistent with its installation profile can be negated to be trustworthy; the corresponding elimination DTrust-E uses \rightarrow -introduction to induce *write* of any package consistent with the conflict resolution. The Introduction rule for mistrust MTrust-I says that trust is removed for local packages conflicting with an intended installation (a queried package); the corresponding MTrust-E allows to trust any package which is consistent with the conflict resolution by removal of the mistrusted package in the installation profile. This holds for any required dependency in other repositories, as expressed by the side condition that requires checking for any $C < B$. By the latter set of rules, *distrust* is a flag for preventing installation of conflicting external packages, while *mistrust* is a flag for facilitating removal of conflicting packages present in the installation profile. Notice that both untrust functions are triggered by the querying operation on a repository, hence conflicts are highlighted before installation.

$$\begin{array}{c}
 \frac{}{\Gamma^A \vdash \text{Read}(\phi_i^B)} \text{read} \\
 \\
 \frac{\Gamma^A \vdash \text{Read}(\phi_i^B) \quad \Gamma^A; \phi_i^B : \text{profile}}{\Gamma^A \vdash \text{Trust}(\phi_i^B)} \text{trust} \\
 \\
 \frac{\Gamma^A \vdash \text{Read}(\phi_i^B) \quad \Gamma^A \vdash \text{Trust}(\phi_i^B)}{\Gamma^A \vdash \text{Write}(\phi_i^B)} \text{write} \qquad \frac{\Gamma^A \vdash \text{Write}(\phi_i^B)}{\Gamma^A \vdash \phi_i^B} \text{exec} \\
 \\
 \frac{\Gamma^A \vdash \text{Read}(\phi_i^B) \rightarrow \perp}{\Gamma^A \vdash \neg \text{Trust}(\phi_i^B)} \text{DTrust-I} \\
 \\
 \frac{\Gamma^A \vdash \neg \text{Trust}(\phi_i^B) \quad \Gamma^A \vdash \neg \text{Trust}(\phi_i^B) \rightarrow \psi_j^C}{\Gamma^A \vdash \text{Write}(\psi_j^C)} \text{DTrust-E} \\
 \\
 \frac{\Gamma^A \vdash \text{Read}(\psi_i^B) \rightarrow \perp \quad \Gamma^A \setminus \{\phi_j^A\} : \text{profile}}{\Gamma^A \setminus \{\phi_j^A\}; \psi_i^B \vdash \neg \text{Trust}(\phi_j^A)} \text{MTrust-I} \\
 \\
 \frac{\Gamma^A \setminus \{\phi_j^A\}; \psi_i^B \vdash \neg \text{Trust}(\phi_j^A) \quad \Gamma^C; \psi_i^B : \text{profile}}{\Gamma^A \setminus \{\phi_j^A\}; \Gamma^C \vdash \text{Trust}(\psi_i^B)} \text{MTrust-E, } \forall C < B
 \end{array}$$

Fig. 3. The system (un)SecureND: access rules

3.4 Structural Rules

Structural rules hold with restrictions for (un)SecureND, see Fig. 4. As a result the system qualifies as substructural, see e.g. [14].

Weakening is constrained by an instance of *trust*: it says that a valid installation of ϕ_i^A is preserved under a profile extension in view of a trusted package ϕ_j^B , i.e. one whose profile extension is provably consistent.

Contraction is constrained by preservation of package ordering: it says that a valid installation of ϕ_k^A is preserved when removing an instance of identical packages $\phi_i^A; \phi_i^B$, provided one preserves the package from the higher repository in the order dependency, so as to guarantee any further dependency below.

Exchange is doubly constrained by order: it says that a valid installation of ϕ_k^A is preserved under reorder of packages ϕ_i, ϕ_j , if those come from the same repository A and if there is no involved dependency between them.

Finally, the Cut rule expresses valid package execution under profile extension: if a package ϕ_i^B is validly executed under profile Γ^A and a profile Γ^B including ϕ_i^B allows execution of a package ϕ_j^B , then the extended profile $\Gamma^A; \Gamma^B$ allows execution of ϕ_j^B .

$$\begin{array}{c}
\frac{\Gamma^A \vdash \text{Write}(\phi_i^A) \quad \Gamma^A \vdash \text{Trust}(\phi_j^B)}{\Gamma^A; \phi_j^B \vdash \text{Write}(\phi_i^A)} \text{Profile Weakening} \\
\\
\frac{\Gamma^A, \phi_i^A; \phi_j^B \vdash \text{Write}(\psi_k^A) \quad A < B}{\Gamma^A, \phi_i^A \vdash \text{Write}(\psi_k^A)} \text{Profile Contraction} \\
\\
\frac{\Gamma^A, \phi_i^A, \phi_j^A \vdash \text{Write}(\phi_k^A) \quad \phi_i^A \not\prec \phi_j^A}{\Gamma^A, \phi_j^A, \phi_i^A \vdash \text{Write}(\phi_k^A)} \text{Profile Exchange} \\
\\
\frac{\Gamma^A \vdash \phi_i^B \quad \Gamma^B, \phi_i^B \vdash \phi_j^B}{\Gamma^A; \Gamma^B \vdash \phi_j^B} \text{Profile Cut}
\end{array}$$

Fig. 4. The system (un)SecureND: structural rules

4 The Distrusted Uninstall Problem

Consider a profile $\Gamma^A = \{\phi_1^A < \dots < \phi_n^A\}$ and a package ϕ_m^B which one wishes *not to install*. This might be due to a security constraint, or an explicit conflict in view of an installed package $\phi_i^A \in \Gamma$, which one explicitly wants to preserve. We call such a package ϕ_m^B *distrusted*. In the calculus, this corresponds to the conclusion of the DTrust-I rule

$$\Gamma^A \vdash \neg \text{Trust}(\phi_m^B)$$

The *Distrusted Uninstall Problem* is to determine which packages can be installed in Γ^A that do not depend on ϕ_m^B . Our formulation allows to express this principle as the request to obtain the maximal set of formulas $\{\psi_i^N\}$ from any repository $N \geq B$ such that

$$\Gamma^A \vdash \neg \text{Trust}(\phi_m^B) \rightarrow \{\psi_i^N\}$$

By DTrust-E, this guarantees the right to install ψ_i^N . The first step consists in transforming our problem in a formulation that removes the trust condition.

Lemma 1. $\Gamma^A \vdash \neg \text{Trust}(\phi_m^B) \rightarrow \psi_i^N$ iff $\Gamma^A; \neg \phi_m^B \vdash \psi_i^N$.

Proof. For the left-to-right direction: By the assumption $\Gamma^A \vdash \neg \text{Trust}(\phi_m^B)$ and consistency of negation, $\Gamma^A \vdash \text{Trust}(\neg \phi_m^B)$; similarly, from the premise $\Gamma^A \vdash \neg \text{Trust}(\phi_m^B) \rightarrow \psi_i^N$ and consistency of negation we get $\Gamma^A \vdash \text{Trust}(\neg \phi_m^B) \rightarrow \psi_i^N$. Now apply *write* to $\text{Trust}(\neg \phi_m^B)$ and eliminate the function through *exec*; by \rightarrow -E we obtain $\Gamma^A; \neg \phi_m^B \vdash \psi_i^N$.

For the right-to-left direction: By the assumption $\Gamma^A; \neg \phi_m^B \vdash \psi_i^N$ it holds $\Gamma^A; \neg \phi_m^B : \text{profile}$, which justifies $\Gamma^A \vdash \text{Read}(\neg \phi_m^B)$ by *read*, $\Gamma^A \vdash \text{Trust}(\neg \phi_m^B)$ by the previous and *trust* and $\Gamma^A \vdash \neg \text{Trust}(\phi_m^B)$ by \neg -distribution. It follows $\Gamma^A; \neg \text{Trust}(\phi_m^B) \vdash \psi_i^N$ by substitution from the assumption, and $\Gamma^A \vdash \neg \text{Trust}(\phi_m^B) \rightarrow \psi_i^N$ is obtained by \rightarrow -I.

We can now reduce the latter to an operation on all packages coming from the repository involved by the distrust operation:

Lemma 2. *If $\Gamma^A; \neg\phi_m^B \vdash \psi_i^N$ then $\Gamma^A; \Gamma^B \setminus \{\phi_m^B\} \vdash \psi_i^N$, for all consistent profiles Γ^B that include ϕ_m^B .*

Proof. Γ^A can be extended with every consistent package from B ; by definition $\Gamma^A; \neg\phi_m^B \vdash \neg\text{Trust}(\phi_m^B)$, hence by Weakening this is possible except for ϕ_m^B as it does not satisfy *trust*.

The above corresponds to finding the maximal set of formulas in Γ^B that allows to execute ψ_i^N without requiring ϕ_m^B in the profile. To this aim, it is enough to find all $\phi_l^B \not\prec \phi_m^B$, i.e. the set of packages in B that have no dependencies from ϕ_m^B .

What has been so far restricted to one repository, can now be generalised to any repository that preserves the dependency condition:

Lemma 3. *$\Gamma^A; \phi_l^N \vdash \text{Write}(\psi_i^N)$ iff $(\phi_l^N \not\prec \xi_m^N \not\prec \psi_i^N)$ for any distrusted package ξ_m^N and any repository $N > A$.*

Proof. For the right-to-left direction. Assume the following: $\Gamma^A; \phi_l^N \vdash \text{Write}(\psi_i^N)$ and $\Gamma^A; \phi_l^N \vdash \neg\text{Trust}(\phi_m^N)$. Then: if $\phi_l^N < \phi_m^N$, then $\Gamma^A; \phi_l^N \vdash \phi_m^N$ by Atom, contradicting the distrust assumption; and if $\phi_m^N < \phi_l^N$ then similarly $\phi_m^N \vdash \phi_l^N$ and by Weakening it is possible to obtain $\Gamma^A; \phi_l^N, \phi_m^N \vdash \text{Write}(\psi_i^N)$, again contradicting the distrust assumption.

For the left-to-right direction. Assume $(\phi_l^N \not\prec \phi_m^N \not\prec \psi_i^N)$ and $\Gamma^A; \phi_l^N \vdash \neg\text{Trust}(\phi_m^N)$. Then: because $\phi_l^N \not\prec \phi_m^N$, the second assumption above does not require to remove ϕ_l^N as by Lemma 2; and because $\phi_m^N \not\prec \psi_i^N$, installing the latter does not require installing the former. Hence $\Gamma^A; \phi_l^N \vdash \text{Write}(\psi_i^N)$ holds.

Finally, our main result is obtained:

Theorem 1 (Distrusted Uninstall). *Given a package ϕ_m^B distrusted under profile Γ^A , a package ψ_i^N can be installed in Γ^A iff $\phi_m^B \not\prec \psi_i^N$.*

Proof. From Definition 2 and Lemma 3 by substitution.

This last result identifies distrusted packages as those that have at least a dependency from one package conflicting with the current installation profile.

5 The Mistrusted Uninstall Problem

Consider a profile $\Gamma^A = \{\phi_1^A < \dots < \phi_n^A\}$ and a package ϕ_m^B which one wishes to install in it: in the calculus, this corresponds to the conclusion of an instance of the Write rule, $\Gamma^A \vdash \text{Write}(\phi_m^B)$. Assume that ϕ_m^B is in conflict with the given profile

$$\Gamma^A \vdash \text{Read}(\phi_m^B) \rightarrow \perp$$

The *Mistrusted Uninstall Problem* is to determine the set $\Phi^A = \{\phi_i^A \in \Gamma^A \mid \phi_i^A \rightarrow \neg\phi_m^B\}$ which should be removed when installing ϕ_m^B . We will call any such package ϕ_i^A a *mistrusted package*. Hence the problem is to identify the minimal set of formulas Φ^A such that for each $\phi_i^A \in \Phi^A$

$$\Gamma^A \setminus \Phi^A; \phi_m^B \vdash \neg\text{Trust}(\phi_i^A)$$

and by MTrust-E, given any other set of formulas Γ^C required by ϕ_m^B , it allows

$$\Gamma^A \setminus \Phi^A; \Gamma^C \vdash \text{Trust}(\phi_m^B)$$

We start by identifying the minimal subset of packages from the current installation profile that satisfies the conflict:

Lemma 4. *If $\Gamma^A \vdash \text{Read}(\phi_m^B) \rightarrow \perp$, then $\exists \Phi^A \subseteq \Gamma^A$ such that $\Phi^A = \{\phi_i^A < \dots < \phi_n^A\} \vdash \text{Read}(\phi_m^B) \rightarrow \perp$.*

Proof. $\forall \phi_i^A, \phi_j^A \in \Gamma^A$, if $\phi_i^A \vdash \text{Read}(\phi_m^B) \rightarrow \perp$ and $\phi_i^A < \phi_j^A$, then $\phi_j^A \vdash \text{Read}(\phi_m^B) \rightarrow \perp$. And $\forall \phi_h^A < \phi_i^A$, $\phi_h^A \vdash \text{Read}(\phi_m^B)$. Hence it suffices to identify the maximal ϕ_i^A in conflict with ϕ_m^B and to include it in Φ^A together with all packages in Γ^A that depend on it. We will call Φ^A a *maximally mistrusted set*.

Lemma 5. *Consider a maximally mistrusted $\Phi^A \subseteq \Gamma^A$ such that $\Phi^A \vdash \text{Read}(\phi_m^B) \rightarrow \perp$ as of Lemma 4. Then $\forall \phi_i^A \in \Phi^A$, $\phi_i^A < \text{Read}(\phi_m^B) \rightarrow \perp$.*

Proof. This holds by construction of Φ^A in Lemma 4 and the Dependency Insertion Rule.

Lemma 6. *If $\phi_i^A \vdash \text{Read}(\phi_m^B) \rightarrow \perp$, then $\phi_i^A \not\prec \phi_m^B$.*

Proof. Starting from $\phi_i^A \vdash \text{Read}(\phi_m^B) \rightarrow \perp$ we apply D-Trust-I, \neg -distribution, *write* and *exec* to obtain $\phi_i^A \vdash \neg\phi_m^B$, from which we obtain $\phi_i^A < \neg\phi_m^B$ from Dependency Insertion and $\phi_i^A \not\prec \phi_m^B$ by contraposition.

Theorem 2 (Mistrusted Uninstall). *Given a package ϕ_m^B to be installed under profile Γ^A , a package ϕ_i^A is mistrusted in Γ^A iff for all $\Gamma^A \subseteq \{\phi_i^A < \phi_j^A\}$*

1. $\Gamma^A \vdash \phi_j^A \rightarrow \neg\phi_m^B$,
2. $\phi_j^A < \text{Read}(\phi_m^B) \rightarrow \perp$ and
3. $\phi_i^A \not\prec \phi_m^B$.

Proof. The first condition is required by Lemma 5 to include all the dependencies in the maximally mistrusted set. The second condition holds from Lemma 6. Finally, the third condition holds by contradiction: if $\phi_i^A < \phi_m^B$, then $\phi_i^A \vdash \phi_m^B$ by Dependency Insertion; it follows by Weakening that $\phi_i^A; \phi_m^B : \text{profile}$ and hence $\phi^B \vdash \text{Trust}(\phi_i^A)$.

This last result identifies packages to be removed as those that are in maximally mistrusted set and do not satisfy any dependency for the package to be installed under the current profile.

6 An Example

Consider the simple scenario presented in Sect. 1 where a user has the following installation profile:

$$\Gamma^{m-f-nf} \left\{ \begin{array}{l} \Gamma^{main} = \{\phi_1^m, \phi_2^m\} \\ \Gamma^{free} = \{\psi_1^f\} \\ \Gamma^{nonfree} = \{\xi_1^{nf}\} \end{array} \right\}$$

with the following dependencies

$$\Gamma^{m-f-nf} \left\{ \begin{array}{l} \phi_1^m < \psi_1^f \\ \phi_2^m < \psi_1^f \\ \psi_1^f < \xi_1^{nf} \end{array} \right\}$$

Assume the user distrusts a package ψ_n^f , e.g. because it is considered harmful or unsecure. The Distrusted Uninstall Problem asks which packages can be further installed in Γ^{m-f-nf} without installing ψ_n^f . Consider now a package $\psi_2^f \not\prec \psi_n^f$, then the following derivation holds:

$$\frac{\frac{D}{\Gamma^{m-f-nf} \vdash \neg Trust(\psi_n^f)} \quad \frac{D'}{\Gamma^{m-f-nf} \vdash Read(\psi_2^f)} \quad \psi_n^f \not\prec \psi_2^f}{\Gamma^{m-f-nf} \vdash Write(\psi_2^f)}$$

In other words, flagging ψ_n^f as distrustful does not impede the installation of a package ψ_2^f if the latter does not depend on the former.

Assume moreover that the user wishes to install an additional package $\xi_2^{nf} > \phi_1^m$, but such that $\phi_2^m \vdash Read(\xi_2^{nf}) \rightarrow \perp$: in other words, ξ_2^{nf} depends on ϕ_1^m , but is in conflict with ϕ_2^m (which is possible, given the latter does not depend on ϕ_1^m). Then assuming a package ψ_2^f replacing the functionalities of ϕ_2^m , the following derivation holds:

$$\frac{\frac{\phi_2^m \vdash Read(\xi_2^{nf}) \rightarrow \perp \quad \phi_2^m < \psi_1^f}{\Gamma^{m-f-nf} \setminus \{\phi_2^m < \psi_1^f\}; \xi_2^{nf} \vdash \neg Trust(\phi_2^m < \psi_1^f)} \quad \psi_2^f; \xi_2^{nf} : profile}{\Gamma^{m-f-nf} \setminus \{\phi_2^m < \psi_1^f\}; \xi_2^{nf} \vdash Write(\psi_2^f)}$$

In other words the installation of ξ_2^{nf} requires removing $\phi_2^m < \psi_1^f$ and it is compatible with the installation of ψ_2^f .

7 Conclusions

In this paper we have formulated two variants to the Uninstall Problem. Each relies on a different semantic qualification of untrusted packages required to be

removed or prevented from installation in a given installation profile, in order to preserve consistency.

Our approach is grounded on the logic $(\text{un})\text{SecureND}$, including an explicit *trust* function on formulas to guarantee consistency check at each retrieval step (after a *read* function), before installation rights are granted for a package (by a *write* function). The fragment of the language presented in this paper allows to express negation over trust as a dis-installation requirement. Different pairs of introduction/elimination rules determine the selection of one of two resolution strategies: one flags a package external to the installation profile as distrusted and hence as not installable; the other identifies already installed packages to be removed. The selection takes care of identifying and removing all required dependencies. We have illustrated the working protocol through an easy example. As already mentioned, validation of the system is obtained by implementation of the $(\text{un})\text{SecureND}$ calculus as a large inductive type in the Coq proof assistant. The development is available at <https://github.com/gprimiero/SecureNDC>.

A characteristic of the logic $(\text{un})\text{SecureND}$ is its substructural nature, which in future work can be exploited to investigate cases of strengthened and limited resource redundancy for fault tolerance and source shuffling for security. Other applications of negative trust can be investigated to distinguish between malevolent and simply unsuccessful sources.

References

1. Abate, P., Di Cosmo, R., Treinen, R., Zacchiroli, S.: Dependency solving: a separate concern in component evolution management. *J. Syst. Softw.* **85**(10), 2228–2240 (2012)
2. Abate, P., DiCosmo, R., Treinen, R., Stefano Zacchiroli, M.P.M.: A modular package manager. In: Proceedings of the 14th International ACM Sigsoft Symposium on Component Based Software Engineering, CBSE 2011, pp. 179–188. ACM, New York (2011)
3. Le Berre, D., Parrain, A.: On SAT technologies for dependency management and beyond. In: Thiel, S., Pohl, K. (eds.) *Software Product Lines*, 12th International Conference, SPLC 2008, Limerick, Ireland, September 8–12, 2008, Proceedings. Second Volume (Workshops). Lero Int. Science Centre, pp. 197–200. University of Limerick, Ireland (2008)
4. Boender, J.: Formal verification of a theory of packages. *ECEASST* **48** (2011)
5. Boender, J., Primiero, G., Raimondi, F.: Minimizing transitive trust threats in software management systems. In: Ghorbani, A.A., Torra, V., Hisil, H., Miri, A., Koltuksuz, A., Zhang, J., Sensoy, M., García-Alfaro, J., Zincir, I. (eds.) *13th Annual Conference on Privacy, Security and Trust, PST 2015*, Izmir, Turkey, 21–23 July, 2015, pp. 191–198. IEEE (2015)
6. Guha, R.V., Kumar, R., Raghavan, P., Tomkins, A.: Propagation of trust and distrust. In: Proceedings of the 13th International Conference on World Wide Web, WWW 2004, New York, NY, USA, 17–20 May, pp. 403–412 (2004)
7. Ignatiev, A., Janota, M., Marques-Silvam, J.: Towards efficient optimization in package management systems. In: Proceedings of the 36th International Conference on Software Engineering, ICSE 2014, pp. 745–755. ACM, New York (2014)

8. Jøsang, A., Pope, S.: Semantic constraints for trust transitivity. In: Hartmann, S., Stumptner, M. (eds.) APCCM, vol. 43. CRPIT, pp. 59–68. Australian Computer Society (2005)
9. Mancinelli, F., Boender, J., Di Cosmo, R., Vouillon, J., Durak, B., Leroy, X., Treinen, R.: Managing the complexity of large free and open source package-based software distributions. In: 21st IEEE/ACM International Conference on Automated Software Engineering (ASE 2006), 18–22 September, Tokyo, Japan, pp. 199–208. IEEE Computer Society (2006)
10. Marsh, S., Dibben, M.R.: Trust, untrust, distrust and mistrust – an exploration of the dark(er) side. In: Herrmann, P., Issarny, V., Shiu, S. (eds.) iTrust 2005. LNCS, vol. 3477, pp. 17–33. Springer, Heidelberg (2005). doi:[10.1007/11429760_2](https://doi.org/10.1007/11429760_2)
11. Harrison McKnight, D., Chervany, N.L.: Trust and distrust definitions: one bite at a time. In: Falcone, R., Singh, M., Tan, Y.-H. (eds.) Trust in Cyber-societies. LNCS, vol. 2246, pp. 27–54. Springer, Heidelberg (2001). doi:[10.1007/3-540-45547-7_3](https://doi.org/10.1007/3-540-45547-7_3)
12. Primiero, G.: A calculus for distrust and mistrust. In: Habib, S.M.M., Vassileva, J., Mauw, S., Mühlhäuser, M. (eds.) IFIPTM 2016. IAICT, vol. 473, pp. 183–190. Springer, Cham (2016). doi:[10.1007/978-3-319-41354-9_15](https://doi.org/10.1007/978-3-319-41354-9_15)
13. Primiero, G., Raimondi, F.: A typed natural deduction calculus to reason about secure trust. In: Miri, A., Hengartner, U., Huang, N.-F., Jøsang, A., García-Alfaro, J. (eds.) 2014 Twelfth Annual International Conference on Privacy, Security and Trust, Toronto, ON, Canada, July 23–24, pp. 379–382. IEEE (2014)
14. Restall, G.: An Introduction to Substructural Logics. Routledge (2000)
15. Tucker, C., Shuffelton, D., Jhala, R., Lerner, S.: OPIUM: optimal package install/uninstall manager. In: 29th International Conference on Software Engineering, ICSE 2007, pp. 178–188 (2007)
16. Vouillon, J., Di Cosmo, R.: On software component co-installability. *ACM Trans. Softw. Eng. Methodol.* **22**(4), 34:1–34:35 (2013)
17. Ziegler, C.-N., Lausen, G.: Propagation models for trust and distrust in social networks. *Inf. Syst. Front.* **7**(4–5), 337–358 (2005)

Towards Trust-Aware Collaborative Intrusion Detection: Challenges and Solutions

Emmanouil Vasilomanolakis^(✉), Sheikh Mahbub Habib, Pavlos Milaszewicz,
Rabee Sohail Malik, and Max Mühlhäuser

Telecooperation Group, Technische Universität Darmstadt, Darmstadt, Germany
{vasilomano, habib, max}@tk.tu-darmstadt.de,
{pavlos.milaszewicz, rabeesohail.malik}@stud.tu-darmstadt.de

Abstract. Collaborative Intrusion Detection Systems (CIDSs) are an emerging field in cyber-security. In such an approach, multiple sensors collaborate by exchanging alert data with the goal of generating a complete picture of the monitored network. This can provide significant improvements in intrusion detection and especially in the identification of sophisticated attacks. However, the challenge of deciding to which extend a sensor can trust others, has not yet been holistically addressed in related work. In this paper, we firstly propose a set of requirements for reliable trust management in CIDSs. Afterwards, we carefully investigate the most dominant CIDS trust schemes. The main contribution of the paper is mapping the results of the analysis to the aforementioned requirements, along with a comparison of the state of the art. Furthermore, this paper identifies and discusses the research gaps and challenges with regard to trust and CIDSs.

1 Introduction

With the continuous growth of cyber-attacks, Intrusion Detection Systems (IDSs) are nowadays considered a mandatory line of defense for any type of network [6]. However, as isolated IDSs do not scale and are not capable of detecting distributed and highly sophisticated attacks, more collaborative approaches have emerged. The term Collaborative IDS (CIDS) describes systems that exhibit such a cooperative approach [10]. In a CIDS, a plethora of different sensors (e.g., honeypots, firewalls, IDSs, etc.) collaborate by exchanging alert data with the scope of creating a holistic picture of the monitored network. As sensors exchange data and correlate information, it becomes feasible to detect a larger portion of attacks. Moreover, in contrast to isolated IDSs that do not scale, these systems can monitor very large networks.

However, a big challenge in CIDSs is the ability to manage the various sensors in an efficient and productive manner. In this context, the aspect of trust is of high importance for CIDSs. First, with the usage of computational trust it is possible to deal with insider attacks [3]. Such attacks refer to cases in which a number of sensors, inside the CIDS, are infected or compromised. In such an

event, rogue sensors can significantly reduce the accuracy of the overall system by contaminating the alert exchange process with fake alerts. Second, apart from insider attacks, trust mechanisms are valuable for assessing the quality and thus the weight of importance that different sensors ought to have. For instance, in a large CIDS, a multitude of heterogeneous sensors is to be expected; from highly trusted IDSs to honeypots and/or to third party untrustworthy sources of alert data. In all cases, the CIDS needs to be able to assess which sources are more relevant and/or reliable.

In this paper, we attempt to bridge the areas of computational trust and collaborative intrusion detection, discuss the state of the art, and identify the respective research gaps. We firstly propose a number of requirements for reliable Trust Management (TM) in CIDSs. Afterwards, we carefully investigate the related work for the most dominant and promising CIDS trust schemes. The trust components of the identified systems are discussed separately on the basis of the aforesaid requirements. Furthermore, we compare all the trust mechanisms by mapping them to the requirements. In addition, based on our analysis, we identify and discuss research gaps and challenges with regard to trust and CIDSs.

The remainder of this paper is organized as follows. In Sect. 2, we propose a number of requirements for trust mechanisms in CIDSs. On this basis, Sect. 3 provides a brief description and analysis of the most prominent CIDS trust mechanisms. Furthermore, Sect. 4 contains a detailed comparison of these mechanisms by mapping them to the requirements and provides future directions. Lastly, Sect. 5 concludes the paper.

2 Requirements

Managing trust in a CIDS is a complex problem which has many conflicting requirements for which an acceptable tradeoff has to be chosen. In our previous work, we have examined the related work in both CIDSs [10] and computational trust [5]. On this basis, along with an additional study of the state of the art in trust mechanisms for CIDSs, we propose the following requirements. These will be utilized along the discussion of the different approaches in Sect. 3 and will be more extensively analyzed in Sect. 4.

- **Global view:** Some approaches for managing trust require a *global view* of the monitored network, in which an administrator has full control over the sensors or sensors have full-fledged information about the entire network. This is not always realistic; for instance, fully distributed CIDSs usually cannot guarantee such a global view. Hence, approaches that do not require global view can be applied to a larger variety of CIDSs.
- **Minimum overhead:** The overhead associated with the computing and managing of trust in a CIDS should be kept to a minimum. In particular, the overhead can be either communicational or computational. *Communicational overhead* refers to the need of the trust mechanism to generate additional messages. *Computational overhead* is associated with the computational power required to compute the various trust values.

- **Incentive mechanism:** An *incentive mechanism* refers to the ability of a trust system to motivate and reward sensors for behaving in a trustworthy manner. An example of such an incentive can be the ability to give alert data feedback only to sensors with high trust values.
- **Initial trust:** Assigning a reasonable *initial trust* value to a sensor that has recently joined the CIDS is a challenging task [8]. As historical data do not always exist for newcomers, the trust mechanism has to choose between assigning random values, a probation period approach or the assignment of high/low initial trust values. Each approach has certain advantages and disadvantages that will be discussed in the following sections.
- **Forgetting factor:** The *forgetting factor* (or aging) is a parameter that ensures that the most recent feedback, given by nodes, carries more weight than less recent feedback. This is desirable as it allows a more accurate and up-to-date calculation of trust values.
- **Performance history:** The *performance history* describes how a sensor has performed based on historical data (e.g., old transactions).

3 CIDS Trust Management

In this section, we analyze and discuss four trust approaches for CIDSs. The selected systems were identified by analyzing the state of the art. In particular, the emphasis of our analysis lies on the collaboration framework or architecture, the TM mechanisms, and the utilized evaluation methods.

3.1 Dirichlet-Based Trust Management

Fung et al. proposed a TM model to facilitate an effective trust-aware CIDS [4]. The system consists of three main components: the Collaboration component, the TM component and Acquaintance Management component. The Collaboration Framework connects different hosts in a network and allows them to communicate in a fair and scalable manner. The TM framework leverages the collaboration framework to establish trust among networked hosts based on the history of their performance. It uses Bayesian statistics to calculate the trustworthiness of hosts. Finally, the Acquaintance Management is used to manage a list of trustworthy acquaintances using test messages. Each of these components is described in the following.

Collaboration Component. This component has an incentive mechanism for hosts to share information and manage their acquaintances. Each host maintains a list of acquaintances, peers (i.e., other sensors in the CIDS) that it trusts and collaborates with. Each sensor sends two types of requests to its peers, *intrusion consultation messages* and *test messages*. Intrusion consultation messages are sent when a host needs feedback to determine whether an alarm should be raised or not. The amount of information that a host shares with a peer depends

on the trustworthiness of that peer; more trustworthy peers receive more information than less trustworthy ones. Additionally, each host sends test messages periodically. The nature of the test message is known to the sender beforehand. Test messages are used by a sensor to establish the trust levels of its peers. Such messages can be generated artificially using a knowledge database.

Trust Management Component. To establish trust, hosts send requests to peers and evaluate the satisfaction levels of the reply. The alert ranking raised by a host lies in the interval $[0,1]$ where 0 is harmless and 1 is highly dangerous. The satisfaction level of a reply from an acquaintance is a function of three parameters, the expected answer (r), the received answer (a) and the difficulty level (d) of the test message. The values of these three parameters also lie in the interval $[0,1]$. The function $Sat(r, a, d) (\in [0, 1])$ (see Eq. 1) represents the satisfaction level of the given feedback. The value of c_1 determines the level of penalization of a wrong estimate. Parameter c_2 controls sensitivity of the satisfaction level to the distance between the expected and received answer.

$$Sat(r, a, d) = \begin{cases} 1 - \left(\frac{a-r}{\max(c_1 r, 1-r)} \right)^{d/c_2} & a > r \\ 1 - \left(\frac{c_1(r-a)}{\max(c_1 r, 1-r)} \right)^{d/c_2} & a \leq r \end{cases} \quad (1)$$

Bayesian statistics, and specifically the Dirichlet distribution, is used to model the distribution of past satisfaction levels from the acquaintances of each peer. The Dirichlet distribution is utilized since the authors are modeling multi-valued satisfaction levels; it is a continuous, multivariate probability distribution, which is a generalization of the Beta Distribution for multivariate values. This prior distribution is then used to estimate the posterior distributions, i.e. satisfaction levels of future answers.

If X is the random variable representing the satisfaction level of feedback from a peer, then X can take values $\chi = \{x_1, x_2 \dots x_k\}$ of the supported levels of satisfaction where each value lies in the interval $[0,1]$. $\vec{P} = \{p_1, p_2 \dots p_k\}$ is the probability distribution vector of X such that $P\{X = x_i\} = p_i$. The cumulative observations and beliefs of X are represented by $\vec{\gamma} = \{\gamma_1, \gamma_2 \dots \gamma_k\}$. Using the Dirichlet distribution, the vector \vec{p} is modeled as:

$$f(\vec{p} | \xi) = Dir(\vec{p} | \vec{\gamma}) = \frac{\Gamma(\sum_{i=1}^k \gamma_i)}{\prod_{i=1}^k \Gamma(\gamma_i)} \prod_{i=1}^k p_i^{\gamma_i - 1} \quad (2)$$

where ξ is the background information which is represented by $\vec{\gamma}$. Let $\gamma_0 = \sum_{i=1}^k \gamma_i$, then the expected value of probability of X to be x_i is then given by: $E(p_i | \vec{\gamma}) = \frac{\gamma_i}{\gamma_0}$. Moreover, a forgetting factor λ can be used to give recent observations more importance which leads to: $\vec{\gamma}^{(n)} = \sum_{i=1}^n \lambda^{t_i} \times \vec{S}^i + c_0 \lambda^{t_0} \vec{S}^0$.

S_0 is the initial beliefs vector and c_0 is the constant which puts weight on the initial beliefs. t_i represents the time elapsed since the i^{th} evidence. When feedback is received from a peer, it is given a score according to Eq. 1. p_i^{uv} denotes the probability that feedback from peer v to peer u has the satisfaction value x_i . The sum of P^{uv} over all i is equal to 1. \vec{P}^{uv} is modeled using Eq. 2. Y^{uv} is the random variable such that: $Y^{uv} = \sum_{i=1}^k p_i^{uv} w_i$. The following equation then gives the trustworthiness of a peer:

$$T^{uv} = E[Y^{uv}] = \sum_{i=1}^k w_i E[p_i^{uv}] = \frac{1}{\gamma_0^{uv}} \sum_{i=1}^k w_i \gamma_i^{uv} \quad (3)$$

Where γ_i^{uv} is the cumulative evidence that v has replied to u with satisfaction level x_i . As soon as the trustworthiness has been calculated, feedback only from peers whose trustworthiness levels exceed a certain level is considered. An upper bound for the trust level is calculated using the covariance of p_i and p_j . Once feedback from acquaintances has been collected, it is aggregated using the following weighted majority formula:

$$\bar{a}_i^u = \frac{\sum_{T_i^{uv} \geq th^u, v \in A^u} T^{uv} a_i^{uv}}{\sum_{T_i^{uv} \geq th^u, v \in A^u} T^{uv}} \quad (4)$$

Where \bar{a}_i^u is the aggregated ranking of alert i . T^{uv} is the trustworthiness of peer v to peer u . This formula is applied only to feedback from peers with trustworthiness higher than a certain threshold. a_i^{uv} is the ranking of the alert i given from u to v .

Acquaintance Management. The authors contributed an algorithm to maintain a list of acquaintances in the proposed system. Maintaining such a list is necessary since it is not scalable for the host to keep records for all the peers in the network. Each host maintains a list of trusted nodes, with the length of the list depending on the available resources. Since it takes time to determine the trustworthiness of a peer, a probation list should also be maintained. The host communicates with peers in its probation list to determine their trust levels and if the levels exceed a certain threshold, the relevant nodes can be added to the acquaintance list.

Experiments and Results. The authors simulate an environment in which a host is allowed to have an acquaintance list with 40 dishonest peers, divided equally into 4 groups. Each group uses a different strategies for its dishonesty - complimentary, exaggerate positive, exaggerate negative and maximal harm. Complimentary simply inverts the alert level of a message. For example, an actual alert level of 0.7 will be converted to a 0.3. Exaggerate positives and exaggerate negatives convert low positives and negatives to high positives and

negatives [11]. In maximal harm, the peer reports false feedback with the intention of causing the most harm to the host. The trust values of the peers converge after 30 days and, as expected, the trust value of the peers using the maximal harm scheme is the lowest. Fung et al. also conduct an experiment in which a peer behaves honestly for 50 days and then launches a betrayal attack using a maximal harm scheme. The results indicate that the trust value of that peer decreases rapidly due to the forgetting factor used to associate more weight on recent messages. Additionally, once a peer is downgraded from “highly trustworthy” to “trustworthy” the rate of test messages sent to it is increased.

3.2 Trust Diversity

Perez et al. [9] proposed the notion of Trust Diversity (TD) to maximize the information quality and resilience against dishonest behavior of CIDS sensors. TD is defined as the measurement of the dispersion of the trust values of sensors in a given domain. For instance, a low diversity would indicate that the sensors exhibit similar trustworthiness. The goal is to find a placement of sensors such that TD is maximized in all given domains. Thus, all domains will have a roughly equal distribution of trustworthy and untrustworthy sensors, leaving no domain unprotected. Furthermore, when there is high TD, the more reliable sensors can help identifying untrustworthy sensors, making the system more resilient to insider attacks. Finally, higher TD also leads to a system which is more resilient to external attacks.

System Model. The services and resources of a CIDS can be divided into different *domains* (D). Each domain has a set of *requirements* (R) for the proper functioning of the entire system. Each sensor has certain *properties* (P) to monitor certain requirements, denoted by $P(S_j)$. For each sensor, a reputation value ($Rep(S_j)$) is also maintained. This value is based on the assessment of the sensor’s past behavior. The CIDS can be configured to deploy sensors as required in different domains to reach a desired goal.

Sensor Placement. Reconfiguring the placement of sensors is important to increase TD which reduces the uncertainty about the nature of events, that is, if they are malicious or not. It also allows the reassessment of trustworthiness in sensors, as the feedback from more trustworthy sensors can be compared with the feedback given from less trustworthy sensors. The authors propose a *Trust and Reputation module* which uses the past behavior of the sensors to monitor the quality of each domain. The module uses three metrics, the past behavior of a sensor, the past behavior of a sensor’s neighbors and the sensor’s capabilities. The final result is the trustworthiness value, calculated for a sensor considering these metrics. The value is then used to compute the TD in a given domain. Once the TD of all the domains has been computed, an optimization algorithm is used to generate the best possible reconfiguration of the sensors.

Trust and Reputation Management System. To compute the trustworthiness of sensors, Perez et al. make use of a trust and reputation management system. The computed trustworthiness can then be used to maximize TD. Computing the inter-quartile range, mean difference and arithmetic difference are some examples of how this diversity can be quantified. TD is computed at three different levels: at the requirement level such that diversity is maximized among sensors assigned to fulfill a certain requirement; at the domain level to ensure that there is a diverse spread of sensors' reputation levels in all domains; and lastly, at the global system level such that no domain is left unprotected. TD at the requirement level, for a requirement R_k for a domain Ω , denoted by $TD_\Omega \in [0, 1]$ can be calculated by:

$$TD_\Omega(R_k) = \max\{Rep_\Omega(S_{R_k})\} \cdot \psi(Rep_\Omega(S_{j,R_k}) \cdot \mu_\Omega(R_k, S_j)), \forall S_j \in SP(\Omega) \quad (5)$$

where ψ is the dispersion among the sensors' reputation, $\max\{Rep_\Omega(S_{R_k})\}$ denotes the highest reputation value among all sensors in the given domain Ω , $Rep_\Omega(S_{j,R_k})$ is the reputation of the j^{th} sensor in Ω fulfilling requirements R_k and $\mu_\Omega(R_k, S_j)$ is the risk incurred when R_k is not satisfied. Once the TD has been calculated at the requirement level, it can then be calculated at the domain level as: $TD_\Omega = \oplus_{k=1}^{\phi_\Omega(R)} TD_\Omega(R_k)$. $TD_\Omega(R_k)$ is the TD of the k^{th} requirement calculated in (5). $\phi_\Omega(R)$ represents the total number of requirements in Ω . \oplus is an aggregation operation, for example, arithmetic mean or harmonic mean. Finally, TD at the CIDS level can be calculated as: $TD_{CIDS} = \oplus_{i=1}^{\phi_{CIDS}(D)} TD_{D_i}$ where $\phi_{CIDS}(D)$ is the number of domains in the CIDS and TD_{D_i} is the TD of each domain.

Upon receiving the alert of a new event, the monitoring system first assesses the trust in the event being true and then updates the reputation levels of all relevant sensors which are configured to report such an event. The trust of an event is the confidence the system places on an event being true. Three factors are used to compute the trust level in an event: the agreement level of all relevant sensors; the number of domains in which the event was detected and the TD in all such domains. Using these three factors, the trust level of an alert can be calculated as follows:

$$T(E_{R_k}) = \oplus_{i=1}^{\phi_D(E_{R_k})} |\delta_{D_i}(E_{R_k})| \cdot TD_{D_i}(R_k) \quad (6)$$

where $\phi_D(E_{R_k})$ is the number of domains from which the event alert has been issued. $\delta_{D_i}(E_{R_k})$ is the level of agreement of the relevant sensors in the i th domain D_i , relevant to the event E_{R_k} . $TD_{D_i}(R_k)$ is the TD of each domain where the event happened. The agreement of relevant sensors on a given event is calculated using a voting scheme and can be computed as follows:

$$\delta_\Omega(E_{R_k}) = \frac{\sum_{j=1}^{\phi_{S_\Omega}(E_{R_k})} Rep_\Omega(S_{j,R_k})}{\phi_{S_\Omega}(E_{R_k})} - \frac{\sum_{j=1}^{\phi_{S_\Omega}(E_{R_k})} Rep_\Omega(S_{j,R_k})}{\phi_{S_\Omega}(\neg E_{R_k})} \quad (7)$$

Here, $Rep(S_{j,R_k})$ is the reputation of the j^{th} sensor fulfilling requirement R_k , $\phi_{S_\Omega}(E_{R_k})$ is the number of relevant sensors in domain Ω that have issued

a notification for the event, and $\phi_{S_\Omega}(-E_{R_k})$ is the number of sensors that did not report it. A neutral agreement with the computed value of 0 indicates total uncertainty about whether a given event is true or not. A value of 1 indicates full confidence that is true and a value of -1 indicates that it is false. Once a trust value is assigned to an event it can be used to update the reputation of the involved sensors as follows:

$$RepS_j^{(t)} = \omega RepS_j^{(t-1)} + (1 - \omega) \frac{\sum_{k=1}^{\phi_{S_j}(E)} Sat(S_j, E_k) \mu(R_{E_k S_j}) \xi(E_k)}{\phi_{S_j}(E)} \quad (8)$$

where $\phi_{S_j}(E)$ is the total number of events the sensor j has been involved in. $Sat(S_j, E_k)$ is the calculated satisfaction level of the behavior shown by S_j with regards to the event E_k . Moreover, $RepS_j^{(t-1)}$ is the last reputation value of S_j , while $\mu(R_{E_k S_j})$ is the associated risk of the requirement affected by E_k , and $\xi(E_k)$ is a forgetting factor. Lastly, ω is the weight on each term which determines importance of past behavior. The satisfaction level of the behavior of a sensor with regards to an event is dependent on the trust value of that event and the action of the given sensor. It can be computed as follows:

$$Sat(S_j, E_k) = \begin{cases} |\delta(E_k)| & \text{if } (T(E_k) \geq T_\sigma \wedge S_j \subseteq G_s(E_k)) \\ & \vee (T(E_k) < T_\sigma \wedge S_j \not\subseteq G_s(E_k)) \\ -|\delta(E_k)| & \text{otherwise} \end{cases} \quad (9)$$

Here, T_σ is the threshold which decides whether an event is trustworthy or not. $G_s(E_k)$ is the set of all sensors which have issued a notification for the event E_k . Therefore, if an event is trustworthy, and its notification has been issued by the sensor, the reputation of that sensor will increase. The opposite is true for when the event is considered not true. When the TD of a requirement, domain or the entire CIDS falls below a certain level, then a new configuration could be found which increases it once again.

Experiments and Results. The authors experiment with a simulation that includes 500 sensors, 20 domains and 10 requirements. The initial reputation of each sensor is assigned a random value. The first experiment assesses events in a domain with higher TD compared to a domain with lower TD. The authors use an optimization algorithm to search for a placement with high TD. They simulated over 2000 events and assessed the trust levels of these events (see Eq. 7). The value of agreement level is between 1 and -1 . A value close to 1 means that the sensors are in agreement that an event is true while a value close to -1 means that the sensors agree that an event is bogus. For domains with lower TD, the agreement level for honest events and bogus events were 0.3139 and -0.3102 respectively. For the domains with higher TD, the agreement level for the honest and bogus events were 0.7253 and -0.7098 respectively, a significant improvement. Another experiment was to test the effect TD has on the resilience of the system to malicious sensors. The authors incrementally increase

the number of malicious sensors in the system and observe the effect this has on the trust values of the malicious sensors. With higher TD, these trust values decrease more rapidly than in domains with lower TD. This can give a clear indication that the system is being compromised by malicious sensors.

3.3 A Trust-Aware P2P-Based CIDS

Duma et al. [1] proposed a trust-aware Peer-to-Peer (P2P)-based CIDS. The proposed system consists of a trust-aware correlation engine and an adaptive TM scheme. The correlation engine is used to filter data sent by untrustworthy peers. The TM scheme works by using past experience to decide if peers are trustworthy or not. The sensors form a P2P network, in which they are interconnected and communicate to detect and prevent attacks.

Trust Management. To calculate trust among sensors, each peer has a list of peers that it trusts, and checks for peer trustworthiness before taking any decision regarding a possible threat. To adjust the trustworthiness of a peer, the local peer will evaluate any event according to if it was an incident or not, and adjust its trust regarding other peers. In more details, each peer P_i has a list of acquaintance peers, which consists of other peers that P_i has interacted with and their trust value. For each peer P_j which is present in the acquaintance list, P_i keeps two variables: the first one is s_{ij} which represents the number of successful experiences that i had with j. The second one is u_{ij} which represents the number of unsuccessful experiences that i had with j. Having these, peer i computes the trustworthiness of peer j as: $t_{ij} = w_s \frac{s_{ij} - u_{ij}}{s_{ij} + u_{ij}}$. The w_s parameter, is called significance weight and depends on the total number of experiences that are available for the computations regarding trust. If the number of experiences available are too less, then a peer's trustworthiness cannot be computed by this formula. That means that if the total number of experiences $s_{ij} + u_{ij}$ is below a certain minimum number n , then $w_s = (s_{ij} + u_{ij})/n$, otherwise $w_s = 1$.

A *trust threshold* also exists, which is a minimum value of trust that the peers in a list need to have, so that their warnings are taken into consideration. Peers that are below the threshold are marked with a probation flag and have a certain probation period to pass the threshold. If the peer manages to pass the threshold in time, the flag is removed. If not, the peer is removed from the acquaintance list and some new randomly chosen peer will take its place. The new peer will also be flagged and given a probation period to pass the threshold. If it does not, then the aforesaid procedure will take place. This means that for every peer P_j in the list, P_i has a probation flag pf_{ij} that shows if P_j is flagged or not, and a probation time pt_{ij} that shows the time passed since P_j was flagged.

To ensure that the acquaintance list is dynamically built (and managed), making sure that only trustworthy peers remain in the list, four different cases can be distinguished:

- If an attack occurred and P_j sent a warning then $s_{ij} = s_{ij} + 1$, $u_{ij} = u_{ij}$ and $pt_{ij} = pt_{ij} + 1$.

- If an attack took place and P_j did not send a warning then $s_{ij} = s_{ij}$, $u_{ij} = u_{ij} + 1$ and $pt_{ij} = pt_{ij} + 1$.
- If no attack occurred but P_j sent a warning then $s_{ij} = s_{ij}$, $u_{ij} = u_{ij} + 1$ and $pt_{ij} = pt_{ij} + 1$.
- If no attack took place and P_j did not send any warning then s_{ij} , u_{ij} and pt_{ij} remain the same.

Alert Correlation. A peer can utilize the knowledge of the trustworthiness of others to perform alert correlation. The confidence level of a correlated alert is computed as follows:

$$C_i = w_{dir} \cdot c_i + w_{ind} \cdot \frac{1}{N} \sum_{j=1}^N c_j \cdot t_{ij} \quad (10)$$

Here, c_i is the confidence in the correlated alert as correlated by P_i , c_j is the confidence of the alert received from peer P_j , and N is the number of peers that have not been flagged and have sent alerts used in the correlation. w_{dir} and w_{ind} are the direct and indirect weights (direct for locally generated alerts and indirect for received alerts). Regarding N it is required to be above a certain threshold N_{min} , and if it is lower, then the weight w_{ind} is decreased by N/N_{min} . Hence, dependence on only a low number of peers is avoided. However, even with a high N_{min} problems might also appear, when the number of truthful warnings available for correlation is very small. In the end, if the confidence of a certain correlated alert is above a certain threshold, the peer will activate the incidence response module which will take passive or active action towards the threat.

Experiments and Results. The authors conducted experiments for a virtual network (consisting of 36 clients) that was being attacked by a worm. The clients were grouped in 6 sub-networks. A survival rate was defined, as the number of nodes resisting the worm divided by the number of all nodes in the network. The survival rates for the case when the clients were part of the CIDS and when they were not, were compared. The results showed a significant increase of the average survival rate when the CIDS was utilized. According to the experiments, the more peers are in the CIDS the higher the probability is that the worm will be detected. On the one hand, the survival rate decreases by increasing the number of trusted peers needed for correlation (N_{min}). On the other hand, the resilience of the network increases with N_{min} as the impact of a malicious peers is diminished (which also means that the false alarm rate decreases). Thus, by configuring N_{min} one of the following can be achieved: either a faster detection system with a higher survival rate but prone to false alarms, or a more robust system with a lower level of false alarms but which (due to lack of trust) could miss some of the real alarms.

3.4 A Reputation-Based Bootstrapping Mechanism for CIDSs

Perez et al. [8] proposed a reputation management system that addresses the newcomer (bootstrapping) problem in CIDSs. Bootstrapping is a common issue in P2P networks where newcomers join the networks for the first time. Similarly, reputation bootstrapping is a common issue in reputation systems.

System Model. The CIDS is divided into *security domains* (D_1, D_2, \dots, D_n), each of which defines a *Collaborative Intrusion Detection Network* (CIDN) [2]. Each security domain is composed of a multitude of IDSs, one of which is chosen to be its leader (*Domain Leader* (DL)) acting as the representative of the CIDN in the CIDS. The DL's purpose is to share alerts detected by its CIDN with the CIDNs of other domains, and request recommendations from other CIDNs about a newcomer to compute its initial reputation (trust) score [7]. The newcomers can be *static* IDSs (permanently placed), *mobile* IDSs belonging to users who want to collaborate with security domains, or a *security domain* that wants to improve its accuracy in detecting distributed threats by exchanging alert data. Trust is computed on the basis of the initial absence of historical data and on the fact that IDSs join and leave the system regularly.

Reputation Management System. This section describes the reputation management system focusing on the aforesaid three possible newcomers. Note that the model also depends on the *detection skills*, that is the usefulness and willingness of a newcomer, as well as the similarity between two domains. The usefulness function of a newcomer's detection unit (DU_m) is denoted as $\Phi_x(DU_m)$ and its computation can be found in [8]. Similarly, the computations regarding the willingness of the new detection unit which is expressed as $\omega_x(DU_m)$, and the similarity between two domains (D_x and D_y) which is denoted as $\lambda(D_x, D_y)$ can also be found in [8].

Recommendations from the CIDS about a newcomer. Whenever a newcomer (NC) (either a mobile IDS or a security domain), joins a domain D_x , the latter can query other trusted domains within the CIDS asking for recommendations regarding the newcomer. The aim is to find the most reputable path leading to the most trustworthy domain having recommendations about the NC 's behavior in sharing alerts. The best trust path (T_{tp_i}) built up to the domain D_x , which maximizes the confidence that D_x can have on the most trustworthy domain D_y (which will return its recommendation $Rec_y(NC)$), is computed as:

$$T_{tp_i}(D_x, D_y) = \frac{1}{|tp_i|} \cdot \sum_{D_f, D_k \in tp_i} \frac{T(D_j, D_k) \cdot \left(\frac{1}{\Delta t + 1}\right) \cdot \delta(D_j, D_k)}{|tp_i|} \quad (11)$$

and

$$\delta(D_j, D_k) = \begin{cases} 1, & \text{if } D_j, D_k \in AD_z \\ \epsilon_j \in [0, 1], & \text{otherwise} \end{cases}$$

where (D_j, D_k) represents each consecutive pair of domains in the trust path tp_i , $|tp_i|$ is the length of such a trust path built up to D_x and Δt expresses the amount of time elapsed since the last interaction between D_j and D_k . Essentially, this computes a weighted average of the direct trust values of each subsequent pair of domains, $T(D_j, D_k)$, along trust path tp_i . If NC is a *mobile* IDS joining the domain D_x , $mIDS_i$, this D_x can also query other mobile IDSs that are currently collaborating with D_x ($mIDS_j$ s), about their recommendations on $mIDS_i$. Thus, the final recommendation for $mIDS_i$ is:

$$Rec_{mIDS}(D_x, mIDS_i) = \sum_{mIDS_j \in D_x} \frac{Rec_{mIDS_j}(mIDS_i) \cdot (\frac{1}{\Delta t + 1})}{|mIDS_j \in D_x|} \quad (12)$$

Moreover, for computing the confidence that D_x has, on a recommendation gathered from the CIDS, the following formula can be used:

$$T_{mIDS}(D_x, mIDS_i) = \sum_{mIDS_j \in D_x} \frac{Rep_{D_x}(mIDS_j)}{|mIDS_j \in D_x|} \quad (13)$$

Finally, if it is required to compute a single recommendation score by taking into consideration all the recommendations gathered from those sources that maintain behavioral-based information about the newcomer, then the following equation is used:

$$Rec_{CIDS, mIDS}(D_x, NC) = \theta_{tp_i} \cdot Rec_{CIDS}(NC) + \theta_{mIDS} \cdot Rec_{mIDS}(D_x, NC) \quad (14)$$

where θ_{tp_i} and θ_{mIDS} are the trust that D_x has on the domains providing the NC's recommendation score and those provided by the mobile IDSs currently collaborating with D_x . The equations computing them can both be found in the original paper [8]. The proposal for bootstrapping the reputation of a newcomer in a CIDN will be presented bellow, by distinguishing three cases: the reputation bootstrapping model for a *static* IDS, a *mobile* IDS, or a *new security domain*.

Static IDS: The proposed equation to compute the initial reputation score of a newcomer static IDS, namely $sIDS_i$, when it joins the domain D_x at time t is:

$$Rep_{D_x}^{(t)}(sIDS_i) = (\frac{1}{\Delta t + 1}) \cdot Rep_{D_x}^{\Delta t}(sIDS_i) + (\frac{\Delta t}{\Delta t + 1}) \cdot \Phi_{D_x}(sIDS_i)^{\tau(sIDS_i)} \quad (15)$$

where Δt represents the time elapsed since the last time $sIDS_i$ participated in D_x , $Rep_{D_x}^{\Delta t}(sIDS_i)$ indicates the last $sIDS_i$'s reputation score that D_x has stored, and $\Phi_{D_x}(sIDS_i)$ is the usefulness of $sIDS_i$ from the perspective of D_x .

Mobile IDS: The proposed equation to compute the initial reputation score of a newcomer mobile IDS, namely $mIDS_i$, when it joins the domain D_x at time t is (a formal definition for f'_m can be found in [8]):

$$Rep_{D_x}^{(t)}(mIDS_i) = (\frac{1}{\Delta t + 1}) \cdot Rep_{D_x}^{\Delta t}(mIDS_i) + (\frac{\Delta t}{\Delta t + 1}) \cdot f'_m(\Phi, \omega, \tau, Rec_{CIDS, mIDS}) \quad (16)$$

Security Domain: The proposed equation to compute the initial reputation score of a newcomer security domain D_y , that wishes to collaborate with the domain D_x at time t is (a formal definition for f'_d can be found in [8]):

$$Rep_{D_x}^{(t)}(D_y) = \left(\frac{1}{\Delta t + 1}\right) \cdot Rep_{D_x}^{\Delta t}(D_y) + \left(\frac{\Delta t}{\Delta t + 1}\right) \cdot f'_d(\lambda, \tau, Rec_{CIDS}) \quad (17)$$

Experiments and Results. The authors firstly examined the benefits of including mobile IDSs in the system. Their results suggest an improvement of the detection capabilities required by a CIDN to detect distributed threats. Further testing showed that the reputation bootstrapping model can support around 20% of malicious mobile IDSs before being discarded as valuable detection units. In addition, Perez et al. analyzed the variance of the reputation scores of static and mobile IDSs over time with regard to compromise and misbehavior. It was found that reputation scores are rapidly decremented when there are less than 5% of malicious IDSs. This finding was interesting for mobile IDSs, as they follow a similar pattern with static IDSs although mobile IDSs reputation is computed in each movement across the domains. This accuracy is due to the use of recommendations provided by other trusted parties of the CIDS. Further testing showed that this reputation bootstrapping model maintains its robustness for up to around 20% of malicious IDSs without losing its detection accuracy.

4 Discussion

This section begins with a comparison overview of the surveyed approaches with respect to fulfillment of the requirements discussed in Sect. 2. This comparison is also summarized in Table 1. Subsequently, we discuss the lessons learned from the current status of the state of the art and future directions that will advance the intersection of trust and CIDSs.

Table 1. Comparison of surveyed systems with the proposed requirements

| Requirement | Dirichlet-based approach [4] | Trust diversity [9] | Trust-aware CIDS [1] | Bootstrapping approach [8] |
|---------------------|------------------------------|---------------------|----------------------|----------------------------|
| Global view | ✗ | ✓ | ✗ | ✗ |
| Overhead | ●●○ | ●●● | ●○○ | ●●● |
| Incentive | ✓ | ✗ | ✗ | ✗ |
| Initial trust | ✗ | ✗ | ✗ | ✓ |
| Forgetting factor | ✓ | ✓ | ✗ | ✓ |
| Performance history | ✓ | ✓ | ✓ | ✓ |

4.1 Comparison

In the following we discuss each requirement and how the different approaches fulfill it or not.

- **Global view:** The first [4] and third [1] surveyed systems do not require global knowledge as each node computes its peers' trust values independently. In the second approach [9], a global view is required as sensors are moved between domains to increase trust diversity. This implies that an administrator who can control where the sensors are deployed is required. For the last approach [8], the entire system is divided into security domains. For each domain, the domain leader must have knowledge of the topology and behavior of the nodes in its domains. For this reason, only partial view of the network is required by domain leaders.
- **Minimum overhead:** The first approach [4] utilizes test messages, which increase the communication overhead. In fact, the confidence of the trust value is dependent to the number of the sent test messages. The second approach [9] also depends on the sensors being assigned trust values according to their past behavior which also requires the dispatching of test messages. Moreover, the overhead of re-configuring sensors to increase trust diversity also has to be considered. The third approach [1] has lower overhead than the previous ones as it does not require test messages for the computation of trust values. Instead it only uses the knowledge gained from past interactions to calculate trust values. The last approach [8] has significant overhead, due to the number of steps a newcomer has to take when joining the CIDS.
- **Incentive mechanism:** In the first approach [4], an incentive mechanism is proposed where nodes give more feedback to trustworthy peers while not giving as much to untrustworthy peers. This incentive mechanism is important as it ensures that peers which behave in a trustworthy manner are rewarded while untrustworthy peers are ignored. This also reduces overhead communication and computation overhead as peers do not spend time responding to untrustworthy peers. The rest of the approaches, however, do not make use of any incentive mechanism.
- **Initial trust:** The first system [4] assigns the newcomers with random trust values. Nevertheless, the CIDS mitigates the risk by placing newcomer nodes in a probation list for a certain period of time. The second approach [9] also assigns completely random values and takes no steps to mitigate the risk associated with this scheme. The third system [1] keeps newcomers under probation for a fixed period of time. The newcomer is not assigned an initial trust value. After the probation time period has elapsed, if the newcomer's trust value is above a certain threshold, it is considered trustworthy. The last approach [8] exhibits the most sophisticated approach for assigning trust values to newcomers. It takes into account many factors and gathers background information about nodes to solve the newcomer problem.
- **Forgetting factor:** The first [4], the second [9] and the fourth [8] approaches, all make use of forgetting factors in the calculation of trust, so that more

recent messages are given more weight. However, the third approach [1], does not make use of such a technique.

- **Performance history:** All of the surveyed approaches make use of historical data, in different ways, during the trust calculations.

4.2 Challenges and Steps Ahead

The analysis of the state of the art suggests a plethora of novel ways for managing trust. As we have already described in the previous section, each of the aforesaid mechanisms introduces various advantages. Combining the benefits of each approach is one way towards the fulfillment of the requirements. In addition, we argue that a common methodology for the evaluation of trust in CIDSs is required. A basis for this can be the requirements proposed in Sect. 2.

Furthermore, we envision the following research questions for which we argue that, when answered properly, can improve the output quality of CIDSs:

- Which additional *parameters*, inside the alert data, can be utilized for the computation of trust?
- Is it possible to include more *social* attributes/parameters inside the overall trust calculations? For example, it might be that two organizations, inside the CIDS, have some special long-term connection that cannot easily depicted formally.
- How important is the *timeliness* of the exchanged alert data? It can be that some sensors in a CIDS do not publish their alerts immediately due to internal security policies or due to the need for anonymization of the data.
- How important is the *relevance* of the received alert data for a sensor? Can it be that a sensor receives valid data from a highly trustworthy sensor which however are irrelevant? For instance, what about an organization which obtains information about a port-specific attack, which however is completely banned from that particular organization's network.
- How can *uncertainty* be included in the trust model? Would it be of benefit to include *certainty* scores for the trust values? For instance, an approach used in [5] can consider the volume of data utilized for generating a trust score; when sufficient data is not available, the certainty score would be low.

5 Conclusion

With the continuous growth in both the numbers and the sophistication of cyber-attacks, CIDSs are becoming increasingly important. Introducing computational trust techniques in the field of CIDSs can provide substantial benefits for the detection of insider attacks as well as for the creation of highly tailored threat awareness of the monitored network. We proposed requirements for TM in the context of CIDSs. Moreover, we analyzed the most prominent systems with a focus on how they calculate and manage trust in such a context. The paper also provides an overall discussion of the requirements, with respect to their fulfillment in the related work, and highlights the research challenges and gaps that need to be tackled in the future.

Acknowledgments. This work has received funding from the European Union’s Horizon 2020 Research and Innovation Program, PROTECTIVE, under Grant Agreement No 700071.

References

1. Duma, C., Karresand, M., Shahmehri, N., Caronni, G.: A trust-aware, P2P-based overlay for intrusion detection. In: 17th International Workshop on Database and Expert Systems Applications, DEXA 2006, September 2006
2. Fung, C., Zhang, J., Aib, I., Boutaba, R.: Trust management and admission control for host-based collaborative intrusion detection. *J. Netw. Syst. Manag.* **19**, 257–277 (2011)
3. Fung, C.: Collaborative intrusion detection networks and insider attacks. *J. Wireless Mob. Netw. Ubiquit. Comput. Dependable Appl.* **2**(1), 63–74 (2011)
4. Fung, C.J., Zhang, J., Aib, I., Boutaba, R.: Dirichlet-based trust management for effective collaborative intrusion detection networks. *IEEE Trans. Netw. Serv. Manag.* **8**(2), 79–91 (2011)
5. Habib, S.M., Volk, F., Hauke, S., Mühlhäuser, M.: Computational trust methods for security quantification in the cloud ecosystem. In: *The Cloud Security Ecosystem - Technical, Legal, Business and Management Issues*, pp. 463–493. Syngress (2015)
6. Mitchell, R., Chen, I.R.: A survey of intrusion detection techniques for cyber-physical systems. *ACM Comput. Surv. (CSUR)* **46**(4), 55 (2014)
7. Ortega, F.J., Troyano, J.A., Cruz, F.L., Vallejo, C.G., Enríquez, F.: Propagation of trust and distrust for the detection of trolls in a social network. *Comput. Netw.* **56**(12), 2884–2895 (2012)
8. Pérez, M.G., Mármol, F.G., Pérez, G.M., Skarmeta Gómez, A.F.: Building a reputation-based bootstrapping mechanism for newcomers in collaborative alert systems. *J. Comput. Syst. Sci.* **80**, 571–590 (2014)
9. Pérez, M.G., Tapiador, J.E., Clark, J.A., Pérez, G.M., Skarmeta Gómez, A.F.: Trustworthy placements: Improving quality and resilience in collaborative attack detection. *Comput. Netw.* **58**, 70–86 (2014)
10. Vasilomanolakis, E., Karuppayah, S., Mühlhäuser, M., Fischer, M.: Taxonomy and survey of collaborative intrusion detection. *ACM Comput. Surv.* **47**(4), 33 (2015)
11. Yu, B., Singh, M.: Detecting deception in reputation management. In: *Proceedings of the Second International Joint Conference on Autonomous Agents and Multiagent Systems* (2003)

Self-trust, Self-efficacy and Digital Learning

Natasha Dwyer^{1(✉)} and Stephen Marsh^{2(✉)}

¹ Victoria University, Melbourne, Australia
natasha.dwyer@vu.edu.au

² University of Ontario Institute of Technology, Oshawa, Canada
stephen.marsh@uoit.ca

Abstract. Self-trust is overlooked in trust research. However, self-trust is crucial to a learner's success in a digital learning space. In this paper, we review self-trust and the notion of self-efficacy used by the education researchers. We claim self-efficacy is self-trust. We then explore what self-trust and its expression means to one group of learners and use this data to provide design suggestions for digital learning spaces that improve students' self-trust.

Keywords: Self-trust · Self-efficacy · Trust enablement · Digital learning

1 Introduction

Self-trust, informed confidence that one can accomplish a specific task, is key to the success of learners. In this paper, we explore how self-trust can be enabled in the design of a digital learning environment. We start by exploring the nature of self-trust and self-efficacy. We then turn to consider how students express their self-trust and point of view in a digital learning environment. The paper closes with suggestions for designing digital learning spaces that improve students' self-trust.

2 Background Research

The notion of trusting oneself has been neglected by the digital trust research community. When considering the social aspects of trust, the focus is usually on the notion of trust from one party to another, from an individual to a network of people or to a technological system/device. There are some, however, who consider self-trust. Hardin [1] identifies self-trust with the question 'What can I depend on myself to do'? Dasgupta [2] briefly mentions whether one can trust one selves and outlines some of the safeguards society puts in place to protect those who cannot trust themselves. Abdul Rahman and Hailes [1] refer to 'basic trust' and a "pervasive attitude toward oneself and the world". There is an emphasis on one's disposition to trust as a basis for interaction with others, rather the one's trust in oneself.

According to Gibbs [3], self-trust is trust in one's own ability to make decisions on one's own terms with the understanding that one's judgment is valid. This definition builds on Cofta's [4] definition of trust as a relationship within which a trustor is confident that another party (the trustee), to whom a trustor is in a position of

vulnerability, will respond in the trustor's interests, according to, which has traction in the trust research area. Self-trust is the *informed* confidence one has in oneself. If an individual is lacking in self-confidence then he/she is excessively vulnerable. On the other hand, too much self-confidence means that an individual may not comprehend risk appropriately [5]. Just like trust in others, trust in ourselves can be misplaced. Self-trust has a social component; there needs to be some form of validation with others in order for individuals to calibrate their self-efficacy [6] and self-trust does not preclude trust in others [7]. It is necessary for one to understand one's strengths, competencies and beliefs [7]. An important component of trust is that it is intuitive [8]. Individuals are usually highly effective at managing trust in the context of their everyday life and we see in our study that our participants can articulate what they need to express their self-trust and engage others to trust them.

According to Bandura [9], self-efficacy is a 'belief and confidence' that one has to accomplish certain sorts of work, such as the planning and completing of tasks. Self-efficacy is shaped by previous accomplishment, social influence and an individual's sense of agency [7]. Martinez-Maldonado et al. [10] add that experience plays an important role in one's perception of oneself as well as knowledge. We claim that self-efficacy is intrinsic part of self-trust.

Self-trust/efficacy is a result that universities hope they enable for their graduates, as indicated by the 'graduate outcomes' universities set for themselves. High self-trust and efficacy allows university students to not only make crucial decisions and set life goals but to reach them [11]. Trust level to oneself with first-year students sharply raises in connection with their ability not only to set the vital life goals but also to reach them. If the design of an online system improves a student's self-efficacy then a student is more likely to report higher satisfaction with the system [10]. If an individual has high self-efficacy, then this is a predictor of the individual completing a MOOC (a massive open online course) [12]. Perhaps a role for traditional university education is to build individuals' self-trust and efficacy so that they are in a position to complete endeavours such as MOOCs.

3 Methodology

Trust is a notoriously challenging concept to study and the notion of self-trust is arguably more personal and thus difficult to access. However, researchers need to look at what users actually do in real life contexts [13]. Sometimes asking participants to define a concept like trust does not gather in-depth responses as the task is hard work for participants. It is difficult for participants to understand what the researcher means [14]. For instance, it is likely if a survey about game design asks "What is trust in this context?", the question is likely to be avoided by participants. Instead, gathering data in a more indirect fashion has the potential to be successful.

We gathered data from a brief survey for students embedded in a classroom activity, to understand what self-trust means for students in the context of e-learning. These students were undertaking a postgraduate unit 'Analyzing the Web and Social Networks' at Victoria University, Australia. The class has 42 students, 19 males and 23 females.

Students are required to undertake an oral presentation at the end of semester to communicate their findings to their peers. Peer judgment and acceptance, as part of an industry information sharing exercise, were central to the exercise. The task itself of creating a presentation can foster the development of creative self-efficacy because the process develops confidence with tools, developing ideas, presenting arguments to others and responding to feedback [15].

Two options were given to students about the delivery of their findings: presenting in person during class time or submitting a video presentation of their performance. In our short survey students were asked to explain why they made their choice. Our participants were asked “Why did you choose to present in person or create a video (circle the one you chose then quickly tell us why)”.

4 Analysis

As Roghanizad [8] argues, individuals are usually highly effective at understanding the dynamics of trust in their everyday lives and our participants gave us a clear explanation of their choices around self-trust and technology. The participant’s responses reflect the notion of ‘functional advantage’ (outlined in technology acceptance models see [16]). Users of technology are not inherently loyal to one form of technology or mode of interaction, the decision depends upon what is on offer. Users are aware of the possibilities technology offers and want to use what works better for their particular context [16]. In the responses provided by the participants, we see them weigh up how technology can help or hinder them express their self-trust and also gain the trust of their peers. Jervis [17] says simply that the reason why different users have different preferences is because people come from wide ranging experiences, bearing different personalities and opinions.

Some participants chose to present in class because they believe it is easier to control the presentation:

I prefer to present in person because I can get a better sense of what the room is finding most interesting - can emphasise or skim over as required. Can also modify and monitor my own energy as appropriate. Basically I feel as though I have more control of the presentation.

Trust and control are counterparts [18], one can compensate for absence of the other. As Knight [19] states autonomy is safeguarded when students are given “control over the right things at the right time.” On the other hand, other participants thought that video-mediated presentation would present them in the most confident light. Confidence, as Cofta [4] argues, is a key component of trust. If a person can make others confident of their abilities, then that person is trusted. A participant in our study said:

I prefer to make the presentation on video rather than in person because I feel more confident. I’m a shy and introverted person and speaking in front of an audience or in public, it is a bit uncomfortable and I feel stage fright.

Another added that the asynchronous nature of a video presentation changes the type of judgments that are made because the audience is not forced to watch you:

I chose this subject because it really interests me, but I am just a ‘beginner’ in this scenario, and given the high level and experience some students have in class, it was really difficult for me to come up with an interesting subject to present for them, so I thought the video was a good idea, as it gives the freedom to watch it only to those who may be interested in the topic.

Other participants raised the issue of authenticity, a concept very close to the notion of trust. Those looking to trust, automatically synthesise the evidence to trust they are presented with, filtering for authenticity and assessing the credibility of the information [20]. Assessing the credibility of the information, the participants realized they can control how authentic, trustworthy and believable they can appear to be.

Personally, I find it more authentic to present in person, in front of a real audience. Even more so, that I believe they (the audience) could take something out of my outcomes.

I like talking in front of people because I like interactivity and dialog. It feels more real than just a youtube video. If there’s real (live) singing vs. lip sync, I would choose live singing.

It is interesting to note, that even though the individuals in our study are regarded as ‘digital natives’, some of our participants prefer ‘real life’ over digitally mediated interactions. Some participants were interested in in-class presentations because there is a more personal level of interaction that allows instant feedback and flow, echoing Luhmann’s [21] well-known theory on the links between growing familiarity and trust. The more time and interactions individuals have with each other, the more likely that trust will develop between people.

I would like more interaction with the students. I feel like I am talking to the air if doing my presentation in a video. Presentation in class makes me feel more energy.

Easy to contribute to the conversation in case someone has a question or needs clarification.

I am more comfortable presenting in person as its more interactive, helping in getting reactions, input feedback on your presentation.

As indicated by the responses above, many participants who chose to present in class raised the issue of being able to ask questions. The ability to query promotes trust as. asking for clarification shows a need to understand. Answering queries breeds understanding and engagement [22]. Our participants understand this dynamic:

So that my fellow classmates could see and listen to my presentation topic, also gives people a chance to ask me questions about my topic.

Video is not interactive, no questions allowed. It’s a better ‘sound check’ format, i.e.: live reaction.

Another participant added that the topic of a presentation itself should play a role when deciding to present in person or using technology:

It needs to be in person because I’m sharing from my own experience.

And finally, practical considerations also play a role in the choices students make, which are issues designers of online spaces should not overlook:

The clarity of voice and image is guaranteed in a classroom presentation.

We have decided to team up for the presentation but one of us is not available on the day/time of the actual presentation. It is difficult to coordinate time together. Eliminating the date of the actual presentation in the mix gives more freedom in scheduling the recording session. We are both shy and filming the presentation took out some of the anxiety of presenting in class.

In the responses from participants, there are suggestions to improve the design of digital learning spaces so that they enable self-trust and trust. For instance, our participants tell us that an element of control is required in an online environment if the space is to allow them to establish trust with their peers. Many of our participants choose to present themselves in class, in person, rather than using digital tools to present their ideas, so this may mean that they find the current offerings within digital learning spaces fail to meet their needs.

The ability to query is rated as an important element in a trust interaction, according to our participants. The current means to ask questions in online spaces does not seem to be satisfactory for our participants and they seek alternatives. Students also wish to develop familiarity with each other and also demonstrate their authenticity to each other, which are long-term design challenges for the creators of online spaces.

Some of the design features we suggest above for trust enablement are ‘grand challenges’. Currently there are teams of designers who are working on these long-term goals. In the short-term, we propose the use of a questionnaire to be implemented as part of the digital learning experience. The aim of the questionnaire is to help students use technology so that it suits their preferences. Once submitted, the questionnaire could give students automated suggestions about the advantages and disadvantages of different technology use, guide them through their choices and provide examples of the choices students like them have made in the past.

5 Conclusion

Self-trust is a key attribute for learners in digital learning spaces. Education researchers use a similar concept, self-efficacy, which we claim is self-trust. In this paper, we explore the choices one group of participants make to express their self-trust in digital learning spaces. The data gathered suggests ways to improve the design of digital learning environments so that they enable self-trust. Our participants tells us that an element of control can enable self-trust. A means to display authenticity can also assist, as can the facility to ask and answer questions. Some of these design features are ‘grand challenges’ for the creators of digital learning spaces. As a solution for current environments, we suggest the implementation of a questionnaire which can guide students, based on their preferences, towards modes of interacting that enable self-trust.

References

1. Abdul-Rahman, A., Hailes, S.: Supporting trust in virtual communities. In: Proceedings of the 33rd Annual Hawaii International Conference on System Sciences, vol. 1, p. 9. IEEE (2000)
2. Dasgupta, P.: Trust as a commodity. In: *Trust: Making and Breaking Cooperative Relations*, vol. 4, pp. 49–72 (2000)
3. Gibbs, P.: Competence or trust: the academic offering. *Qual. High. Educ.* **4**, 7–15 (1998)
4. Cofta, P.: *Trust, Complexity and Control: Confidence in a Convergent World*. John Wiley & Sons, Hoboken (2007)
5. Nooteboom, B.: Social capital, institutions and trust. *Rev. Soc. Econ.* **65**, 29–53 (2007)
6. Nys, T.: Autonomy, trust, and respect. *J. Med. Philos.* **41**, 10–24 (2016)
7. Govier, T.: Self-trust, autonomy, and self-esteem. *Hypatia* **8**, 99–120 (1993)
8. Roghanizad, M.M., Neufeld, D.J.: Intuition, risk, and the formation of online trust. *Comput. Hum. Behav.* **50**, 489–498 (2015)
9. Bandura, A.: Self-efficacy: toward a unifying theory of behavioral change. *Psychol. Rev.* **84**, 191 (1977)
10. Martinez-Maldonado, R., Anderson, T., Shum, S.B., Knight, S.: Towards supporting awareness for content curation: the case of food literacy and behavioural change (2016)
11. Krishchenko, E., Shevyreva, E., Tushnova, Y.: Subjectivity formation in the system of higher education. In: СБОРНИКИ КОНФЕРЕНЦИЙ НИЦ СОЦИОСФЕРА, pp. 109–111. Vedecko vydavateľske centrum Sociosfera-CZ sro (Praha) (2016)
12. Zhu, M., Bergner, Y., Zhang, Y., Baker, R., Wang, Y., Paquette, L.: Longitudinal engagement, performance, and social connectivity: a MOOC case study using exponential random graph models. In: *Proceedings of the Sixth International Conference on Learning Analytics & Knowledge*, pp. 223–230. ACM (2016)
13. Schilke, O., Cook, K.S.: Sources of alliance partner trustworthiness: integrating calculative and relational perspectives. *Strateg. Manag. J.* **36**, 276–297 (2015)
14. Dwyer, N.: *Traces of digital trust: an interactive design perspective*. Victoria University (2011)
15. Martin, C., Nacu, D., Pinkard, N.: Revealing opportunities for 21st century learning: an approach to interpreting user trace log data. *J. Learn. Analytics* **3**, 2 (2016)
16. Chang, Y.-Z., Ko, C.-Y., Hsiao, C.-J., Chen, R.-J., Yu, C.-W., Cheng, Y.-W., Chang, T.-F., Chao, C.-M.: Understanding the determinants of implementing telehealth systems: a combined model of the theory of planned behavior and the technology acceptance model. *J. Appl. Sci.* **15**, 277 (2015)
17. Jervis, R.: *Perception and Misperception in International Politics*. Princeton University Press, Princeton (2015)
18. Bijlsma-Frankema, K., Costa, A.C.: Understanding the trust-control nexus. *Int. Sociol.* **20**, 259–282 (2005)
19. Knight, S., Anderson, T.D.: Action-oriented, accountable, and inter (Active) learning analytics for learners. In: *The 6th International Learning Analytics and Knowledge Conference*, pp. 47–51 (2016)
20. Gambetta, D., Bacharach, M.: *Trust in Signs*. *Trust and Society*, pp. 148–184. Russell Sage Foundation, New York (2001)
21. Luhmann, N.: Familiarity, confidence, trust: problems and alternatives. In: *Trust: Making and Breaking Cooperative Relations*, vol. 6, pp. 94–107 (2000)
22. Marsh, S., Dwyer, N., Basu, A., Storer, T., Renaud, K., El-Khatib, K., Esfandiari, B., Noël, S., Bicakci, M.V.: Foreground trust as a security paradigm: turning users into strong links. In: *Information Security in Diverse Computing Environments*, pp. 8–23. IGI Global (2014)

Trust Metrics

Advanced Flow Models for Computing the Reputation of Internet Domains

Hussien Othman¹, Ehud Gudes¹, and Nurit Gal-Oz²(✉)

¹ Ben-Gurion University, Beer-Sheva 84105, Israel
{hussien,ehud}@cs.bgu.ac.il

² Sapir Academic College, D.N Hof Ashkelon, 79165 Ashkelon, Israel
galoz@sapir.ac.il

Abstract. The Domain Name System (DNS) is an essential component of the Internet infrastructure that translates domain names into IP addresses. Recent incidents verify the enormous damage of malicious activities utilizing DNS such as bots that use DNS to locate their command & control servers. We believe that a domain that is related to malicious domains is more likely to be malicious as well and therefore detecting malicious domains using the DNS network topology is a key challenge.

In this work we improve the flow model presented by Mishsky et al. [12] for computing the reputation of domains. This flow model is applied on a graph of domains and IPs and propagates their reputation scores through the edges that connect them to express the impact of malicious domains on related domains. We propose the use of clustering to guide the flow of reputation in the graph and examine two different clustering methods to identify groups of domains and IPs that are strongly related. The flow algorithms use these groups to emphasize the influence of nodes within the same cluster on each other. We evaluate the algorithms using a large database received from a commercial company. The experimental evaluation of our work have shown the expected improvement over previous work [12] in detecting malicious domains.

1 Introduction

Domain reputation has become an essential tool in fighting advanced malware and Advanced Persistent Threats (APT). Since detecting sophisticated malware in real-time is difficult, the use of domain reputation is a key objective for companies like Dambella [8] or Cyren [7] which provide services for identifying threats imposed by malicious domains and IPs.

The common approach to assess domains is to compute features from DNS records and queries responses, and use these features to train a classifier that labels domains as malicious or benign. This approach is effective as long as the attackers do not manipulate these features. However, DNS features are not robust [16], since the attackers can change the features of malicious domains to evade detection. For example, they can change Time To Live (TTL) for DNS queries, patterns in domain names, etc.

The Notos model for assigning reputation to domains [2] was the first to use statistical features in the DNS topology data and to apply machine learning methods to construct a reputation prediction classifier. FluxBuster [14], is a passive DNS traffic analysis tool for detecting malicious flux networks. Using DNS mapping data, Perdisci et al. [14] apply an hierarchical clustering on the domains where the distance between two domains is calculated according to the number of mutual resolved IPs, so that each of these clusters represents a candidate flux network. This work motivated our clustering based approach.

In this paper we extend the flow model presented by Mishsky et al. [12] for calculating the reputation scores of domains. Flow models are used in trust based reputation systems such as EigenTrust [11] and Pagerank ([13]). The assumption underlying the work of Mishsky et al. [12] is that IPs and domains which are neighbors of malware-generating IPs and domains, are more likely to become malware-generating as well. The outline of their approach is as follows. Domains and IPs are represented as nodes in a directed weighted graph. The nodes in the graph are initially assigned a reputation value based on two lists of labeled domains: a list of ‘bad’ domains which is compiled from various internet databases (e.g., VirusTotal [19]) and a list of ‘good’ domains based on the Alexa database [1]. A major challenge in building the graph is the assignment of weights to edges to reflect the strength of relation between the nodes they connect. Their approach uses the statistical features which are associated with mappings of domains and IPs. Starting with this graph, a flow algorithm is applied to propagate the initial reputation scores from each node to its neighbors in an iterative manner so that the impact of malicious nodes on their neighbors is presented. The evaluation of this model [12] demonstrates its ability to identify malicious domains and the experimental results on real-life data are quite impressive.

The main contribution of the current work is the improvement of previous work [12] in two steps. We first extend the graph by using new attributes which are related to the registration information of domains and name servers hosting them. Next we propose the clustering approach to strengthen weights on edges between domains and IPs that seem to be highly correlated. We examine two clustering methods, Categorical and Communities. Categorical clustering groups domains based on their mutual attributes, while Communities clustering groups domains and IPs based on their mutual relations.

The rest of this paper is organized as follows: in Sect. 2 we present the related work and discuss briefly the work of [12] which is our starting point. In Sects. 3 and 4 we present our contribution to improve this work, starting with the new attributes we propose; we explain our clustering methods and the respective flow algorithms. We demonstrate the results of our experimental evaluation in Sect. 5, and in Sect. 6 we conclude and discuss future work.

2 Related Work

There are quite a few papers which use DNS data logs to detect Botnets and malicious domains.

Most of the papers use the DNS traffic behavior, rather than mapping information. Some recent papers such as Notos [2] and FluxBuster [14] use the mapping information in order to compute a reputation score for malicious domains. However, both require a huge amount of mappings between IPs and Domains in order to succeed. For example, in [14] false positives and true negatives in the experiments are explained by the fact that some domains are not mapped to enough IPs. Villamarin-Salomon et al. [18] provide C&C (Command and Control) detection technique motivated by the fact that bots typically initiate contact with C&C servers to poll for instructions. For each domain, they aggregate the number of non-existent domains (NXDOMAIN) responses per hour (denoted NX-during-hour- i) and also compute the query rate of each second level domain (SLD) per hour. They use the average and standard variation of these rates as features to detect abnormally high or temporally concentrated rates. Based on these features, a vector of a certain domain is classified as anomalous using the Mahalanobis Distance metric [9]. The suspects produced by this system were also independently reported as suspicious by other detectors. The Exposure system [3] collects data from DNS answers returned from authoritative DNS servers and uses a set of 15 features that are divided into four feature types: time-based features, DNS answer-based features, TTL value-based features, and domain name-based features. The above features are used to construct a classifier based on the J48 decision tree algorithm [22] in order to determine whether a domain name is malicious or not.

Choi et al. [5] use passively monitored DNS traffic to detect botnets, which form a group activity in similar DNS queries simultaneously. They assume that infected hosts (bots) perform DNS queries at the following occasions: successful host infection, malicious attack DDoS attack, Spam mailing, C&C server link failures (repetitive attempts), server migration (to communicate new site location) and IP address changes. Using this data they construct a feature vector and perform clustering to identify the malicious domains. Their proposed technique can detect botnets irrespective of their communication protocols and C&C server migration however it faces the problem of obtaining DNS traffic data (compared to the use of DNS topological data which is much easier to obtain). Another recent paper by Pedrici et al. [15] uses traffic data and behavior based traffic from which a traffic graph is built and analyzed.

2.1 Starting Point: A Topology Based Flow Model

Mishsky et al. [12] address the problem of detecting unknown malicious domains by estimating their reputation score. A classical flow algorithm for propagating trust is applied on a DNS topology graph database, for computing reputation of domains and thus discovering new suspicious domains. This model uses DNS IP-Domain mappings and statistical information but does not use DNS traffic data as done by others (e.g., [2]). The motivation for using a flow algorithm is the hypothesis that IPs and domains which are neighbors of malware-generating IPs and domains, are more likely to become malware-generating as well. In [12] a graph is constructed to reflect the topology of domains and IPs and their

relationships and a flow model is applied to propagate the knowledge received in the form of black-list, to label domains in the graph as suspected domains. Domains and IPs are represented as nodes in a directed weighted graph while the edges of the graph describe their network connections. The two types of vertices in the topology graph, domains and IPs, derive four types of edges between them: Domain-Domain, IP-IP, Domain-IP and IP-Domain. Two domains are connected in the graph if they have in common a mutual parent. Two IPs are connected if they have the same ASN, BGP, registrar, and date attributes. A Domain is connected to an IP (or IP to Domain), if the IP was resolved for that domain according to A-Records (a database of successful mappings between IPs and domains). To construct real-life graphs data collected from an ISP and from WHOIS database [21] is used. The graph nodes are initially labeled based on a list of malicious domains which is compiled from various Internet databases (e.g., VirusTotal [19]). Weights are assigned to the graph edges based on statistical features which are associated with the domains and IP mappings. The flow algorithm applied on this graph, propagates reputation from one node to another while discounting the strength of the connection between them as expressed by the weight on the edge connecting them. The results of the flow algorithm, which is similar to Eigentrust [11] are used to indicate the extent to which nodes are suspected to be malicious.

3 DNS Topology Graph Extension

To construct the DNS topology graph we use additional attributes to those used by Mishsky et al. [12], that better determine the significance of the connections between the nodes. The new features are based on the assumption that malicious domains reuse valuable resources. Most popular attacks depend on the availability of many resources which are often purchased [20]. For example: domain names are registered or transferred for a price, large numbers of infected hosts are available for rent, bullet-proof servers are available for rent [20], etc. Moreover, many types of resources are made to be reusable so that they can be resold multiple times to maximize financial gain. The reuse of resources across different attacks may reveal connections between malicious domains.

In this work we investigate the use of three types of resource attributes:

Domain's parent: A k – Top level domain ($kTLD$) is the k suffix of the domain name. Following [2] for each domain we define k parents as $iTLD$ for $i = 1..k$. For example for domain `finance.msn.com`, the 3TLD is the same as the domain name `finance.msn.com`, the 2TLD is `.msn.com` and the 1TLD is `.com`. This attribute was used by [12]. In our experiments we restrict the parents to 2TLD.

Registration: Registrant name, Address, Email, Organization and Registrar. These attributes are important since the WHOIS information [21] about a malicious domain sometimes includes certain pseudo-identity details such as the same/similar fake registrant name, the same registrant email, same registrant address, etc [20].

NameServer: The motivation for using name server relation is that one name server can provide the DNS records for a large number of malicious domains.

Similar to [12], we define a weight function that assigns a weight to each edge in the graph. There are two types of vertices in the topology graph: domains and IPs, deriving four types of edges between them. Let Set_{IP} denotes the set of all the IPs and let Set_{domain} denotes the set of all the domains.

Let $Attributes_{domains}$ be the set of attributes we use in the graph to define the relation between domains. $Attributes_{domains} =$ (registrant name, registrant city, registrant country, registrant email provider, registrant organization, registrar, name server, parent).

Let Set_{α} be the set of all domains which have in common attribute α where $\alpha \in Attributes_{domains}$.

Let w be a weight function $w : (v, u) \rightarrow [0, 1]$ used to assign weight to the edge (u, v) where $u \in Set_{IP}$ or $u \in Set_{domain}$ and $v \in Set_{IP}$ or $v \in Set_{domain}$.

For edges of type *domain-ip*, *ip-ip*, *ip-domain* we use the same formula as in [12].

- Weights on *domain-ip* edges: For $d \in Set_{domain}$ and a list of A-records, let I_d be all the IPs that were mapped to d . For each $ip \in I_d$ we define:

$$w(d, ip) = \frac{1}{|I_d|}.$$
- Weights on *ip-domain* edges: For $ip \in Set_{IP}$ and the list of A-records, let D_{ip} be all the domains mapped to ip . For each $d \in D_{ip}$ we define:

$$w(ip, d) = \frac{1}{|D_{ip}|}.$$
- Weights on *ip-ip* edges: Let *commonAtt* be a combination of ASN, BGP, registrar, date attributes. Let $Set_{commonAtt}$ be the set of all IPs with the attribute combination *commonAtt*. For each $ip_1, ip_2 \in Set_{commonAtt}$ s.t. $ip_1 \neq ip_2$ we define:

$$w(ip_1, ip_2) = \frac{1}{|Set_{commonAtt}|}.$$
- Weights on *Domain-Domain* edges: here we use the new attributes defined above. Let Set_{α} be the set of all domains with the same attribute α as domain d . For each $d_1, d_2 \in Set_{\alpha}$ s.t. $d_1 \neq d_2$ we define the following weight metric:

$$w_{\alpha}(d_1, d_2) = \frac{1}{|Set_{\alpha}|}.$$
 We define $w(d_1, d_2)$ as average of all the w_{α} for $\alpha \in Attributes_{domains}$.

4 Clustering Based Flow Model

The new attributes we add to distill the significance of the connections between the nodes, improve the method of [12], but not to the extent expected (see Sect. 5.3). A possible reason for this is the relatively low flow received by nodes of low centrality according to [12]. To overcome this problem we use clustering to classify domains. By clustering we identify groups of domains that are related to each other and direct the flow accordingly.

We use two types of clustering, *Categorical Clustering* which involves domain nodes only, and *Communities Clustering* which involves both domains and IPs. We refer to the latter as communities. For each type of clustering we have a Flow model that uses only relevant edges.

4.1 Categorical Clustering

Categorical clustering identifies groups of strongly connected domains where the strength of connection is determined by the number of attributes they share with each other. We define v_d , the vector of attributes for domain d as follows:

$v_d = (\text{Registrar, Registrant name, Registrant email provider, Registrant city, Registrant country, Registrant organization, Name Server, Parent})$.

In the categorical algorithm, we use K-Modes [10] that extends the k-means paradigm to cluster categorical data. Following the K-Modes algorithm we estimate the strength of connection between any two domains, by calculating the dissimilarity distance between their attribute vectors.

The **clustering based flow Algorithm**, described in Algorithm 1, is different from the flow algorithm presented in Mishsky et al. [12]. In Algorithm 1 the edges are restricted to two types: IP to IP edges and Domain to IP or IP to Domain edges. The edges between domains are removed, and the propagation of flow from domain to another domain is done by dividing the average score in the cluster to all the domains it includes. In each iteration, we propagate the flow between IPs and Domains. Only at the end of the iteration we propagate flow within each cluster of domains.

Algorithm 1 starts with the initialization phase in lines 23–30 in which the score of each domain is set to the average score of all the domains in its cluster. These scores are passed as parameter to the algorithm as $V_{initial}$. In each iteration of the flow (lines 6–19), we propagate scores from domain to IP and IP to domain (lines 6–12) and from IP to IP (lines 13–19).

At the end of each iteration (line 18), we compute the new score of each cluster and propagate the increment gained in the iteration to all the domains in the cluster. The flow algorithm is executed separately by Algorithm 2 to propagate benign reputation and malicious reputation. It calls Algorithm 1 with an initial set of domains that are labeled either malicious or benign.

Applying the algorithm separately to propagate malicious and benign reputation may result in a node that is labeled both benign and malicious. The final labeling of such node depends on the relative importance given to each label as done in Algorithm 2. The parameters of this algorithm are two vectors of domains' reputation scores, one with their reputation as benign and the other with their reputation as malicious. The initial score of a domain as malicious is set in the malicious vector to 1 if it appears in the black lists (obtained by various Internet databases e.g., VirusTotal [19]) and 0 otherwise. The reputation score of domains in the benign vector is set the same way using the benign domains as obtained from Alexa [1]. In line 6 the scores are combined, and using a threshold of θ we label the domains as malicious.

4.2 Communities of IPs and Domains

In contrast to clustering of domains only, the community based clustering uses the entire generated graph including IPs, Domains and the weighted edges. Our goal is to examine whether malicious domains and IPs are grouped together in

Algorithm 1. Flow with clustering

```

1: procedure CLUSTERBASEDFLOW(Vector  $V_{initial}$ , Iterations  $n$ )
2:   for  $C \in Clusters$  do
3:     InitializeScores( $C, V_{initial}$ )
4:   end for
5:   for 1 to  $n$  do
6:     for  $e = (d, ip) \in Set_{IP-Domain-Edges} \vee e = (ip, d) \in Set_{Domain-IP-Edges}$  do
7:        $prevScore_d = score_d$ 
8:        $score_d += weight_e * score_{ip}$ 
9:        $score_{ip} += weight_e * prevScore_d$ 
10:       $C = GetCluster(d)$ 
11:       $C.increment += score_d - prevScore_d$ 
12:    end for
13:    for  $e = (ip_1, ip_2) \in Set_{IP-IP-Edges}$  do
14:       $prevScore_{ip_1} = score_{ip_1}$ 
15:       $score_{ip_1} += weight_e * score_{ip_2}$ 
16:       $score_{ip_2} += weight_e * prevScore_{ip_1}$ 
17:    end for
18:    for  $C \in Clusters$  do
19:      propagateScore( $C$ )
20:    end for
21:  end for
22: end procedure
23: procedure INITIALIZESCORES(Cluster  $C$ , Vector  $V_{initial}$ )
24:    $totalScore = 0$ 
25:   for  $d \in C$  do
26:      $totalScore = totalScore + V_{initial}(d)$ 
27:   end for
28:   for  $d \in C$  do
29:      $score_d = V_{initial}(d) + \frac{totalScore}{C.size}$ 
30:   end for
31: end procedure
32: procedure PROPAGATESCORE(CLUSTER  $C$ )
33:   for  $d \in C$  do
34:      $score_d = score_d + \frac{C.increment}{C.size}$ 
35:   end for
36: end procedure
37: procedure GETCLUSTER(DOMAIN  $d$ )
38:   for  $C \in Clusters$  do
39:     if  $d \in C$  then
40:       return  $C$ 
41:     end if
42:   end for
43: end procedure

```

clusters we call communities. The community detection algorithm we use follows the Louvain Method for community detection [4]. In our graph, domains and IPs are connected to each other via multiple relation measures. Domains are connected to each other according to their parent domain, registrant information and name server serving them. IPs are connected by ASN, registrar,

Algorithm 2. Combine benign and malicious flow (n, V_{benign}, V_{mal})

Input

n : number of iterations, V_{benign} : a vector of benign domains, V_{mal} : a vector of malicious domains

```

1:  $V_{benign} \leftarrow ClusterBasedFlow(V_{benign}, n)$ 
2:  $V_{mal} \leftarrow ClusterBasedFlow(V_{mal}, n)$ 
3:  $Set_{Mal} \leftarrow \emptyset$ 
4: for  $d \in Set_{Domain}$  do
5:   if  $V_{mal}[d] + W_{benign} \cdot V_{benign}[d] > \theta$  then  $Set_{Mal} \leftarrow Set_{Mal} \cup \{d\}$ 
6:   end if
7: end for
8: return  $Set_{Mal}$ 

```

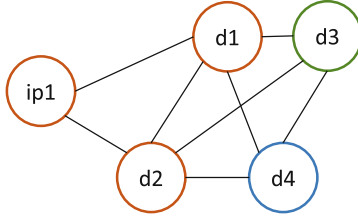


Fig. 1. Community graph example: the score propagated from d_1 to d_3 and d_4 is weaker than the score propagated from d_1 to d_2 which share the same IP ip_1 .

BGP prefix, date. IPs and Domains are connected by A-records mapping. By clustering we divide our graph to communities where each community contains IPs and domains that have the strongest connections to each other. To avoid false positives, we build a new weighting function for the flow algorithm based on communities. For example in Fig. 1 all the domains have in common a mutual parent, registrant information and name server, hence we add edges between each pair. However, d_1 and d_2 have in common a mutual IP, ip_1 . Assuming d_1 is malicious, in the previous method, the domains d_2, d_3, d_4 will get equally bad score from d_1 that will be propagated to their neighbors. Using communities we may have d_1, d_2 in the same cluster due to their connection to ip_1 while d_3, d_4 remain out of the cluster, hence they will receive a smaller bad score from d_1 . This way we refine the propagation and make a more careful distinction.

The community clustering begins with a graph of IPs and Domains as nodes and the weighted edges connecting them. We use the weights defined in Sect. 3 with one exception: for IP-Domain and Domain-IP edges, we multiply the weight by a factor of α , $\alpha > 1$. The aim of this factor is to strengthen the connection between IP and Domain so it will not be discarded by the Louvain algorithm. Since in our graph there are significantly greater number of edges between domains than edges between IP and Domain, domains are more likely to be added to a community that contains Domains with same mutual attributes and discard the IP connection. We prevent this by adding the α factor to these edges. Once we have clustered the graph to communities we use the community-based flow algorithm to compute the reputation of each domain.

The **Communities-based flow algorithm**, uses two different flow models, within a community and between different communities. These models differ in the weighting scheme they use:

- *Flow inside a community*: The weight of each edge (u, v) where $u \in C_i$ and $v \in C_i$ is calculated using the same weight function as defined in Sect. 3 considering only the induced subgraph constructed by all the nodes of C_i .
- *Flow across communities*: we allow flow between communities. Hence, we could not treat each community as a separated graph. Rather, we treat it all as one big graph and apply a new weighting scheme. The new weighting scheme applies to any type of edge (IP-IP, IP-Domian, Domian-IP, Domain-Domain). Let v denote a vertex in the graph. An *inside edge* (u, v) where $u \in C_i$ and $v \in C_i$ is an edge connecting two vertices inside the community. An *outside edge* (u, v) where $u \in C_i$ and $v \notin C_i$ is an edge connecting a vertex inside the community with a vertex outside the community. Let n_1 denote the number of inside edges connecting v . Let n_2 denote the number of outside edges connecting v . For each inside edge we assign the weight: $\frac{1}{n_1 x}$.

For each outside edge we assign the weight: $\frac{1}{(n_1 + n_2)y}$.

For each vertex v , the sum of all the weights on the edges connected to it is 1, therefore: $\frac{n_1}{n_1 x} + \frac{n_2}{(n_1 + n_2)y} = 1$.

Let $f = \frac{y}{x}$, the weight of each inside edge:

$$\frac{(n_1 + n_2)f}{(f + 1)n_2 n_1 + f n_1^2} \quad (1)$$

The weight of each outside edge:

$$\frac{1}{(f + 1)n_2 + f n_1} \quad (2)$$

The ratio between the weights is

$$\frac{\textit{inside}}{\textit{outside}} = \frac{f(n_1 + n_2)}{n_1} \quad (3)$$

If n_2 is much greater than n_1 (most of the edges are outside edges) the weight inside is much greater than outside. If n_1 is much greater than n_2 (most of the edges are inside edges), the weights are almost equal (where f is close to 1). We therefore use f to tune the ratio between weights inside and outside a community, and examine this parameter in our experiments.

5 Evaluation

The evaluation of our methods uses real data collected from several sources. We first describe our data and the criteria obtained for evaluating the results.

5.1 Data Sources

We use the following five sources of data to construct the graph.

- A-records: a database of successful mappings between IPs and domains, collected by Cyren [7] from a large ISP over several months. This data consists of over one million domains and IPs which are used to construct the graph nodes.
- Feed-framework: a list of malicious domains collected and analyzed by Cyren over the same period of time as the collected A-records. This list is intersected with the domains that appeared in the A-records and serves as the initial known malicious domains vector.
- WHOIS [21]: a query and response protocol that is widely used for querying databases that store the registered users or assigners of an Internet resource, such as a domain name, an IP address block, or an autonomous system. We use WHOIS to get the IP data, which consists of the five characteristics of IP (ASN, BGP prefix, registrar, country, registration date) and to get the registrant information on the domains.
- Virustotal [19]: a website that provides scanning of domains for viruses and other malware. It uses information from 52 different anti-virus products and provides the time that a malware domain was detected by one of them.
- Alexa [1]: Alexa database ranks websites based on a combined measure of page views and distinct site users. Alexa lists the “top websites” based on this data averaged over a three-months period. We use the set of top domains as our initial benign domains, intersecting it with the domains in the A-records.

Table 1 presents the size and the characteristics of the data we use to construct our graph.

After constructing the graph each experiment is conducted twice, once with initial malicious domains and once with initial benign domains. The score of each domain is computed (those who did not receive any flow remain with score zero). The scores are sorted and compared to Virus Total scans.

Table 1. Data characteristics

| | | |
|----------|---------------------------------------|-----------|
| Vertices | Number of domains | 1007833 |
| | Number of IPs | 345451 |
| | Number of Malicious domains | 462 |
| Edges | Number of edges from IP To Domain | 1116823 |
| | Number of edges from Domain To Domain | 119055774 |
| | Number of edges from IP To IP | 29260535 |

5.2 Evaluation Criteria

In the real world one would prefer to minimize the risk and get a list of suspicious domains even if only a small part of them is malicious. In many cases the cost of checking suspicious benign domains is worth the risk of getting attacked by a real malware. Our aim is to identify suspicious domains to reduce the amount of unnecessary checks.

We run Algorithm 1 with the lists of malicious domains and with the list of benign domains separately and obtain two vectors: $Vector_{mal}$ - representing the malicious reputation of domains and $Vector_{benign}$ - representing the benign reputation of domains. The flow algorithm assigns reputation scores to all domains. Using $W_{benign} \in [-1, 0]$, to discount the benign measure, the score of a domain d is computed according to Algorithm 2 for combining benign and malicious flow:

$$score[d] = Vector_{mal}[d] + W_{benign} \cdot Vector_{benign}[d] \quad (4)$$

We sort the domains by descending reputation score and use the score of the domain on the k -th position as the threshold, denoted θ . The group of domains with score higher than the threshold are selected, denoted $HSet_k$: $HSet_k = \{d \in Domains | score[d] \geq \theta\}$.

We compare this set to the ground truth which is available from VirusTotal [19] to obtain the domains in $HSet_k$ that are correctly tagged as malicious denoted $GTSet_k$. Let $MalSet$ denote the set of malicious domains according to VirusTotal [19], then $GTSet_k = \{d \in HSet_k | d \in MalSet\}$. The evaluation criteria is the ratio of domains that were correctly identified as malicious out of the set $HSet_k$. The $TRatio$ criteria representing the precision of positive prediction, is calculated as follows:

$$TRatio = \frac{|GTSet_k|}{|k|} \quad (5)$$

A domain tested with VirusTotal [19], is considered as tagged only if it was tagged by at least two anti-virus programs. A similar approach was used by Cohen et al. [6] to compare the precision of a list of suspicious accounts returned by a Spam detector against a randomly generated list.

5.3 Experiment Results

In this section we analyze the results of the proposed clustering methods and compare them to the results of the original method [12] we attempt to improve. For categorical clustering we use the k-modes clustering algorithm to construct clusters of domains which have common attributes. We evaluate the results with respect to the number of clusters generated by the k-modes algorithm as shown in Fig. 2.

When the number of clusters (k) is between 500 and 1000 we identify the largest number of malicious domains. As the number of clusters increases and the average cluster size decreases accordingly, less malicious domains are identified since less flow enters the cluster from outside. On the other hand, smaller values

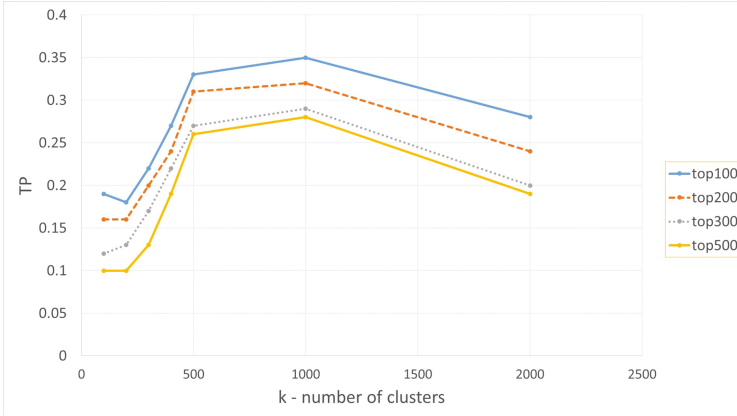


Fig. 2. Categorical Clustering: precision with respect to the number of clusters

of k which result in larger clusters introduce too much noise and too many false positives. We have verified against VirusTotal [19] about 35% of the top 100 scores and 28% of the top 500 scores.

We compare the results of the categorical clustering to the original methods of [12] by testing all algorithms with the data described in Sect. 5.1. Table 2 presents the improvement of the methods proposed in this paper in two steps. First by introducing the new attributes, and second by using both attributes and clustering. However it is important to note that our dataset is smaller and less populated than the one used in [12].

Table 2. Comparison of the results with the original method [12]

| Method | Top100 | Top200 | Top300 | Top500 |
|--|--------|--------|--------|--------|
| [12] | 10% | 10% | 9.6% | 9.2% |
| [12] + New attributes | 20% | 17.5% | 16.6% | 16.2% |
| [12] + New attributes + Categorical clustering ($k = 1000$) | 35% | 32% | 29% | 28% |

In the second test, we divide our graph into communities using the Louvain algorithm as described in Sect. 4, where each community expresses better correlation between its domains and IPs. To prevent the Louvain algorithm from discarding the connections between Domains and IPs, we use the α factor as described in Sect. 4. The best results for our graph were obtained with $\alpha = 100$. Figure 4 shows the impact of each common attribute on the community clustering. The figure shows that domains with common attributes do not necessarily belong to the same community. For example, on the average only 0.18 of pairs of domains that have the same Registrar attribute belong to the same community.

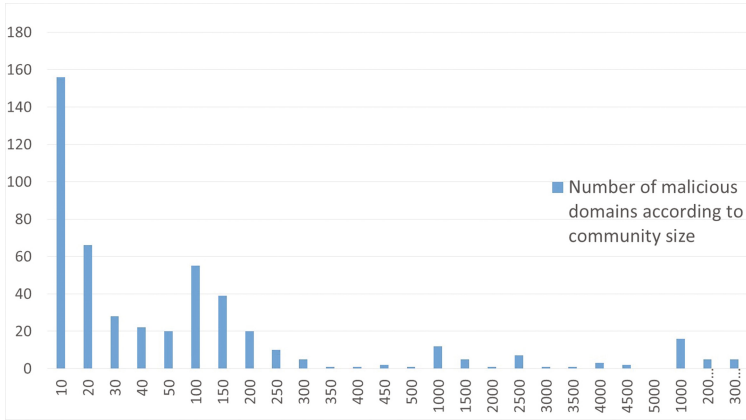


Fig. 3. Distribution of malicious domains in communities with respect to community size

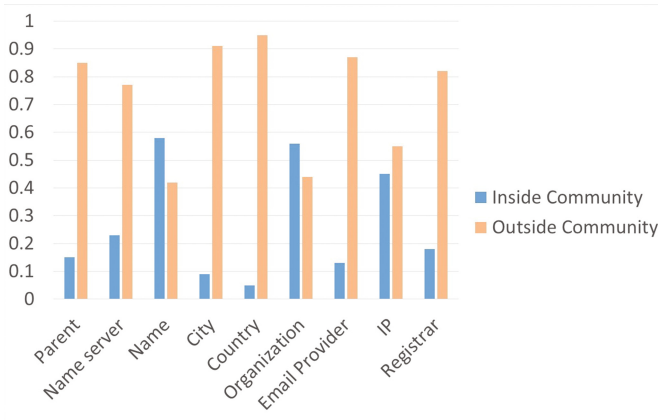


Fig. 4. Average distribution of mutual attribute inside and outside the community.

Only 0.45 of the IPs belong to the community of the domains to which they are mapped.

Figure 3 depicts the size of communities in which malicious domains reside. For example, about 160 malicious domains reside in communities of size smaller than 10.

Flow inside communities: We set the weights on the edges as described in Sect. 4 and run the flow algorithm. Table 3 depicts the percentage of malicious domains detected out of the top rated. The results present a significant improvement over previous experiments. We have identified 68 malicious domains among the top 100 scores, and 101 malicious domains among the top 200 scores. Furthermore, 51 out of 65 top rated malicious domains were identified which represent

Table 3. Flow inside communities results

| Top100 | Top200 | Top300 | Top500 |
|--------|--------|--------|--------|
| 68% | 50.5% | 33.6% | 20.2% |

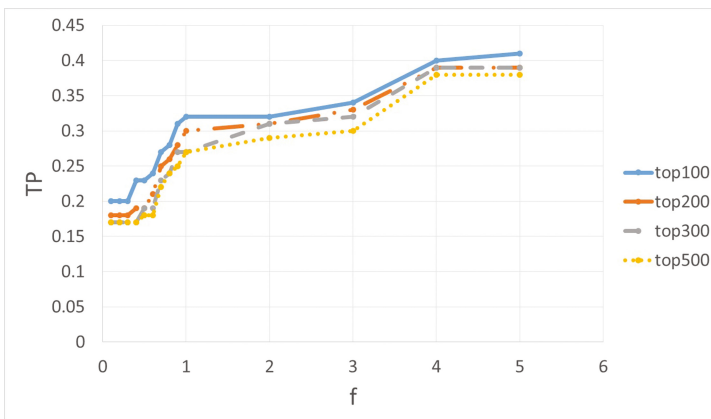
Table 4. Flow across communities results

| Top100 | Top200 | Top300 | Top500 |
|--------|--------|--------|--------|
| 42% | 39% | 39% | 38% |

about 78% detection. Table 3 also shows that when considering more than the 200 top rated malicious domains, the precision decreases and less domains are detected. The reason is the restriction of flow inside the communities only. There is no propagation of malicious score from malicious domains to communities which don't contain malicious domains initially. Therefore even if malicious domains reside in these communities, they are not detected by the flow algorithm. Roughly only about 5% of the graph could be reached via flow inside communities which contain malicious domains.

Flow across communities: In this experiment we allow flow between communities so that a domain's score is affected by both, domains inside its community and related domains that reside outside its community. The best results are shown in Table 4. The results of the flow inside communities are better for the top 100 and 200 scores but for top 300 scores and further the flow across communities performs much better. This demonstrates the limitation of the flow inside communities, which is unable to find the malicious domains identified by the second method.

Figure 5 depicts the results obtained with respect to the f parameter defined in Sect. 4 for tuning the ratio between weights of inside-edges and outside-edges. We show the results of the experiments where f varies between 0.1 and 5. The best results are obtained for higher values of f in this range, where the weights inside communities are increased and outside communities are decreased.

**Fig. 5.** Flow across communities: precision with respect to f parameter

However, in experiments with larger values of f where the relative importance of outside edges becomes very small, the behavior is similar to the behavior of flow inside communities only.

From a security point of view, the flow across community can be used as a means to identify more malicious domains at the price of reduced precision.

6 Conclusions and Future Work

Computing accurately domain reputation is an important factor in preventing the spread of malware by malicious domains. The use of Trust flow for computing domain reputation was first reported in [12]. In this paper we improve the results of [12] using new attributes and a clustering based flow model. We have presented two types of clustering. The first type, categorical clustering, involves clustering of domains only and is based on multiple attributes. The improvement demonstrated by this algorithm was minor. The second type, communities clustering, involves clustering of both domains and IPs by using the concept of Communities. This clustering further improves the results. The best results in the top 100 scores were achieved from flow inside communities only (close to 80%), in which we restricted the flow to domains and IPs residing in the same community. However, in that model we didn't get enough "bad" scores to evaluate the top 500 scores. The other communities based model yield quite good results with respect to top 100 and top 500 scores, while identifying almost 40% malicious domains in the top 500 scores. This is an improvement over [12] which found only 30% malicious domains.

In future work we intend to extend our work in two directions. The first is to combine our flow model with classification models (e.g., [2]) to overcome situations where statistical features are not good enough to distinct between malicious and benign domains (for example, domains that are resolved to small set of IP addresses). The other direction is to integrate the results of profiling methods used for anomaly detection [17], with the flow model. Based on behavioral attributes an anomaly score can be used to identify malicious domains and the relation between reputation and anomaly score can be examined.

Acknowledgement. This research was supported in part by the Lynn and William Frankel Center for Computer Sciences at Ben-Gurion University, Israel, and we like to thank them for their support. We also thank the reviewers for very useful comments.

References

1. Alexa. The web information company (2014). <https://www.alexa.com/>
2. Antonakakis, M., Perdisci, R., Dagon, D., Lee, W., Feamster, N.: Building a dynamic reputation system for DNS. In: USENIX Security Symposium, pp. 273–290 (2010)
3. Bilge, L., Kirda, E., Kruegel, C., Balduzzi, M.: Exposure: finding malicious domains using passive DNS analysis. In: NDSS (2011)

4. Blondel, V.D., Guillaume, J.-L., Lambiotte, R., Lefebvre, E.: Fast unfolding of community hierarchies in large networks. CoRR, abs/0803.0476 (2008)
5. Choi, H., Lee, H.: Identifying botnets by capturing group activities in dns traffic. *Comput. Netw.* **56**(1), 20–33 (2012)
6. Cohen, Y., Gordon, D., Hendler, D.: Early detection of outgoing spammers in large-scale service provider networks. In: Rieck, K., Stewin, P., Seifert, J.-P. (eds.) DIMVA 2013. LNCS, vol. 7967, pp. 83–101. Springer, Heidelberg (2013). doi:[10.1007/978-3-642-39235-1_5](https://doi.org/10.1007/978-3-642-39235-1_5)
7. Cyren. The web information company (2016). <http://www.cyren.com/>
8. Dambella. The web information company (2016). <https://www.damballa.com>
9. De Maesschalck, R., Jouan-Rimbaud, D., Massart, D.L.: The mahalanobis distance. *Chemometr. Intell. Lab. Syst.* **50**(1), 1–18 (2000)
10. Huang, Z.: Extensions to the k-means algorithm for clustering large data sets with categorical values. *Data Min. Knowl. Discov.* **2**(3), 283–304 (1998)
11. Kamvar, S.D., Schlosser, M.T., Garcia-Molina, H.: The Eigentrust algorithm for reputation management in p2p networks. In: Proceedings of the 12th International Conference on World Wide Web, pp. 640–651. ACM (2003)
12. Mishsky, I., Gal-Oz, N., Gudes, E.: A topology based flow model for computing domain reputation. In: Samarati, P. (ed.) DBSec 2015. LNCS, vol. 9149, pp. 277–292. Springer, Cham (2015). doi:[10.1007/978-3-319-20810-7_20](https://doi.org/10.1007/978-3-319-20810-7_20)
13. Page, L., Brin, S., Motwani, R., Winograd, T.: The pagerank citation ranking: Bringing order to the web (1999)
14. Perdisci, R., Corona, I., Giacinto, G.: Early detection of malicious flux networks via large-scale passive DNS traffic analysis. *IEEE Trans. Dependable Sec. Comput.* **9**(5), 714–726 (2012)
15. Rahbarinia, B., Perdisci, R., Antonakakis, M.: Segugio: efficient behavior-based tracking of malware-control domains in large ISP networks. In: 45th Annual IEEE/IFIP International Conference on Dependable Systems and Networks, DSN 2015, Rio de Janeiro, Brazil, 22–25 June 2015, pp. 403–414 (2015)
16. Stinson, E., Mitchell, J.C.: Towards systematic evaluation of the evadability of bot/botnet detection methods. In: Proceedings of the 2nd Conference on USENIX Workshop on Offensive Technologies, WOOT 2008, Berkeley, CA, USA, pp. 5:1–5:9. USENIX Association (2008)
17. Villamarín-Salomón, R., Brustoloni, J.C.: Identifying botnets using anomaly detection techniques applied to DNS traffic. In: 2008 5th IEEE Consumer Communications and Networking Conference, CCNC 2008, pp. 476–481. IEEE (2008)
18. Villamarín-Salomón, R., Brustoloni, J.C.: Bayesian bot detection based on DNS traffic similarity. In: Proceedings of the 2009 ACM Symposium on Applied Computing, pp. 2035–2041. ACM (2009)
19. VirusTotal. A free virus, malware and URL online scanning service (2014). <https://www.virustotal.com/>
20. Xu, W., Sanders, K., Zhang, Y.: We know it before you do: predicting malicious domains. In: Virus Bulletin Conference (2014)
21. Whois. IP data (2014). <https://who.is/>
22. Witten, I.H., Frank, E.: *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, San Francisco (2005)

Trust Trust Me (The Additivity)

Ken Mano^(✉), Hideki Sakurada, and Yasuyuki Tsukada

NTT Communication Science Laboratories, NTT Corporation,
3-1 Morinosato Wakamiya, Atsugi, Kanagawa 243-0198, Japan
{mano.ken,sakurada.hideki,tsukada.yasuyuki}@lab.ntt.co.jp

Abstract. We present a mathematical formulation of a trust metric using a quality and quantity pair. Under a certain assumption, we regard trust as an additive value and define the soundness of a trust computation as not to exceed the total sum. Moreover, we point out the importance of not only soundness of each computed trust but also the stability of the trust computation procedure against changes in trust value assignment. In this setting, we define trust composition operators. We also propose a trust computation protocol and prove its soundness and stability using the operators.

Keywords: Trust · Metric · Protocol · Soundness · Stability · Subjective logic

1 Introduction

We discuss mathematical formulation of a trust metric. There are two classical approaches to such formulation, namely logical [1, 2, 7–9, 11, 13], and computational [3, 4, 10, 12, 14] approaches. The logical approach involves modal logics such as Epistemic logic or Doxastic logic and is aimed at revealing the logical structure of a trust problem. The computational approach introduces operations on a trust metric to compute the required trust values, and involves probability theory, the subjective logic or fuzzy logic to justify the validity of the computed values. Our approach belongs to the latter and has the characteristics that a trust metric is formulated as a quality and quantity pair.

Why quality and quantity? To recall how such problems have been treated in existing research, let us consider the case of the subjective logic.

The subjective logic is a logical system with the set of opinions (b, d, u) as its domain. The elements b , d and u of the tuple represent the proportions of belief, disbelief and uncertainty, respectively. Therefore, it is assumed that $b, d, u \in [0, 1]$ and $b + d + u = 1$. This is not a tailor-made theory for trust, but rather a general system for uncertainty. There are studies that have applied the subjective logic to the computation of trust metrics [5, 6].

A sequential composition of opinions called discounting, denoted by \otimes , is defined as follows. Suppose A 's opinion on the trust concerning B is

(b_{AB}, d_{AB}, u_{AB}) , and B says that his opinion on the trust concerning C is (b_{BC}, d_{BC}, u_{BC}) . Then, the trust of A concerning C is

$$\begin{aligned} & (b_{AB}, d_{AB}, u_{AB}) \otimes (b_{BC}, d_{BC}, u_{BC}) \\ &= (b_{AB} \cdot b_{BC}, b_{AB} \cdot d_{BC}, 1 - b_{AB} \cdot b_{BC} - b_{AB} \cdot d_{BC}). \end{aligned}$$

Since the certainties (belief and disbelief) are defined as a multiplication of values in $[0, 1]$, they decrease unless the case of perfect trust or distrust, and uncertainty increases accordingly. If we interpret b and d as probability, this seems natural. But is it always valid for trust composition?

Let us consider the following story regarding measurement as an analogy. To measure a target C , we must use two measuring instruments A and B sequentially. That is, B directly measures C and makes some output. Then A measures the output of B , and finally makes some output that the observer actually sees.

Then, if the accuracy of A is 12 bits and that of B is 16 bits, the total accuracy is 12 bits. If the accuracy of A is 20 bits and that of B is 16 bits, the total accuracy is 16 bits. The accuracy is regarded as a quantitative metric of the trustworthiness of the results, and their composition is not determined by multiplication but by *min*.

We are seemingly making a similar judgment in the everyday life. For instance, let us consider a situation where A is informed concerning C from B who has been a friend of A for over 10 years. If the information is “I came to know C last year, and he is a fairly good guy.”, then it is a rational option for A to believe the information as it is. On the other hand, if the information is “ C is a friend from childhood, and I entrust him with the management of all my property.”, then A typically will not believe the information as it is. Although the degree to which the value of the information is discounted depends on the person, it is natural to regard the value as quantitatively limited by the length of the friendship between A and B .

Thus we propose using the quantity as an element of the trust metric (without converting it to a proportion) to represent the uncertainty caused by the quantity. We then define the quantity of the sequential composition of two trust values as the *min* of their quantities. The idea of using a quality and quantity pair is not new. For instance, it is used implicitly in [3].

For the trust computation we also need parallel composition. In the subjective logic, parallel composition is called consensus, which is defined based on the quantitative summation of evidence of belief and disbelief. At that time, a supplemental parameter called atomicity is introduced in order to map the opinions to evidences. For trust as a quality and quantity pair, we can define the corresponding composition simply using a quantity-weighted average, without any supplemental parameters. But here we face the problem of evidence independence in the subjective logic.

For instance, suppose that C performed a good action for each of A and B , and they regard the actions as evidences of trust, respectively. Then, in order to combine these evidences using consensus, the actions must be probabilistically independent of each other.

We regard this requirement not always appropriate at least in the context of human trust. This is because there is no general way to decide whether or not two actions performed by a person are independent. Moreover, is independence truly necessary? If we think that a good guy tends to perform a good action, then any two good actions that he performs are somewhat dependent on each other. Should we abandon combining them? Let's return to common sense. People would naturally think as follows: he is really a good guy' cause he did good twice!

This casual sense provides us with a completely new idea for trust computation, namely to regard trust as an additive value. We say a value is additive when the value of the whole system is the total sum of all subsystem values. We do not claim that this is the only solution to this problem. However, we believe this is at least one valid mathematical modeling of trust.

Treating trust adequately as an additive value is nontrivial. Even if the definition of each composition is valid, its application generally may not be valid since the computed value can be invalidly amplified by duplicate counting. To avoid such invalidity, we must clarify the way of determining the basic trust values that each person initially holds, and the way of combining them. We formulate this problem using a kind of ordered algebra where the partial order \preceq represents the amount of information.

This paper is organized as follows. Section 2 explains how our model is applied in reality. In Sect. 3, we describe the basic problem setting and the trust composition operators. In Sect. 4 we define the validity of trust computation (called soundness and stability in this paper) and introduce the syntax of linear terms for representing valid computation. In Sect. 5, we present a protocol for distributed trust computation and prove its validity using the algebraic properties of operators. In Sect. 6, we present a comparison with existing studies, and in Sect. 7 we discuss inherent issues when applying our model. Due to the lack of space, all proofs are omitted.

2 Application

We consider a situation in which trust values are distributed in a network. That is, we assume that people hold their trust values concerning others, and do not assume the existence of a trusted third party. We also assume they may answer correctly, ignore the question, or tell a lie when they are asked about their trust values.

The trust computation presented in this paper is applicable to any network service, e.g., SNS and market place, in which trust or reputation information is needed. For instance, the stars used by Amazon can be regarded as trust information represented by quality and quantity pair, where quality is represented by the proportion of five cases. Moreover, PKI and ad hoc networks are expected to be good applications.

One of the main contributions of our paper as regards such applications is to enable de-centralized management of trust information. Distributed trust

management has some advantages compared to server-centric management. One is that local trust information is easy for the holder to add and/or update, and thus users can obtain more correct and up-to-date information. It is also advantageous that the user can choose a preferred source of trust information. By asking to a person who is reliable and who has the same taste, we can obtain desirable trust information.

3 Trust

3.1 Quality and Quantity of Trust

In this section, we define trust as a pair of quality and quantity. For any person A and person B distinct, a trust t_{AB} of A concerning B is a pair (p_{AB}, q_{AB}) . Here q_{AB} is a non-negative real called the quantity of t_{AB} . Its intended interpretation is the amount of interaction between A and B , for instance, the number of communication messages, and the transaction value. p_{AB} is called the quality of t_{AB} , and we assume $p_{AB} \in [0, 1]$. We also assume $p_{AB} = 0$ when $q_{AB} = 0$, so we often write 0 instead of $(0, 0)$. For instance, let quantity be the number of queries and quality the rate of correct answers. If A sent B 100 queries, and received 90 correct answers in the past, then the trust of A to B is $(0.9, 100)$.

3.2 Composition Operators of Trust

We introduce two types of trust composition operators: parallel and sequential. *Parallel composition \uplus of trust*

Assume someone asked A and B about their trust concerning C , and received $t_{AC} = (p_{AC}, q_{AC})$ and $t_{BC} = (p_{BC}, q_{BC})$, respectively. Then, assuming that A and B are totally reliable, how should the person consolidate these two values? We define the parallel composition \uplus of trust as follows:

$$t_{AC} \uplus t_{BC} = \left(\frac{q_{AC} \cdot p_{AC} + q_{BC} \cdot p_{BC}}{q_{AC} + q_{BC}}, q_{AC} + q_{BC} \right).$$

That is, the composition of quantities is simple addition and that of qualities is a quantity-weighted average. We define $(0, 0) \uplus (0, 0) = (0, 0)$. This definition can be justified by the following analogy: quantity is the number of independent trials, and quality is the success probability. For instance, if $t_{AC} = (0.9, 100)$ and $t_{BC} = (0.8, 1000)$, then $t_{AC} \uplus t_{BC} = (0.81, 1100)$.

The operator \uplus is associative and commutative, and satisfies $0 \uplus t = t$.

Sequential composition $$ of trust*

Suppose that B told A that the trust of B concerning C is $t_{BC} = (p_{BC}, q_{BC})$, and that the trust of A concerning B is $t_{AB} = (p_{AB}, q_{AB})$. Then, to compute the trust concerning C , A should discount t_{BC} by t_{AB} .

We define the sequential composition $*$ of trust as follows:

$$t_{AB} * t_{BC} = (p_{AB} \cdot p_{BC}, \min(q_{AB}, q_{BC})).$$

According to the analogy of probability, the quality of the composition result is the expected value in the case of p_{BC} with probability p_{AB} , and 0 with probability $1 - p_{AB}$. The definition of quantity is based on the idea that A can quantitatively rely on the trust value t_{BC} provided by B at most q_{AB} .

For instance, if $t_{BC} = (0.9, 1000)$ and $t_{AB} = (0.8, 100)$, then $t_{AB} * t_{BC} = (0.72, 100)$. The sequential composition represents an inferiorization of trust by communication. We think that information is degraded by communication since people can tell a lie. A liar can provide either a higher or lower trust value than the truth, but because of the nature of the trust problem, higher is worse. Moreover, since we cannot generally gather all the trust information in a network, the computed value is necessarily quantitatively smaller than the network-wide total value. The above definition reflects these observations.

Roughly speaking, the above definition of $*$ implicitly assumes the following properties of a lie: if the trust of A concerning B is (p_{AB}, q_{AB}) , when B informs A of the trust (p, q) ,

1. p is at most $1/p_{AB}$ -times higher than the truth, and
2. q is not too large when the truth is less than or equal to q_{AB} .

Under such assumptions, the above definition is justified. In Sect. 4.1 we present a generalized form of these assumptions. The operator $*$ is associative and commutative, and satisfies $0 * t = 0$.

Example 1. We do not insist that the above is the only possible definition of compositions. It is simply a running example, and variations are possible depending on the purpose and user preference. The following are examples of such variations.

1. Replacing the parallel composition with

$$t_{AC} \uplus_{max} t_{BC} = (max(p_{AC}, p_{BC}), max(q_{AC}, q_{BC})).$$

2. Replacing the sequential composition with

$$t_{AB} *_2 t_{BC} = (p_{AB} \cdot p_{BC}, min(p_{AB} \cdot q_{AB}, q_{BC})).$$

The former example has little practical significance, but is useful for making it clear that the validity argument in this paper does not depend on probability theory.

The latter example is more significant. In the definition of the quantity of the composition, the first argument of min is replaced with $p_{AB} \cdot q_{AB}$, which implies that a lower p_{AB} yields a smaller quantity. Intuitively, this definition says that information from low quality source is unreliable both qualitatively and quantitatively. It is, however, noticeable that this sequential composition is neither associative nor commutative.

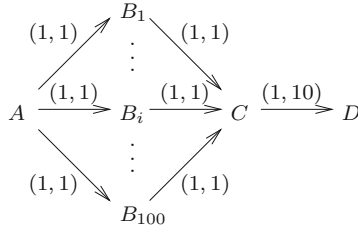


Fig. 1. Duplicate counting

3.3 Problem of Duplicate Counting

In the previous section, we defined composition operators of trust. However, their applications are not always valid. This is closely related to duplicate counting and the additivity of trust. In this section, we present three types of duplications in which the applications of operators are invalid.

Example 2. Assume that trust values among $A, B_1, \dots, B_{100}, C, D$ are defined as $t_{AB_i} = (1, 1)$, $t_{B_i C} = (1, 1)$ and $t_{CD} = (1, 10)$ as shown in Fig. 1. Then, let us consider the following examples of trust calculations:

1. “The trust of A concerning D is $t_{AB_1} * t_{B_1 C} * t_{CD} \uplus \dots \uplus t_{AB_{100}} * t_{B_{100} C} * t_{CD} = (1, 100)$ ”.
2. “The trust of A concerning C is $t_{AB_1} * t_{B_1 C} \uplus \dots \uplus t_{AB_{100}} * t_{B_{100} C} = (1, 100)$ ”.

Are these calculations valid? ■

The problem with the former example is clear, that is, although the quantity is originally 1, it is (or at least seems to be) invalidly amplified to 100 because of the duplicate counting of the trust value t_{CD} . On the other hand, there seems to be no apparent duplication in the latter example, but the problem here is how $t_{B_i C}, \dots, t_{B_{100} C}$ are determined. If such trust values are determined since B_1, \dots, B_{100} observed just one action of C simultaneously, then the total quantity should be 1.

For instance, suppose that there is an NGO with 100 members B_1, \dots, B_{100} . Assume that C made a donation of 100 dollars because he approved of its aim, and that, based on this single fact, each B_i decided to give 100 dollars’ worth of trust concerning C . Then is it valid to *add* the trust values quantitatively and to conclude that C obtained 10000 dollars’ worth of trust?

The two problems are similar but different. The problem of 2 is concerned with how to determine the basic trust values, while the problem of 1 is concerned with how to calculate using the basic values.

Let us first consider the problem of 2. We introduce the distinction between basic trust values and others. A basic trust value (or simply, a basic trust) is a trust value determined by each person based on his direct and exclusive experiences. The other trust values are those computed using communicated information.

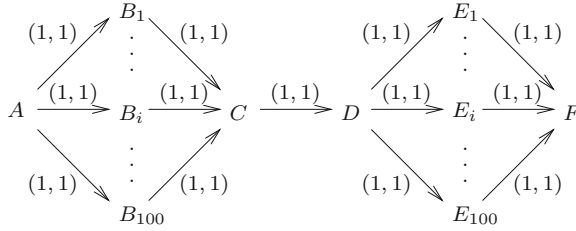


Fig. 2. Duplicate counting of communication pathways

We say that someone’s experience is direct if he sees it with his eyes or hears it with his ears. Hearsay information and conjecture are not direct. We say someone’s experience is exclusive if he is the only one who experienced it. If we must think of several people’s experiences concerning a single event, a share of the quantity is distributed to each person so that the sum is 1, e.g., $1/n$ to n individuals. If the share cannot be determined, such an experience is regarded as not direct. Under this assumption, the addition of quantity in Θ is justified.

We denote the set of all persons by \mathcal{P} , and assume \mathcal{P} is finite. For any $A, B \in \mathcal{P}$ distinct we write t_{AB} to denote the basic trust of A concerning B . We also call t_{AB} a basic trust concerning B , or simply a basic trust. Based on the assumption that a basic trust is determined by direct and exclusive experiences, we regard basic trust as an additive value, and define the total basic trust t_B of B as the total sum of the basic trusts concerning B :

$$t_B = \sum_{P \in \mathcal{P} - \{B\}} t_{PB}.$$

Next, let us consider the problem of 1. In Fig. 1, if t_{CD} is the only non-zero trust concerning D , the result of this example exceeds the total basic trust concerning D because there is duplicate counting of t_{CD} . Such duplication must be avoided for a valid computation of additive values.

Then, what about the duplicate counting of basic trusts not directly concerning D , that is, the trusts on the communication pathways to D , when calculating a computed trust concerning D ?

Example 3. In the situation shown in Fig. 2, consider a calculation that involves, for example, first calculating the following values for 100 paths,

$$A \rightarrow B_i \rightarrow C \rightarrow D \rightarrow E_i \rightarrow F \quad (i = 1, \dots, 100)$$

and then summing them. Is this valid? ■

Note that the paths are chosen so that they share just one C - D edge. In the above example, the calculation result does not exceed the total basic trust concerning F . However, in the summation

$$\sum_{i=1}^{100} t_{AB_i} * t_{B_i C} * t_{CD} * t_{DE_i} * t_{E_i F} = (1, 100),$$

a large amount of trust is divided into 100 parts, which run through the C - D edge with relatively small quantity. Therefore, this violates the basic idea of sequential composition whereby C can quantitatively rely on the trust value provided by D at most the quantity of t_{CD} .

In fact, if t_{CD} is updated to $(0.5, 2)$ by a new experience, the calculation result changes to $(0.5, 100)$. This means that a result with quantity 100 is heavily influenced by a change in quantity 1. Such a situation is contrary to the nature of quantity, and thus should be avoided.

In the next section, we will formulate two properties implying that the above problem does not occur using a binary relation on trusts.

4 Network of Trust

Using the composition operators presented in the previous section, we investigate trust computation by gathering trust values from people on a network.

4.1 Soundness and Stability

In this section, we define the validity of trust computation independent of the specific way of calculation. In the rest of this paper, the quality and quantity of trust t is denoted by $p(t)$ and $q(t)$, respectively.

First, we define a binary relation \preceq on trusts as follows:

$$t \preceq t' \text{ iff } \exists t_1, t_2 \ t \uplus t_1 = t_2 * t'.$$

This definition states that the left-hand side $t \uplus t_1$, of which t is a part, is equal to the right-hand side $t_2 * t'$, which is inferior to t' by t_2 . That is, \preceq means that the left-hand side is partial and inferior to the right-hand side, and thus is regarded as representing the relative amount of information.

For instance, $(0.8, 10) \preceq (0.9, 100)$ clearly holds. Moreover, $(0.8, 100) \preceq (0.9, 100)$ (by letting $t_1 = (0, 0)$ and $t_2 = (8/9, 100)$) and $(0.9, 50) \preceq (0.8, 100)$ (by letting $t_1 = (0.7, 50)$ and $t_2 = (1, 100)$) also hold. On the other hand, $(0.8, 100) \not\preceq (0.9, 10)$ and $(0.9, 90) \not\preceq (0.8, 100)$.

We present basic properties of \preceq . For any trust Δt , $t \preceq_{\Delta t} t'$ iff $q(t) \preceq q(t') \wedge t' \preceq t \uplus \Delta t$.

Lemma 4. \preceq satisfies the following properties.

1. Reflexivity: $t \preceq t$.
2. Transitivity: If $t \preceq t'$ and $t' \preceq t''$, then $t \preceq t''$.
3. Anti-symmetry: If $t \preceq t'$ and $t' \preceq t$, then $t = t'$.
4. Decreasing: $t * t' \preceq t'$.
5. Monotonicity: If $t \preceq t'$, then $t'' \uplus t \preceq t'' \uplus t'$.
6. Semi-distribution: $t * (t' \uplus t'') \preceq t * t' \uplus t * t''$.
7. Overtaking: For any trust Δt , if $t \preceq_{\Delta t} t'$, then $t'' \uplus t \preceq_{\Delta t} t'' \uplus t'$ and $t'' * t \preceq_{\Delta t} t'' * t'$. ■

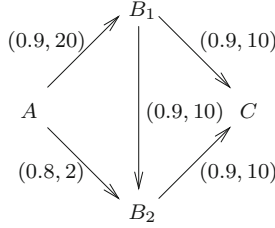


Fig. 3. Sound and unsound trust computation

Intuitively, $t \preceq_{\Delta t} t'$ represents a relation where t is not superior to t' ($q(t) \leq q(t')$), but can overtake it by making Δt progress further than t' ($t' \preceq t \uplus \Delta t$). The overtaking property implies that this relation is preserved by \uplus and $*$.

Remark 5. The monotonicity of $*$ concerning \preceq does not hold in general. When $t \preceq t'$, each $t'' * t \preceq t'' * t'$, $t'' * t \succeq t'' * t'$ and another case (namely where they are incomparable with respect to \preceq) are possible. Of course, whether or not the monotonicity of the sequential composition holds depends on the definition of the composition operators. For instance, let \preceq_{max} be the relation defined using the parallel composition \uplus_{max} of Example 1 and $*$. Then, $*$ is monotonic with respect to \preceq_{max} , and \preceq_{max} is characterized as follows:

$$t \preceq_{max} t' \text{ iff } p(t) \leq p(t') \wedge q(t) \leq q(t').$$

The lack of monotonicity of $*$ is one of the main difficulties as regards proving the validity of trust computation. The relation $\preceq_{\Delta t}$ introduced in Lemma 4 is needed to overcome it. ■

In the following, we formulate the soundness of a computed trust using the partial order \preceq defined above.

Definition 6. We say a computed trust t concerning B is sound if $t \preceq t_B$. ■

In the context of this paper, it is generally impossible to totally and completely compute $t_B = \sum_{P \in \mathcal{P} - \{B\}} t_{PB}$. The intuition behind the definition is that t may be partial and inferior, but correct in the sense that it can be the result of a valid computation that does not contain duplicate counts of the basic trusts concerning B and thus treats them adequately as additive values.

Example 7. Suppose basic trusts are defined as in Fig. 3. A computed trust $s = t_{AB_1} * t_{B_1C} \uplus t_{AB_1} * t_{B_1B_2} * t_{B_2C} = (0.7695, 20)$ of A concerning C is sound since $(0.7695, 20) \preceq (0.9, 20) = t_C$. So as $s' = t_{AB_2} * t_{B_2C} = t_C$. However, $s \uplus s' = (0.765, 22)$ is not sound as a computed trust concerning C since $(0.765, 22) \not\preceq t_C$. In fact, t_{B_2C} is counted twice here. ■

However, it is insufficient to consider each computed trust for determining the validity of trust computation. As shown in Example 3, it is possible that the computed trust itself is sound but is overly influenced by a change in a

basic trust. We next define the stability of computation as not to occur such a problem. But here is a technical difficulty that procedures for trust computation discussed in this paper are partial (that is, may not output any value) and non-deterministic in general. In the next section, we present a procedure that distributedly computes trusts using a protocol that is non-deterministic with respect to the selection of the request's receivers and the construction of a response. Moreover, we assume that receivers of requests may ignore them or tell a lie. Below we formulate such a procedure as a function from the inputs to the set of possible outputs, and define stability using Hoare's preorder.

Definition 8. A basic trust assignment T (or simply, assignment) is a function from a pair of distinct persons to a trust. $T(A, B)$ denotes the basic trust of A concerning B with respect to T . Instead of $T(A, B)$, we also write t_{AB}^T , or simply t_{AB} , when T is apparent from the context. Moreover, the basic trust assignment obtained by increasing the value t_{EF} of T by Δt is denoted by $T \uplus_{EF} \Delta t$. ■

Definition 9. A trust computation procedure f is a procedure that, given assignment T and $A, B \in \mathcal{P}$ distinct as inputs, outputs a trust on termination. The set of all possible outputs of f with inputs T, A and B is denoted by $f(T, A, B)$. ■

Definition 10. A preorder \sqsubseteq on trust sets is defined as follows:

$$\mathcal{T} \sqsubseteq \mathcal{T}' \text{ iff } \forall t \in \mathcal{T} \exists t' \in \mathcal{T}' t \preceq t'.$$

Moreover, for a trust set \mathcal{T} and a trust t , we define $\mathcal{T} \uplus t$ as

$$\mathcal{T} \uplus t = \{t' \uplus t \mid t' \in \mathcal{T} \cup \{0\}\}.$$

Definition 11. We say a trust computation procedure f is stable if it satisfies the following properties for any distinct $A, B \in \mathcal{P}$:

1. $f(T, A, B) \sqsubseteq \{0\}$ if the total basic trust concerning B with respect to T is 0.
2. $f(T \uplus_{EF} \Delta t, A, B) \sqsubseteq f(T, A, B) \uplus \Delta t$ for any distinct $E, F \in \mathcal{P}$ and a trust Δt .

Intuitively, the second condition means that the computed trust $f(T \uplus_{EF} \Delta t, A, B)$ is bigger than $f(T, A, B)$ since the assignment for t_{EF} is increased by Δt , but the difference is bounded by Δt itself. Roughly speaking, the stability of the trust computation procedure means that the procedure adequately treats all basic trusts as additive values.

Example 12. Suppose that T is an assignment obtained from that in Example 3 by replacing t_{CD} with $(1, 0.5)$, and that $T' = T \uplus_{CD} (1, 0.5)$. Then, let us consider the (deterministic) procedure using the same formula as in the example.

$$\sum_{i=1}^{100} t_{AB_i}^T * t_{B_i C}^T * t_{CD}^T * t_{DE_i}^T * t_{E_i F}^T = (1, 50),$$

$$\sum_{i=1}^{100} t_{AB_i}^{T'} * t_{B_i C}^{T'} * t_{CD}^{T'} * t_{DE_i}^{T'} * t_{E_i F}^{T'} = (1, 100) \not\leq (1, 50) \uplus (1, 0.5).$$

Thus, this procedure is not stable. ■

Lemma 13. If a trust computation procedure is stable, then its output is sound as the computed trust concerning the third input. ■

Next, we present the assumption concerning a lie mentioned in Sect. 3.2 in a more general form using \preceq . In this paper, we assume that each lie from one person to another in a trust communication is determined by the communicated trust. The function representing the communicated trust containing a lie from B to A is called a lie function of B to A , denoted by L_{AB} . If B holds a (true) trust s and sends it to A , then A actually receives $L_{AB}(s)$. For lie functions, we assume that the following inequation holds:

$$t_{AB} * L_{AB}(s) \preceq s.$$

That is, any lie of B to A can be canceled by the application of “ $t_{AB} * _$ ”. We call this assumption the upper limit assumption on a lie.

We do not claim that this assumption is realistic. It is very strong, or rather too idealized. But what we are concerned with here is whether or not soundness and stability are conserved under such a strong and idealized assumption. In the next sections, we present a trust computation with linear terms, which is sound and stable without a lie. Under the limit assumption on a lie, soundness is conserved but its proof is nontrivial, and more surprisingly, there is a counter example for stability.

4.2 Computation with Linear Term

Let us consider a directed graph with people as vertices where each edge goes from a trustor to a trustee. We define linear terms to represent computation without duplicate counting. For any distinct $A, B \in \mathcal{P}$ we introduce a constant symbol \tilde{t}_{AB} called a basic trust symbol, and consider terms constructed with the symbols, \uplus and $*$.

Definition 14. Let A , B and C be any distinct vertices. We define an A - B linear term and the graph (a set of directed edges) represented by the term as follows.

1. \tilde{t}_{AB} is an A - B linear term representing the singleton set with the A - B edge as its only member.
2. If A - B linear terms S_1, \dots, S_n ($n \geq 1$) represent graphs that share no edges, then $S_1 \uplus \dots \uplus S_n$ is an A - B linear term representing $S_1 \cup \dots \cup S_n$.
3. If S is a B - C linear term and $A \in \mathcal{P}$ does not appear in the graph represented by S , then $\tilde{t}_{AB} * S$ is an A - C linear term that represents the graph S increased by the A - B edge. ■

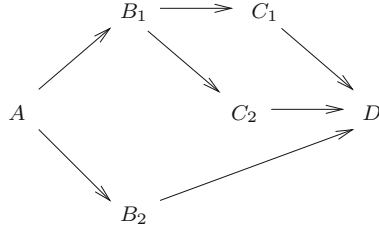


Fig. 4. Graph represented by linear term

Example 15. Figure 4 shows the graph represented by a linear term $\tilde{t}_{AB_1} * (\tilde{t}_{B_1C_1} * \tilde{t}_{C_1D} \uplus \tilde{t}_{B_1C_2} * \tilde{t}_{C_2D}) \uplus \tilde{t}_{AB_2} * \tilde{t}_{B_2D}$. There is no linear term representing the graph in Fig. 1. In this graph, a linear term can represent, for instance, its path $\tilde{t}_{AB_i} * \tilde{t}_{B_iC} * \tilde{t}_{CD}$. ■

Let C and D be any distinct persons. The following properties of the A - B linear term S derive directly from the definition.

- The graph represented by S contains the C - D edge iff \tilde{t}_{CD} appears in S .
- (Linearity) \tilde{t}_{CD} appears in S at most once.

Thus, given an A - B linear term S and an assignment T , the trust obtained from S by interpreting each occurrence of \tilde{t}_{CD} as t_{CD}^T is called the trust linearly computed by S with respect to T , denoted by $[S]^T$.

Lemma 16. For any A - B linear term S and assignment T , $[S]^T \preceq t_B$. ■

Note that we cannot employ simple induction on the construction of the term because of the lack of $*$'s monotonicity.

5 Trust Computation Protocol

In this section, we present our protocol for computing trust in a distributed manner. The results in this section depend only on the properties in Lemma 4, associativity, commutativity and zero of operators, and thus are independent of the specific definition of operators.

The basic protocol is a non-deterministic protocol exchanging the following messages:

Request: A pair $\langle C, P \rangle$ of the target C to whom a trust is computed in the session, and the sequence P along with which the request is relayed.

Response: A pair $\langle s, S \rangle$ of a computed trust s , and the linear term S by which s is linearly computed.

For any $A, A' \in \mathcal{P}$ distinct, we assume $t_{AA'} = 0$ if A has never communicated with A' . When A receives a request $\langle C, P \rangle$ from D , he processes it as follows:

1. If $t_{AC} \neq 0$, then A sends himself a response $\langle t_{AC}, \tilde{t}_{AC} \rangle$.
2. Then A non-deterministically chooses B_1, \dots, B_n satisfying the following three conditions and sends them a request $\langle C, P \cdot A \rangle$:
 - B_i is neither A nor C .
 - B_i does not occur in P .
 - $t_{AB_i} \neq 0$.
3. A waits as long as possible for responses from B_1, \dots, B_n .
4. From among the received responses, A chooses $\langle s_{B'_1C}, S_{B'_1C} \rangle, \dots, \langle s_{B'_kC}, S_{B'_kC} \rangle$ so that $S_{B'_1C}, \dots, S_{B'_kC}$ share no basic trust symbol with each other (if A chooses nothing, the process terminates immediately), and sends the pair $\langle t_{AB'_1} * s_{B'_1C} \uplus \dots \uplus t_{AB'_k} * s_{B'_kC}, \tilde{t}_{AB'_1} * S_{B'_1C} \uplus \dots \uplus \tilde{t}_{AB'_k} * S_{B'_kC} \rangle$ to D . If $B'_i = A$, then $t_{AB'_i} * s_{B'_iC}$ denotes $s_{B'_iC}$, and $\tilde{t}_{AB'_i} * S_{B'_iC}$ denotes $S_{B'_iC}$.

Here we are assuming that for every response a participant in the protocol can determine the corresponding request. In step 3 we do not have to wait for all responses from B_1, \dots, B_n ; the basic idea of this paper is that we cannot totally and completely compute the trusts. If someone wants to initiate a session to compute a trust concerning C , he sends himself a request $\langle C, \lambda \rangle$, where λ denotes the empty sequence.

Lemma 17. Let T be an assignment determined by the basic trusts all persons actually hold. Suppose, in a session with the basic protocol, no participant tells a lie, and A sends a response $\langle s, S \rangle$ answering a request $\langle C, P \rangle$. Then S is an A - C linear term and $s = [S]^T$. ■

Using the basic protocol, we can define the following trust computation procedure in a straightforward manner. Given inputs T , A and B ,

1. The basic trust of each person is determined according to T .¹
2. A initiates a trust computation session concerning B .
3. If A receives a response, he outputs its first element.

Figure 5 shows an example execution of the procedure $f(T, A, D)$ using the basic protocol. Suppose that the only non-zero basic trusts to D are $t_{B_1D} = (0.9, 10)$ and $t_{CD} = (1, 10)$ represented with solid arrows, and that $t_{B_1C} = (0.9, 30)$, $t_{B_2C} = (0.9, 5)$, $t_{AB_1} = (0.9, 15)$, and $t_{AB_2} = (0.9, 20)$. Assume that the participants do not tell a lie. Requests and responses are represented with dash arrows. The execution of $f(T, A, D)$ proceeds as follows:

- First A sends a request $\langle D, \lambda \rangle$ to himself, and receives it. Then he chooses receivers B_1 and B_2 , and sends them $\langle D, A \rangle$. Then he waits for the responses.

¹ We agree that it is unnatural that T determines each person's basic trust. In fact, each person's basic trust is given and the formal input T is determined accordingly.

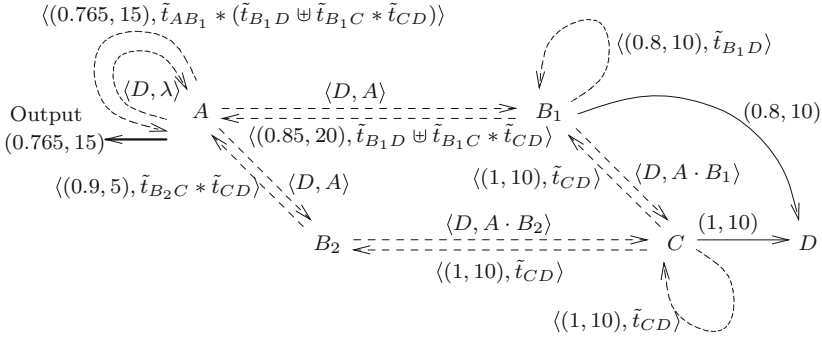


Fig. 5. Execution of trust computation

- Upon receiving the request from A , B_1 sends himself a response $\langle(0.8, 10), \tilde{t}_{B_1 D}\rangle$ since he holds non-zero basic trust concerning D . Then B chooses C to send a request $\langle D, A \cdot B_1 \rangle$ to C , then waits for the response.
- Upon receiving the request from B_1 , C sends himself a response $\langle(1, 10), \tilde{t}_{CD}\rangle$ since he holds non-zero basic trust concerning D . C chooses no receiver for the request, and thus it is the only response. So he sends response $\langle(1, 10), \tilde{t}_{CD}\rangle$ to B_1 .
- B_1 consolidates the two obtained responses, and sends a response $\langle s_{B_1}, S_{B_1} \rangle = \langle(0.85, 20), \tilde{t}_{B_1 D} \uplus \tilde{t}_{B_1 C} * \tilde{t}_{CD}\rangle$ to A .
- On the other hand, upon receiving the request from A , B_2 chooses C , sends a request $\langle D, A \cdot B_2 \rangle$ to C and waits. (He sends nothing to himself since his basic trust concerning D is 0.) C processes the request in the same way as with B_1 , and sends the response $\langle(1, 10), \tilde{t}_{CD}\rangle$ to B_2 . Upon receiving it, B_2 sends a response $\langle s_{B_2}, S_{B_2} \rangle = \langle(0.9, 5), \tilde{t}_{B_2 C} * \tilde{t}_{CD}\rangle$ to A .
- A receives the responses from B_1 and B_2 . Their linear terms share the same basic trust symbol \tilde{t}_{CD} , so A chooses the response from B_1 and sends a response $\langle s_A, S_A \rangle = \langle(0.765, 15), \tilde{t}_{AB_1} * (\tilde{t}_{B_1 D} \uplus \tilde{t}_{B_1 C} * \tilde{t}_{CD})\rangle$ to himself.
- Upon receiving of the response, A outputs $(0.765, 15)$.

Theorem 18. Let f be a trust computation procedure defined using the basic protocol.

1. Assume that, while executing f , the participants in the session tell lies only when they determine the first element of the request within the upper limit assumption on a lie. Then, every trust computed by f is sound.
2. If no protocol participant tells a lie, f is stable. ■

If a participant lies, the trust computation procedure defined by the basic protocol can be unstable. For instance, let us consider a situation in which B tells a lie when he provides A a trust s_{BC} . Let L_{AB} and L_{AB}^+ be lie functions when the basic trust of A concerning B is t_{AB} and $t_{AB} \uplus \Delta t$, respectively. Also suppose

$$\begin{aligned}
t_{AB} &= (1, 1), \\
\Delta t &= (0, 1), \\
s_{BC} &= (0.5, 2), \\
L_{AB}((0.5, 2)) &= (0.5, 2), \\
L_{AB}^+((0.5, 2)) &= (1, 2).
\end{aligned}$$

Note that in the above setting both L_{AB} and L_{AB}^+ satisfy the upper limit assumption on a lie, and $L_{AB}((0.5, 2))$ cannot have a larger value with respect to \preceq since $t_{AB} = (1, 1)$. In this case, however,

$$\begin{aligned}
(t_{AB} \uplus \Delta t) * L_{AB}^+(s_{BC}) &= (0.5, 2), \\
t_{AB} * L_{AB}(s_{BC}) \uplus \Delta t &= (0.25, 2).
\end{aligned}$$

Thus, the second condition of stability does not hold here.

6 Related Work

The problem of trust computation in a network has been studied in [5,6]. The authors use the two operators called discounting \otimes and consensus \oplus introduced in [4], which roughly correspond to the sequential and parallel compositions, respectively, in this paper. Two criticisms were presented [14] of their formulation of the discounting:

1. It does not have a natural interpretation in terms of evidence handling.
2. It is not distributive with respect to the consensus, that is, $t \otimes (t' \oplus t'') \neq (t \otimes t') \oplus (t \otimes t'')$.

As regards distribution, we do not think the equality always holds. However, they must be related, and the subjective logic does not provide any generic way to discuss it.

Thus, [14] proposed a reformulation of discounting based on scalar multiplication. The new discounting \boxtimes is defined as $t_{AB} \boxtimes t_{BC} = g(t_{AB}) \cdot t_{BC}$, where $g(x)$ is a non-negative real, and g can be chosen at will, depending on the context. This has a very simple interpretation in evidence space, and satisfies distribution with respect to consensus. But there is a problem regarding the choice of g . For instance, as for the friendship example in Introduction, it seems impossible to choose one discount rate $g(t_{AB})$.

To solve these problems, we separately and directly represent the uncertainty caused by the quantity of evidence concerning t as $q(t)$.

Semi-distribution is weaker than distribution, but has a very natural interpretation in the trust calculation with linear terms, that is, trust information t' and t'' obtained from two distinct sources is more trustworthy than $t' \uplus t''$ from one source. The information order \preceq on trusts enables us to reflect such a causally correct fact in the theory.

The notion of the canonical expression [5,6] corresponds to that of the linear term in this paper in the sense that these are expressions in which every person-to-person edge appears only once. The authors explain that canonical expressions

are necessary since the values of $t \otimes (t' \oplus t'')$ and $(t \otimes t') \oplus (t \otimes t'')$ differ. This is best understood as an independence issue. That is, the consensus operator works properly only for a pair of independent trust information, while $(t \otimes t')$ and $(t \otimes t'')$ are not independent of each other. However, the independence notion is explained very informally in [4].

On the other hand, canonical expressions are unnecessary for the trust calculation in [14] since the distribution of discounting holds there. But as mentioned above, there seem to be some cases where their definition of discounting is invalid.

We do not insist that linear terms are necessary for valid trust computation. The validity we need are soundness and stability, and the utilization of linear terms is a sufficient condition for them.

7 Discussion

For the actual implementation there are some problems to be solved. One is how to define the criterion of quantity; it should be uniform, independent of user preference. A promising candidate for a practically useful criterion is the monetary value. Another problem is how to determine a basic trust by direct and exclusive experiences. Usual pecuniary transactions naturally achieve this by determining the quantity of trust from the monetary value. It would be difficult to determine the trust by the experience that cannot be evaluated in terms of money, or that is shared with a large unspecified number of people. However, this seems to be an intrinsic problem whose solution needs psychological and sociological findings, beyond the scope of this paper.

8 Conclusion

We formulated a trust metric using a pair of quality and quantity, and presented the algebraic properties of its composition operations. Moreover, we defined the validity of the trust computation in terms of the operations, and thus we do not need probabilistic assumptions.

We can consider variations of trust formulations and composition definitions including a many-value extension of quality. We are also interested in a relaxation of the stability condition so that the basic protocol can be satisfied. Moreover, stable trust computation is closely related to the maximum flow problem. Extensions and clarifications in these directions constitute future work.

An evaluation is needed to justify the validity and efficacy of our approach. Building a prototype would be helpful for showing the advantages of the approach, e.g., the robustness of our metric against attacks in trust networks. These topic will also be considered future work.

References

1. Demolombe, R.: Reasoning about trust: a formal logical framework. In: Jensen, C., Poslad, S., Dimitrakos, T. (eds.) *iTrust 2004*. LNCS, vol. 2995, pp. 291–303. Springer, Heidelberg (2004). doi:[10.1007/978-3-540-24747-0_22](https://doi.org/10.1007/978-3-540-24747-0_22)

2. Demolombe, R.: Transitivity and propagation of trust in information sources: an analysis in modal logic. In: Leite, J., Torroni, P., Ågotnes, T., Boella, G., Torre, L. (eds.) CLIMA 2011. LNCS (LNAI), vol. 6814, pp. 13–28. Springer, Heidelberg (2011). doi:[10.1007/978-3-642-22359-4_2](https://doi.org/10.1007/978-3-642-22359-4_2)
3. Huang, J., Nicol, D.M.: A calculus of trust and its application to PKI and identity management. In: IDtrust 2009, pp. 23–37 (2009)
4. Jøsang, A.: A logic for uncertain probabilities. *Int. J. Uncertainty Fuzziness Knowl.-Based Syst.* **9**(3), 279–311 (2001)
5. Jøsang, A., Gray, E., Kinateder, M.: Simplification and analysis of transitive trust networks. *Web Intell. Agent Syst.* **4**(2), 139–161 (2006)
6. Jøsang, A., Hayward, R., Pope, S.: Trust network analysis with subjective logic. In: 29th Australasian Computer Science Conference, pp. 85–94 (2006)
7. Liau, C.-J.: Belief, information acquisition, and trust in multi-agent systems - a modal logic formulation. *Artif. Intell.* **149**(1), 31–60 (2003)
8. Lorini, E., Demolombe, R.: From binary trust to graded trust in information sources: a logical perspective. In: Falcone, R., Barber, S.K., Sabater-Mir, J., Singh, M.P. (eds.) TRUST 2008. LNCS (LNAI), vol. 5396, pp. 205–225. Springer, Heidelberg (2008). doi:[10.1007/978-3-540-92803-4_11](https://doi.org/10.1007/978-3-540-92803-4_11)
9. Lorini, E., Demolombe, R.: From trust in information sources to trust in communication systems: an analysis in modal logic. In: KRAMAS 2008, pp. 81–98 (2008)
10. Muller, T., Liu, Y., Zhang, J.: The fallacy of endogenous discounting of trust recommendations. In: AAMAS 2015, pp. 563–572 (2015)
11. Singh, M.P.: Trust as dependence: a logical approach. In: Sonenberg, L., Stone, P., Tumer, K., Yolum, P. (eds.), AAMAS 2011, pp. 863–870 (2011)
12. Theodorakopoulos, G., Baras, J.S.: Trust evaluation in ad-hoc networks. In: WiSe 2004, pp. 1–10 (2004)
13. Venkat Rangan, P.: An axiomatic basis of trust in distributed systems. In: IEEE S&P, pp. 204–211 (1988)
14. Škorić, B., de Hoogh, S., Zannone, N.: Flow-based reputation with uncertainty: evidence-based subjective logic. *Int. J. Inf. Secur.* **15**(4), 381–402 (2015)

Towards Statistical Trust Computation for Medical Smartphone Networks Based on Behavioral Profiling

Weizhi Meng¹(✉) and Man Ho Au²

¹ Department of Applied Mathematics and Computer Science,
Technical University of Denmark, Kongens Lyngby, Denmark
weme@dtu.dk

² Department of Computing, The Hong Kong Polytechnic University,
Hong Kong, Hong Kong
csallen@comp.polyu.edu.hk

Abstract. Due to the popularity of mobile devices, medical smartphone networks (MSNs) have been evolved, which become an emerging network architecture in healthcare domain to improve the quality of service. There is no debate among security experts that the security of Internet-enabled medical devices is woefully inadequate. Although MSNs are mostly internally used, they still can leak sensitive information under insider attacks. In this case, there is a need to evaluate a node's trustworthiness in MSNs based on the network characteristics. In this paper, we focus on MSNs and propose a statistical trust-based intrusion detection mechanism to detect malicious nodes in terms of behavioral profiling (e.g., camera usage, visited websites, etc.). Experimental results indicate that our proposed mechanism is feasible and promising in detecting malicious nodes under medical environments.

Keywords: Emerging network · Medical smartphone network · Intrusion detection · Insider attack · Statistical trust computation

1 Introduction

Over the past decade, healthcare has undergone a significant change through digitizing every aspect of medical infrastructure, including patient records, medical devices and patient/physician communication. As medical industry is evolving rapidly, mobile devices have become a popular platform to carry information and speed up electronic data transfers. For instance, smartphones have been applied in various healthcare organizations, helping record patient's medical conditions and access patient's records in real-time during ward visits. Due to the popularity of smartphones, an emerging medical network has been evolved, called *medical*

W. Meng—The author is previously known as Yuxin Meng.

smartphone network (MSN), which can be considered as a special kind of wireless sensor network [15]. These devices are generally connected to the organization's wireless network and each of them can be considered as a node. It is known that healthcare organizations (and networked medical devices) are particularly vulnerable to accidental failures, privacy violations, intentional disruption, and widespread disruption [4]. Therefore, there is a great need for protecting MSNs against various attacks, especially insider threats.

Due to the importance and sensitivity of MSNs, it is crucial to identify malicious devices within such network in a fast way. In this work, we advocate the effectiveness of trust-based IDSs and propose a statistical trust-based intrusion detection mechanism to identify malicious nodes in MSNs. In particular, our mechanism employs a statistical trust computation based on behavioral profiling. The contributions of our work can be summarized as below:

- Behavioral profiling is used by IDSs to model system or network events. In this work, we target on behavioral profiling and show how to build a behavioral profile in MSNs.
- As a study, we select four features (e.g., camera usage, visited websites) in building behavioral profiles. Accordingly, we develop a statistical trust computation method to evaluate a node's trustworthiness. Experimental results show that our approach is feasible and promising at identifying malicious MSN nodes in a quick manner.

The remaining parts of this paper are organized as follows. In Sect. 2, we introduce related studies on trust-based intrusion detection mechanisms. Section 3 describes our proposed intrusion detection mechanism and statistical trust computation with selected features. Section 4 describes and analyzes our evaluation, and Sect. 5 concludes our paper.

2 Related Work

Insider attacks are one of the major threats for distributed network systems like wireless sensor networks (WSNs). The basic question is how to properly evaluate the trustworthiness of a node.

Distributed trust-based intrusion detection. Collaborative intrusion detection networks (CIDNs) [16] have been proposed and implemented, which enable an IDS node to achieve more accurate detection by collecting and communicating information with other IDS nodes.

For instance, Li *et al.* [5] identified that most distributed intrusion detection systems (shortly DIDS) might rely on centralized fusion, or distributed fusion with unscalable communication mechanisms. They then proposed a distributed system according to the emerging decentralized location and routing infrastructure. They assumed that all peers are trusted, which makes the system vulnerable to insider attacks (i.e., betrayal attacks where some nodes suddenly become malicious). To detect insider attacks, Duma *et al.* [1] proposed a P2P-based overlay for intrusion

detection (Overlay IDS) that mitigated the insider threat by using a trust-aware engine for correlating alerts and an adaptive scheme for managing trust.

Challenge-based intrusion detection. Later, challenge-based CIDNs were proposed, where the trustworthiness of a node depends on the received answers to the challenges. Fung *et al.* [2] proposed a HIDS collaboration framework that enables each HIDS to evaluate the trustworthiness of others based on its own experience by means of a forgetting factor. The forgetting factor can give more emphasis on the recent experience of the peer. Then, they improved their trust management model by using a Dirichlet-based model to measure the level of trustworthiness among IDS nodes according to their mutual experience [3]. This model had strong scalability properties and was robust against common insider threats. Experimental results demonstrated that the new model could improve robustness and efficiency.

To improve the performance, Li *et al.* [6] pointed out that different IDSs may have different levels of sensitivity in detecting particular types of intrusions based on their own signatures and settings. They therefore defined a notion of *intrusion sensitivity* and explored the feasibility of using this notion to evaluate the trust of an IDS node. They further designed a trust management model based on *intrusion sensitivity* to improve the robustness of CIDNs [7], and proposed a machine learning-based approach in automatically allocating the values of *intrusion sensitivity* [8]. Other related studies on improving IDSs can be referred to alert reduction [9], alert verification [13,14] and filtration [10–12].

3 Our Approach

According to the recent study [15], a centralized architecture is desirable for detecting malicious nodes in MSNs, as healthcare organizations are often short of IT-trained personnel. Due to this, centralized security mechanisms can help reduce the number of potential attack vectors. Therefore, a hierarchical trust-based intrusion detection mechanism is one of the potential solutions, which can secure MSNs against insider attacks.

As medical networks are more special than traditional networks, healthcare organizations can define many strict rules and sensitive keywords to control the environment, so the network traffic could be relatively stable in most cases. Due to this, we believe that statistical approach can be used, which may be simple but efficient. Motivated by this, we propose a statistical trust-based intrusion detection mechanism to identify malicious nodes in MSNs.

The high-level detection flows are depicted in Fig. 1, including *behavioral data collection*, *profile construction*, *statistical trust computation*, and *detection and alert*. To collect behavioral data is a crucial step for establishing a robust trust-based intrusion detection scheme. The data are used to build a behavioral profile (as *normal behavior*). Then, the trustworthiness of a node can be computed by our statistical approach through identifying the deviations between the historical profile and current profile. Finally, an alert can be sent to security officers if any trust value is lower than a pre-defined threshold.

Behavioral Profiling in MSNs. As described earlier, a behavioral profile is a collection of required information aiming to describe the characteristics of an object under pre-defined rules. For instance, it is similar to a business card that contains some basic features like name, department and business phone number. To create a stable profile, there is a need for using sensible specifications to define the behavior.

Table 1 gives a list of basic features of smartphone users, such as phone calls (including outgoing, incoming and video), location, time, SMS, visited websites, Email address, application usage, etc. It is worth noting that this list provides some common features, but not a full list of those basic features. In MSNs, it is not possible to collect all these data due to its uniqueness and requirements (i.e., there is a chance of leaking information to third-parties). As a study, after communicating and seeking the suggestions from healthcare managers, we choose to collect four features in *each day* to construct a behavioral profile: camera usage, visited websites, Short Message Service (SMS) and Email address. All these features have the potential to be utilized to leak sensitive information, if a device is compromised by attackers.

Statistical Trust Computation. In MSNs, security policies usually define ‘good’ behavior; thus, it is not hard to detect anomalies. However, as network communication is dynamic and hard to predict, it can greatly increase false positives if identifying a malicious node via only one or two unusual events. As a result, trust values can be used to evaluate the severity of unusual behavior. As a study, our work proposes a statistical approach for computing a node’s trust value. The calculation of trust values (T) can be described as below:

Table 1. Basic features of smartphone users.

| | | |
|--------------------|------------------|-----------------------------|
| Outgoing calls | Incoming calls | Video calls |
| Location | Time | Short Message Service (SMS) |
| Favourite websites | Email address | IP of access points |
| Bluetooth ID | Camera usage | Application usage |
| Keystroke | Downloaded files | Media player usage |

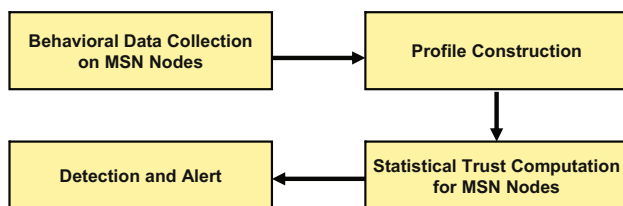


Fig. 1. The high-level typical detection flows.

$$T = 1 - \prod_{k=1}^{k=n} \frac{M}{I} \quad (n, M, I \in \mathbf{N}) \quad (1)$$

where n denotes the number of features, M represents the number of malicious activities, and I represents the total number of recorded events. Taking two features A and B as an example, if there are two out of ten events and one out of eight events are malicious for A and B respectively, then the trust value is 0.975 ($1 - 2/10 * 1/8$). Based on Eq. (1), a malicious node can be determined by setting a trust threshold. Let τ denote the trust threshold, then we can consider:

- If $T \geq \tau$, then the node is considered as a normal node.
- If $T < \tau$, then the node is regarded as a malicious (or untrusted) node.

According to the features of MSNs, our hierarchical trust-based intrusion detection mechanism has two major advantages. (1) *Simple but efficient*. According to Eq. (1), the calculation of trust values is easy through recording required information and data. In addition, the existing central server can mostly have enough computational power and storage space in our scenario. (2) *Fault Tolerance*. As smartphone usage is dynamic, it may produce many false positives by detecting malicious nodes via only one or two unusual events. Thus, our approach considers a set of features in computing trust values, aiming to provide good fault tolerance in practical applications.

4 Evaluation

In this section, we evaluate our approach in a healthcare environment located in China. Due to privacy concerns, our mechanism was deployed in a partial MSN, which consists of 10 nodes. A central server was used to collect relevant information from each node and compute trust values, which was composed of an Intel(R) Core (TM)2, Quad CPU 2.66 GHz. In particular, we conduct two major experiments. (1) The first experiment evaluates our mechanism in a normal MSN environment, aiming to observe the trend of trust values and identify a proper threshold. (2) The second experiment explores the feasibility of our mechanism under an adversary scenario, where we randomly select some nodes to behave maliciously (i.e., violating normal profile).

4.1 Experiment-1

In this experiment, we attempt to observe the trend of trust values in a normal MSN environment. According to Eq. (1), it is easily understandable if M becomes smaller, then T will become larger. As M is always smaller than I , T should fall into the range of $[0,1]$. A larger T means that a node is more credible. Ideally, T is expected to 1; however, it is very hard to achieve this in real scenarios. Therefore, a major goal of this experiment is to identify a proper threshold for detecting malicious nodes in MSNs. The trend of average and the lowest trust value within a month is depicted in Fig. 2. The main observations are described as follows.

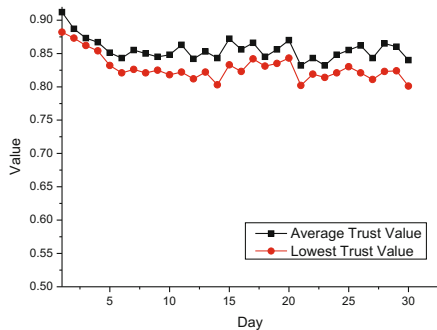


Fig. 2. The trend of average and the lowest trust value.

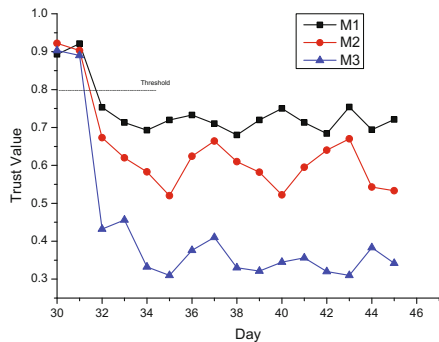


Fig. 3. The trust values of malicious nodes in the MSN.

- Average trust value is an average value of all nodes' trust values, which reflects the overall network performance. Figure 2 shows that the trend of average trust value was higher than 0.85 for the first five days. This is because the MSN was just initialized and each node required a period of time in connecting with other nodes. Afterwards, average trust value became more stable, ranging from 0.84 to 0.87.
- The MSN has 10 nodes, so that the lowest trust value indicates the worst node's performance. Similarly, Fig. 2 describes that the trend of the lowest trust value was peak during the first five days, but gradually decreased and became stable later, ranging from 0.8 to 0.85.

On the whole, it is observed that trust values can be always higher than 0.8 in a normal MSN environment. We believe that the collected one month' data can represent a common MSN performance. Therefore, we choose 0.8 as the trust threshold in this work.

4.2 Experiment-2

In this experiment, we aim to evaluate the performance of our approach in a malicious scenario, where some nodes act unusually, i.e., violating the defined profile. More specifically, we randomly selected three nodes (named $M1$, $M2$ and $M3$) as malicious to launch unusual events. For example, one node may visit unusual websites in a random way, or send an email to an undefined receiver. The unusual events for each malicious node are summarized in Table 2, where each node could make different unusual events. The malicious actions started from Day 31, and the trust values of these malicious nodes are depicted in Fig. 3. The main observations are described as below.

- As introduced, all three nodes started conducting unusual behavior from Day 31, it is observed that their trust values could quickly decrease to below the threshold of 0.8 at the same day. The trust value of $M1$, $M2$ and $M3$ ranged from 0.7 to 0.8, from 0.5 to 0.7, and from 0.3 to 0.45, respectively.

Table 2. Simulated unusual events for each malicious node.

| Node | Camera usage | Visited websites | SMS | Email address |
|-----------|--------------|------------------|-----|---------------|
| <i>M1</i> | ✓ | - | - | - |
| <i>M2</i> | - | ✓ | ✓ | - |
| <i>M3</i> | ✓ | ✓ | - | ✓ |

- As shown in Table 2, *M1* only violated the usage of camera, while *M3* performed unusual events in relation to camera usage, visited websites and Email address. As a result, *M3* got the lowest trust value among the three malicious nodes.

Overall, the experimental results indicate that threshold of 0.8 is appropriate in our settings, and our mechanism is feasible and promising to identify malicious nodes in a quick manner (i.e., identifying malicious nodes at the same day). Generally, more unusual events result in a lower trust value. This conclusion is also confirmed by IT administrators in the participating healthcare organization.

5 Conclusion

With more devices interconnected, medical smartphone networks (MSNs) have become an emerging architecture in healthcare organizations. In this work, we focus on MSNs and propose a statistical trust-based intrusion detection mechanism by combining behavioral profiling and statistical trust computation to detect anomalies. A hierarchical infrastructure is adopted to help control trust computation and apply security policies in MSNs. Experimental results indicate that our proposed mechanism is feasible and encouraging in detecting malicious nodes in a quick manner. This is an early study on designing appropriate trust-based intrusion detection schemes for medical networks. There are many possible topics for our future work. One is to investigate how to efficiently identify a trust threshold in different network environments. It is also an interesting topic to consider more features in trust computation, exploring the impact of each feature and developing a weighted statistical trust computation.

Acknowledgments. We would like to thank the cooperation from the participating healthcare managers and IT administrators. Part of this work was supported by the National Natural Science Foundation of China (Grant No. 61602396).

References

1. Duma, C., Karresand, M., Shahmehri, N., Caronni, G.: A trust-aware, P2P-based overlay for intrusion detection. In: DEXA Workshop, pp. 692–697 (2006)
2. Fung, C.J., Baysal, O., Zhang, J., Aib, I., Boutaba, R.: Trust management for host-based collaborative intrusion detection. In: De Turck, F., Kellerer, W., Kormentzas, G. (eds.) DSOM 2008. LNCS, vol. 5273, pp. 109–122. Springer, Heidelberg (2008). doi:[10.1007/978-3-540-87353-2_9](https://doi.org/10.1007/978-3-540-87353-2_9)

3. Fung, C.J., Zhang, J., Aib, I., Boutaba, R.: Robust and scalable trust management for collaborative intrusion detection. In: Proceedings of the 11th IFIP/IEEE International Conference on Symposium on Integrated Network Management (IM), pp. 33–40 (2009)
4. Healey, J., Pollard, N., Woods, B.: The Healthcare Internet of Things: Rewards and Risks, March 2015. <http://www.mcafee.com/mx/resources/reports/rp-healthcare-iot-rewards-risks.pdf>
5. Li, Z., Chen, Y., Beach, A.: Towards scalable and robust distributed intrusion alert fusion with good load balancing. In: Proceedings of the 2006 SIGCOMM Workshop on Large-Scale Attack Defense (LSAD), pp. 115–122 (2006)
6. Li, W., Meng, Y., Kwok, L.-F.: Enhancing trust evaluation using intrusion sensitivity in collaborative intrusion detection networks: feasibility and challenges. In: Proceedings of the 9th International Conference on Computational Intelligence and Security (CIS), pp. 518–522. IEEE (2013)
7. Li, W., Meng, W., Kwok, L.-F.: Design of intrusion sensitivity-based trust management model for collaborative intrusion detection networks. In: Zhou, J., Gal-Oz, N., Zhang, J., Gudes, E. (eds.) IFIPTM 2014. IFIP AICT, vol. 430, pp. 61–76. Springer, Heidelberg (2014). doi:[10.1007/978-3-662-43813-8_5](https://doi.org/10.1007/978-3-662-43813-8_5)
8. Li, W., Meng, Y., Kwok, L.-F., Ip, H.H.S.: Enhancing collaborative intrusion detection networks against insider attacks using supervised intrusion sensitivity-based trust management model. *J. Netw. Comput. Appl.* **77**, 135–145 (2017)
9. Meng, Y., Kwok, L.-F.: Enhancing false alarm reduction using voted ensemble selection in intrusion detection. *Int. J. Comput. Intell. Syst.* **6**(4), 626–638 (2013)
10. Meng, Y., Kwok, L.-F., Li, W.: Towards designing packet filter with a trust-based approach using Bayesian inference in network intrusion detection. In: Keromytis, A.D., Pietro, R. (eds.) SecureComm 2012. LNICST, vol. 106, pp. 203–221. Springer, Heidelberg (2013). doi:[10.1007/978-3-642-36883-7_13](https://doi.org/10.1007/978-3-642-36883-7_13)
11. Meng, Y., Li, W., Kwok, L.: Evaluation of detecting malicious nodes using Bayesian model in wireless intrusion detection. In: Lopez, J., Huang, X., Sandhu, R. (eds.) NSS 2013. LNCS, vol. 7873, pp. 40–53. Springer, Heidelberg (2013). doi:[10.1007/978-3-642-38631-2_4](https://doi.org/10.1007/978-3-642-38631-2_4)
12. Meng, W., Li, W., Kwok, L.-F.: EFM: enhancing the performance of signature-based network intrusion detection systems using enhanced filter mechanism. *Comput. Secur.* **43**, 189–204 (2014)
13. Meng, Y., Kwok, L.-F.: Adaptive blacklist-based packet filter with a statistic-based approach in network intrusion detection. *J. Netw. Comput. Appl.* **39**, 83–92 (2014)
14. Meng, W., Li, W., Kwok, L.-F.: Design of Intelligent KNN-based alarm filter using knowledge-based alert verification in intrusion detection. *Secur. Commun. Netw.* **8**(18), 3883–3895 (2015)
15. Meng, W., Li, W., Xiang, Y., Choo, K.-K.R.: A Bayesian Inference-based detection mechanism to defend medical smartphone networks against insider attacks. *J. Netw. Comput. Appl.* **78**, 162–169 (2017)
16. Wu, Y.-S., Foo, B., Mei, Y., Bagchi, S.: Collaborative intrusion detection system (CIDS): a framework for accurate and efficient IDS. In: Proceedings of the 2003 Annual Computer Security Applications Conference (ACSAC), pp. 234–244 (2003)

Reputation Systems

Reputation-Enhanced Recommender Systems

Christian Richthammer^(✉), Michael Weber, and Günther Pernul

Department of Information Systems, University of Regensburg, Regensburg, Germany
{christian.richthammer,michael.weber,guenther.pernul}@ur.de
<http://www-ifs.uni-regensburg.de>

Abstract. Recommender systems are pivotal components of modern Internet platforms and constitute a well-established research field. By now, research has resulted in highly sophisticated recommender algorithms whose further optimization often yields only marginal improvements. This paper goes beyond the commonly dominating focus on optimizing algorithms and instead follows the idea of enhancing recommender systems with reputation data. Since the concept of reputation-enhanced recommender systems has attracted considerable attention in recent years, the main aim of the paper is to provide a comprehensive survey of the approaches proposed so far. To this end, existing work are identified by means of a systematic literature review and classified according to carefully considered dimensions. In addition, the resulting structured analysis of the state of the art serves as a basis for the deduction of future research directions.

Keywords: Recommender systems · Decision support systems · Reputation · Trust · Reputation-enhanced recommender systems

1 Introduction

The rise of the World Wide Web has made sharing and accessing various kinds of information easier and faster than ever before. However, this trend has also led to the phenomenon of information overload, which may overwhelm users in the course of their decision making processes [17]. Recommender systems are intended to solve this problem by making users aware of only those items they are probably interested in [18, 31]. Because of the constantly high research interest in the development of techniques predicting how much users will like different items, recommender algorithms are highly sophisticated by now. Further optimization efforts often yield only marginal improvements [26, 33]. Therefore, it has been suggested to broaden the horizon of recommender systems research and integrate relevant concepts from related fields.

Trust and reputation systems show substantial connections to recommender systems, especially to collaborative filtering systems [19]. Thus, there are several proposals on trust-enhanced recommender systems [41]. These systems consider trust in the form of explicitly declared trust or friendship relationships

(e.g. web of trust on Epinions¹) in the recommendation process. However, these trust links are only available in small numbers because modern online platforms are typically characterized by short-term interactions in a “universe of strangers” [14]. In addition to this main limitation, the explicit declaration of trust relationships requires considerable efforts from users [5].

Because of these drawbacks of explicit trust links, this paper specifically focuses on the enhancement of recommender systems with reputation data. Reputation is another kind of construct relevant when taking advice from others [5]. It is closely linked to trust [19] or even used to establish trust (“reputation-based trust” [6]). However, it fits the aforementioned peculiarities of modern online platforms better. Reputation values are calculated on a global scale instead of being limited to the trust links of one single user. On the one hand, this mitigates the problem of sparsely available personal trust links. On the other hand, reputation values are computationally less expensive because they are computed once for the entire community whereas trust values have to be determined from the perspective of every individual user [28]. Since the concept of reputation-enhanced recommender systems has attracted considerable attention in recent years, several combination approaches have been proposed. In this paper, we comprehensively identify the existing methods by means of a systematic literature review based on well-established guidelines and classify them according to carefully considered dimensions. Thus, the state of the art of reputation-enhanced recommender systems is revealed in an exhaustive manner. Moreover, we are able to point out possible directions for future work in this research stream. In general, our results also provide an important basis for the further exchange of ideas between recommender and reputation systems researchers.

The remainder of the paper is organized as follows. Section 2 introduces the main principles of recommender and reputation systems and relates them to each other according to their similarities and differences. Based on this, Sect. 3 discusses the process and the outcomes of a systematic literature review on reputation-enhanced recommender systems. This, in turn, leads to the formulation of future research directions in Sect. 4. Section 5 concludes the paper.

2 Background

Modern Internet platforms, such as e-commerce marketplaces and social media websites, are omnipresent in today’s society. Recommender and reputation systems are pivotal decision support components of these platforms.

2.1 Recommender Systems Principles

As already mentioned, the main motivation for the use of recommender systems is the information overload problem [31]. To tackle this issue, recommender systems are supposed to provide users with only the most relevant information

¹ <http://www.epinions.com/>.

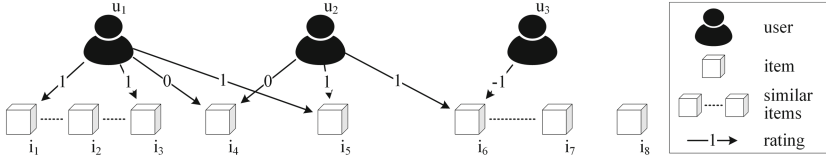


Fig. 1. Exemplary user-item relations using $\{-1, 0, 1\}$ as possible rating values.

and only those items that are worth considering. This is done by predicting the ratings of the items a particular user has not rated yet and recommending those which receive the highest predicted ratings. Figure 1 depicts the entities and relationships considered in the two main types of recommender systems: collaborative filtering and content-based filtering [3].

Collaborative filtering [15, 38] is based on the idea that people tend to agree with people they agreed with in the past and thus captures the typical human behavior of relying on the opinions of acquaintances with similar tastes. When employing the user-based nearest neighbor algorithm, as one particular form of collaborative filtering, the predicted ratings for each item are calculated by aggregating the ratings of the other users weighted by their similarities (in rating behaviors) to the user in focus. Ratings can take different forms such as $\{0, 1\}$ (has experiences, has no experiences), $\{-1, 0, 1\}$ (negative, neutral, positive), and $\{1, 2, 3, 4, 5\}$ (opinions from very negative to very positive). In the example depicted in Fig. 1, user u_1 is similar to u_2 as both assigned the same rating to item i_4 and i_5 , respectively. u_1 is less similar to u_3 as they do not have any ratings in common. Since u_2 has positively rated i_6 , which has not been rated by u_1 yet, a user-based collaborative filtering system would recommend i_6 to u_1 .

By contrast, content-based filtering [27] assumes that people will like items similar to the ones they liked in the past. It is solely based on the user's own ratings and the similarities of items determined according to their features. In the example depicted in Fig. 1, u_1 has positively rated i_1 and i_3 . Since i_2 is similar to i_1 and i_3 , a content-based filtering system would recommend i_2 to u_1 .

2.2 Reputation Systems Principles

Reputation systems [19] are needed because users usually have no or only few direct experiences with other users on digital platforms. Thus, a user does not know whether to trust another user or not. Reputation systems can alleviate this issue by assisting the user in determining the trustworthiness of other users. Figure 2 depicts the entities and relationships involved in the calculation of users' reputation values indicating their trustworthiness.

After each encounter, users are able to rate the behavior of their counterpart. In e-commerce, for example, a customer can judge a seller's behavior according to factors like on-time delivery and adequate product quality. Similar to recommender systems, common rating scales are $\{-1, 0, 1\}$ and $\{1, 2, 3, 4, 5\}$. The reputation system collects the feedback data and employs them to calculate a

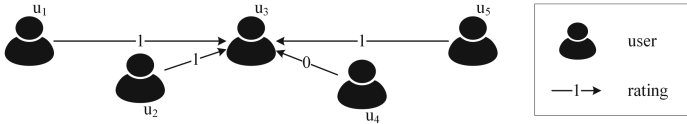


Fig. 2. Exemplary user-user relations using $\{-1, 0, 1\}$ as possible rating values.

reputation value for each user according to the following process [35]. At first, the reputation system may filter or weight the ratings depending on different parameters such as the timestamp of the encounter. Then, it aggregates the ratings by employing one of several possible aggregation techniques (e.g. arithmetic mean). Finally, the reputation system communicates the aggregated reputation values to the users of the platform. In the example depicted in Fig. 2, u_3 has received one neutral and three positive ratings. As a result, a reputation system using no filtering or weighting criteria and using the arithmetic mean as its aggregation technique would assign a reputation value of 0.75 to u_3 .

2.3 Relating Reputation Systems to Recommender Systems

As can be inferred from the remarks in the preceding subsections, the main similarity of recommender and reputation systems is that both kinds of decision support systems are based on user experiences and feedback [19]. Moreover, the two kinds of systems are frequently applied in similar contexts. Besides e-commerce as the most important of the common application areas, other examples include online communities, service selection, and peer-to-peer networks. These fundamental similarities make combined considerations feasible and allow creating a common feedback model as depicted in Fig. 3. The model includes two sets of entities: users $U = \{u_1, u_2, \dots, u_n\}$ and items $I = \{i_1, i_2, \dots, i_m\}$. Users can have experiences with items, which are referred to as the set of item ratings $IR \subseteq U \times I$ (with rating values $r_{IR} : IR \mapsto R$). IR is usually focused on by recommender systems. Furthermore, users can have experiences with other users, which are referred to as the set of user ratings $UR \subseteq U \times U$ (with rating values $r_{UR} : UR \mapsto R$). UR is usually focused on by reputation systems.

Moreover, recommender and reputation systems differ in certain facets and assumptions, which makes combined considerations potentially meaningful [19]. Recommender systems emphasize the similarity of users regarding their subjective tastes whereas reputation systems are especially applied to taste-independent aspects [20]. Therefore, the calculations of (collaborative filtering) recommender systems are typically based on the opinions of local communities consisting of the most similar users [3]. As opposed to this, the calculations of reputation systems are mostly done on a global basis because reputation is considered as a collective measure of trustworthiness [19]. Thus, recommendation values are subjective and determined from the perspective of one particular entity whereas reputation values are objective and the same from the perspectives of all entities.



Fig. 3. Common feedback model of recommender and reputation systems.

3 State of the Art

Based on the background information introduced in the previous section, we survey the state of the art of reputation-enhanced recommender systems. To this end, we conduct a systematic literature review following the well-recognized guidelines by Webster and Watson [45] and Levy and Ellis [22]. In particular, we act on the eight-step process by Okoli and Schabram [30], which specifies these guidelines in detail.

3.1 Literature Review Protocol

In order to fulfill the demand of vom Brocke et al. [42] that not only the findings of a literature review but also the process of searching and filtering the literature should be comprehensively described, the implementation of each of Okoli and Schabram’s eight steps [30] is discussed in the following.

(1) Purpose of the literature review. By systematically examining the existing ways to enhance recommender systems with reputation data and relating them to one another, the state of the art of this research stream is revealed.

(2) Protocol and training. When conducting a systematic literature review, it is crucial to act according to a detailed protocol. The most important aspects are pointed out for each step within this subsection. Training is not applicable to this paper because the literature review has essentially been conducted by the first author only. Nevertheless, conceptual feedback by the co-authors has been taken into consideration.

(3) Searching for the literature. The main issue to consider regarding the literature search is systematics. In this literature review, the following five digital libraries are used: ACM Digital Library, AIS Electronic Library, IEEE Xplore Digital Library, ScienceDirect, and Scopus. As demanded by vom Brocke et al. [42], they are chosen because they provide access to the journals and conference proceedings that are most relevant to the topic of this paper. In order to discover as many potentially relevant publications as possible, we use the very general search phrase “recommend* AND reputation”. We also use the search phrase “collaborative AND reputation” because there are several publications in the recommender systems field mentioning only collaborative filtering instead of recommender systems in general. Since recommender systems are relevant in

multiple research disciplines (e.g. computer science, engineering, mathematics), we do not exclude any of them from the initial search. We also do not exclude any work based on the year of publication. Moreover, we search for both journal articles and conference papers. The initial search carried out in November 2016 resulted in 420 hits at ACM, 19 hits at AIS, 341 hits at IEEE Xplore, 241 hits at ScienceDirect, and 1,367 hits at Scopus.

(4) Practical screen. Since we use very general search phrases and do not exclude any disciplines from our search, we receive a high number of initial search results (especially considering the narrow focus of this paper). All these publications enter the screening process by title, in which many of the clearly irrelevant ones can be removed. The relevance of the remaining papers is then judged based on their abstracts. Again, they are removed only if they are clearly not applicable to the scope of the literature review. If there are any doubts about their relevance, they are kept for the time-consuming full text review. In order to be relevant, a proposal first of all has to contain both an actual recommender and an actual reputation component. On the one hand, this excludes papers using the term “recommendation” to describe a rating or second-hand information in the reputation systems domain. On the other hand, this also excludes work creating recommendations by simply ranking items according to their reputation values. In addition, publications are considered as relevant only if the calculations of recommendation and reputation values as well as the combinations of recommender and reputation components are sufficiently described.

(5) Quality appraisal. Publications may be judged based on the ranking of their outlets. Since we examine an emerging research stream for which the number of publications in top journals and at top conferences is still low, however, we do not limit our focus to highly recognized and popular work only.

(6) Data extraction. In this step, the information from those publications the full text review brings forth as relevant are collected. In order to be able to compare the publications in a structured manner, we develop a dedicated taxonomy as a basis for the data extraction step (cf. Sect. 3.2). Particular attention is paid to the hybridization approach, the type of recommender system, and the evaluation described in the paper.

(7) Synthesis of studies. Based on the notes of the data extraction step, the relevant publications are analyzed in detail. With the help of our taxonomy, we provide a structured overview of existing work (cf. Sect. 3.3) and are able to identify directions for future research efforts (cf. Sect. 4).

(8) Writing the review. Presenting the insights gained in the synthesis step concludes the eight-step process of conducting a systematic literature review.

3.2 Taxonomy Development

As previously described, the data extraction step requires the excerpting of the publications judged as relevant in the full text review. In the following, a taxonomy providing a clear structure for this activity is developed.

First and foremost important, reputation-enhanced recommender systems can be analyzed according to their **hybridization approaches**. Following Burke’s [10] overview of methods for the hybridization of two or more recommendation techniques, we define the first dimension for distinguishing different approaches to enhance recommender systems with reputation data: the *hybridization method* dimension. We adapt the methods listed by Burke [10] to the hybridization scenario of this paper, resulting in the following six categories:

- *Weighted*: The respective outputs of a recommender and a reputation system are combined based on a weighting factor.
- *Switching*: If a recommender system is not able to generate enough suggestions, a reputation system is used instead or in addition.
- *Mixed*: The outputs of both systems may be presented at the same time. In particular, the final recommendation value is high only if both individual values are high.
- *Rec-*rep*-cascade*: A reputation system refines the output of a recommender system.
- *Rep-*rec*-cascade*: A reputation system pre-filters the input for a recommender system.
- *Augmentation*: Reputation data is considered directly within the calculations of the recommender system.

Furthermore, Fig. 3 (cf. Sect. 2.3) shows that there are two kinds of data bases in connection with recommender and reputation systems: *IR* used for item-related feedback and *UR* used for user-related feedback. Although it is most common for recommender systems to operate on *IR* and for reputation systems to operate on *UR*, both systems can also use the respective other data base. For example, there are recommender systems for contact recommendation on online social network sites (i.e. employing *UR*) as well as reputation systems for the taste-independent judgment of products (i.e. employing *IR*). Therefore, when enhancing recommender systems with reputation data, there are four combination possibilities regarding the chosen data base of the systems (cf. Table 1). Based on these four possibilities, we deduce the second dimension of the taxonomy employed for the data extraction: the *data base* dimension. It features two categories. First, recommender and reputation systems can use *different data bases*. Second, they can use the *same data base*.

Table 1. Combining recommender and reputation systems based on their data bases.

| | Recommender system | Reputation system | Data base dimension |
|---|--------------------|-------------------|----------------------|
| 1 | <i>IR</i> | <i>UR</i> | Different data bases |
| 2 | <i>UR</i> | <i>IR</i> | |
| 3 | <i>IR</i> | <i>IR</i> | Same data base |
| 4 | <i>UR</i> | <i>UR</i> | |

In addition, reputation-enhanced recommender systems can be compared according to the underlying **types of recommender system**. Therefore, the third dimension focuses on the *recommendation approach*. Regarding its categories, we distinguish between the three commonly accepted approaches [3]: content-based filtering (*CbF*), collaborative filtering (*CF*), and hybrid (*CbF/CF*). Although the ideas behind recommendation algorithms are generally applicable to different contexts, the respective publications typically focus on a specific domain. This constitutes the fourth dimension of the taxonomy: the *application area* dimension. Possible values include *movies*, *products*, and *hotels*. However, we do not define a fixed list of categories for this dimension at this point because there is no comprehensive list in the literature we could rely on.

Apart from the characteristics of the developed systems, it is crucial to judge publications according to their **evaluations** because not all kinds of evaluation may proof the value of a proposal equally well. For example, real-world case studies are more meaningful than fictional scenarios by far. Here, we rely on the “how” of evaluation as described by Prat et al. [32] and adapt the dimensions and categories that are most relevant to our analysis. First, there is the *evaluation technique* dimension with its categories: *case study*, *field study*, *action research*, *static analysis*, *dynamic analysis*, *controlled experiment*, *simulation*, *testing*, *informed argument*, *scenario*, *survey*, and *focus group*. And second, there is the *relativeness* dimension with its categories: *absolute* and *relative*.

3.3 Overview of Existing Work

In total, our full text review consists of 82 papers published between 2004 and 2017. In the following, the ideas of the work finally judged as relevant to the scope of this paper are comprehensively described. The remarks are structured according to the hybridization method dimension. In addition, Table 2 compares the publications according to the complete taxonomy developed in Sect. 3.2. Please note that Abdel-Hafez et al. [1] describe two distinct hybridization approaches in their paper.

Weighted. McNally et al. [29] introduce a weighted hybridization approach for the HeyStaks social search platform [36] in which recommender and reputation values are based on different data bases. The recommender component determines the relevance scores of the search results with respect to a given search query whereas the reputation component aggregates the reputation scores of those HeyStaks members that are responsible for the existence of the search results. Alotaibi and Vassileva [4] pursue a similar approach for their recommender system for scientific papers. The recommender component is based on the content similarity between a candidate paper and the user’s current interests as well as on the ratings other users have assigned to the paper. The reputation component relies on the reputation of the author of the candidate paper (e.g. h-index). In the crowdsourcing recommender of Wang et al. [43], the recommender component identifies appropriate tasks based on user similarities whereas the reputation component relies on the reputations of the task requesters. The

Table 2. Publications compared according to the developed taxonomy.

| Ref. | Hybridiz. method | Data base | Recommend. approach | Application area | Evaluation technique | Relativeness |
|------|------------------|-----------|---------------------|------------------|----------------------|--------------|
| [4] | Weighted | Different | CbF/CF | Documents | n/a | n/a |
| [13] | Weighted | Different | CF | Products | Contr. exp | Relative |
| [29] | Weighted | Different | CF | Search | Contr. exp. | Relative |
| [43] | Weighted | Different | CF | Crowdsourcing | Contr. exp. | Relative |
| [1] | Weighted | Same | CF | Movies | Case study | Relative |
| [2] | Weighted | Same | CF | Movies | Case study | Relative |
| [44] | Weighted | Same | CbF/CF | Products | Case study | Relative |
| [7] | Switching | Same | CF | Restaurants | Scenario | Relative |
| [8] | Switching | Same | CF | Tourism | Scenario | Relative |
| [9] | Switching | Same | CF | Restaurants | Scenario | Relative |
| [18] | Mixed | Same | CF | Hotels | Scenario | Absolute |
| [47] | Mixed | Same | CF | Applications | Simulation. | Absolute |
| [48] | Mixed | Same | CbF | Tourism | Simulation. | Absolute |
| [12] | Rec-rep-c. | Different | CbF/CF | Products | n/a | n/a |
| [1] | Rec-rep-c. | Same | CF | Movies | Case study | Relative |
| [21] | Rec-rep-c. | Same | Not def. | Products | Contr. exp. | Absolute |
| [16] | Rep-rec-c. | Different | CbF/CF | Documents | Simulation. | Absolute |
| [40] | Rep-rec-c. | Different | CF | Services | Contr. exp. | Absolute |
| [49] | Rep-rec-c. | Different | CF | Products | Case study | Absolute |
| [11] | Augment. | Different | CF | News | Simulation. | Absolute |
| [23] | Augment. | Different | CbF | Documents | Case study | Relative |
| [24] | Augment. | Different | CbF | Documents | Case study | Relative |
| [25] | Augment. | Different | CbF/CF | Blog articles | Case study | Relative |
| [34] | Augment. | Different | CF | Products | Contr. exp. | Relative |
| [37] | Augment. | Different | CF | Web services | Contr. exp. | Relative |
| [39] | Augment. | Different | CF | Products | Case study | Relative |

system proposed by Cui et al. [13] combines the reputation value of an item (determined according to its favorable rating ratio) with the recommendation value of the user providing the respective item. Abdel-Hafez et al. [1] describe a weighted hybridization method in which the recommender and the reputation system use the same data base. The first step is to perform the Borda count method separately for the ranked output lists of the recommender system and the reputation system. By assigning weights to the two Borda count lists, the weighted sum of the Borda count scores is determined for each item. The item with the highest total score is recommended to the user. Abdel-Hafez et al. [2] introduce a recursive variant of this approach. In another proposal belonging to this category, Wang et al. [44] suggest the weighted enhancement of a product's recommendation value with its reputation and its purchase frequency.

Switching. The switching method is used by Bedi et al. [7] in their restaurant recommender termed SRPRS. The system produces a list of recommendations based on the degrees of importance of the items retrieved from similar users. Only if the recommendation list does not contain as many items as requested, it is extended based on the degrees of importance of all items whose reputation values are greater than some threshold. The ideas of SRPRS can also be found in two other proposals identified in the literature review: MARST [8] and SAPRS [9]. Although the exact items considered for these systems may slightly differ (MARST considers not only restaurants but also hotels and points of interest), they all focus on scenarios in which the recommender and the reputation component rely on the same data base.

Mixed. The service recommender developed by Yazidi et al. [48] is divided into several subsystems. Among others, there is a recommender component identifying relevant services based on the user's context and profile as well as a reputation component managing the reputation value of the services. A service is recommended only if it is positively evaluated by all subsystems. Yan et al. [47] describe a system to recommend the usage of mobile applications based on the applications' local recommendation values as well as their public reputation values. The applications are recommended only if they possess both a high personalized recommendation value and a high public reputation value. Jøsang et al. [18] introduce an operator which returns a high total value only if both the recommendation and the reputation score are high. This is supposed to "amplify the discriminating power" [18]. Similarly to the approaches employing the switching method, the systems based on the mixed method all combine recommender and reputation systems relying on the same data base.

Rec-Rep-Cascade. Constantinov et al. [12] propose a rec-rep-cascade hybridization using different data bases. First, a recommender system determines a product the customer is supposed to be particularly interested in. Then, a reputation system depicts information relevant for the assessment of the trustworthiness of the sellers offering the product. Because of the limited size of the platform, the reputation information is limited to only one seller. On a larger platform, however, there would be many providers offering the same item. Then, the reputation system helps determine the most trustworthy one. In contrast, Abdel-Hafez et al. [1] consider a cascade hybridization of a recommender and a reputation system relying on the same data base. They enhance a recommender system's output by re-sorting the top- M recommendations based on their reputation values. Thus, only the top- M items according to the recommender system enter the second step of the cascade. Finally, the top- N ($N < M$) items of the re-sorted list are recommended to the user. Similarly, the idea of Ku and Tai [21] is to provide one or more item recommendations to the user at first. Then, the user is supposed to take a look at the reputation of the items and probably also at their rating distributions. As opposed to the other publications discussed in this section, the authors do not propose a new system but conduct a study on the effects of recommendation information and reputation information on buying intentions.

Rep-Rec-Cascade. Tserpes et al. suggest that “providers that systematically fail to comply with their obligations against the consumers will be isolated” [40] and thus to use reputation data as a pre-filtering mechanism prior to the recommendation process. Guo et al. [16] realize this by extending their document recommendation system with a reputation component keeping track of the reputation values of the users according to their activities and the acceptance rates of the documents shared by them. If the reputation value of a user drops below a particular group’s threshold, he can no longer access this group and his sharing activities are no longer considered in any recommendations. The recommender system introduced by Yu et al. [49] also excludes users with negative reputation values from the item recommendation process.

Augmentation. In contrast to the proposals discussed so far, the following approaches integrate the reputation data directly into the computation process of the recommender system. In all of them, the recommender component is concerned with items whereas the reputation values belong to users (e.g. sellers, providers). Qian et al. [34] as well as Tang et al. [39] employ the users’ reputation values to control the importance of the ratings in the matrix factorization process of their product recommenders. Cimini et al. [11] use the reputations of news item creators to replace or at least supplement the consideration of similarity values in the collaborative filtering calculations of their news recommender system. The news item creators’ reputation values are based on the number of users that have liked the respective news items. Similarly, Su et al. [37] use the reputations of web service users to enhance the similarity calculations within the collaborative filtering process of their quality of service prediction approach. The reputation values are calculated according to the beta-family of probability density functions [46]. Liu et al. [25] suggest to overcome the limitation of content-based filtering systems of recommending only items similar to the ones a user has previously liked by augmenting the user’s rating matrix with his group’s preference scores. The group’s preference score for an item is derived according to the reputation of the users who have pushed the particular item. A user’s reputation value, in turn, is based on the amount of articles pushed by him as well as the number of users following these articles. Liu et al. [23, 24] also use this idea for a document recommender based on the similarity between the topic interests of a community and the target documents. The topic interests are determined according to the topics collected by the community and the reputation of the users who have collected them. The users’ reputation values, in turn, are based on the number of push interactions indicating that other users found a document helpful.

3.4 Limitations of the Literature Review

Overall, our review serves as a comprehensive summary of the state of the art of reputation-enhanced recommender systems and can, as such, be used for understanding or new research. Even though we ensured a high quality of the review by relying on well-recognized guidelines, there are some limitations to discuss.

Analyzing the literature according to a newly developed taxonomy carries the risk that the insights gained might be of little value if the dimensions are poorly defined. To mitigate this potential shortcoming, we derived the data base dimension from commonly accepted principles regarding recommender and reputation systems and kept its values generalized. The hybridization method dimension is based on published research as it adapts the values of Burke’s [10] work on hybrid recommender system. The same applies to the recommendation approach and evaluation dimensions, which rely on the remarks of Adomavicius and Tuzhilin [3] and Prat et al. [32], respectively.

Another possible limitation is that relevant literature might not be included in our search results. Since we chose five of the most relevant databases, used them with very general search phrases, and conducted forward as well as backward searches, however, it is unlikely that we missed many relevant publications.

4 Future Research Directions

The analysis of the literature yields several observations. First of all, the publication years of the papers suggest a growing interest in reputation-enhanced recommender systems especially since 2011. Turning to the contents of the existing work, important insights on the state of the art of the research stream can be gained by assigning the publications to the different hybridization approaches, whose dimensions and categories are introduced as the most important ones of our taxonomy in Sect. 3.2.

Table 3. Publications classified according to the hybridization approach dimensions.

| | Different data bases | Same data base |
|-----------------|-------------------------|----------------|
| Weighted | [4, 13, 29, 43] | [1, 2, 44] |
| Switching | | [7–9] |
| Mixed | | [18, 47, 48] |
| Rec-rep-cascade | [12] | [1, 21] |
| Rep-rec-cascade | [16, 40, 49] | |
| Augmentation | [11, 23–25, 34, 37, 39] | |

As Table 3 shows, each hybridization method is covered by at least three proposals. Each category of the data base dimension is covered by multiple publications as well. However, not all combinations of data base and hybridization method categories have been addressed so far. Our search results do not contain any proposals regarding the switching and the mixed hybridization with different data bases as well as the rep-rec-cascade and the augmentation hybridization with the same data base. Therefore, the first future research direction is to investigate whether the missing combinations are applicable to meaningful use cases

and whether corresponding systems lead to performance improvements. Abdel-Hafez et al. [1], for example, justify their decision to focus on the rec-rep-cascade hybridization instead of the rep-rec-cascade hybridization with the assumption that personalized recommender-generated lists would be more accurate than non-personalized reputation-generated lists and therefore should be used as the primary candidate recommendation list. Although this assumption is intuitively understandable, its validity is still worth investigating.

Focusing on the evaluation dimensions, Table 2 (cf. Sect. 3.3) reveals that some of the publications are not thoroughly evaluated by comparing them to related work or not evaluated at all. Those publications that have actually been evaluated all show improvements in terms of the employed metrics, which supports the implicit claim of this paper that enhancing recommender systems with reputation data leads to better recommendation performance. Nevertheless, some of the evaluations are based on fictional and overly simplistic scenarios. Although demonstrations, as these light-weight forms of evaluation should rather be denoted, can show the feasibility and meaningfulness of the proposals, the second future research direction is to investigate how the systems that have been evaluated insufficiently or not at all actually compare to related baseline recommendation techniques using real-world data.

The ultimate goal regarding the evaluation dimensions, and thus the third future research direction, is to not only compare the developed systems to baseline recommendation techniques but also among one another. To determine the best proposal for a specific use case, it is necessary to make the respective evaluations comparable by always using the same metrics and data sets. This is far from being an easy task because not all of the existing approaches are described in sufficient detail to be able to re-implement them and compare them to one another.

5 Conclusion

The marginal improvements that may be achieved from further optimizing highly sophisticated recommender algorithms have motivated scholars to broaden the horizon of recommender systems research and integrate relevant concepts from related fields. Since trust and reputation systems show substantial connections to recommender systems, there have been attempts to consider trust relationships in the recommendation process. However, personal trust links are only available in small numbers on modern online platforms because these are typically characterized by short-term interactions. As the concept of reputation is closely linked to trust but fits the peculiarities of modern online platforms better, this paper focused on the integration of reputation data instead of trust relationships. In fact, the corresponding research stream of reputation-enhanced recommender systems has attracted considerable attention in recent years. Therefore, our main goal was to provide a comprehensive survey of the approaches proposed so far. At first, we identified existing work in a systematic and exhaustive search process. Then, in order to relate the publications to one another, we developed a dedicated taxonomy based on commonly accepted principles and published research.

Comparing the proposals according to the taxonomy resulted in a structured overview of the state of the art of the research stream.

On the one hand, our results help stimulate further innovation in reputation-enhanced recommender systems. Future research is not only needed to close or explain the identified gaps but also to improve the existing proposals. After all, there still is constant innovation in the respective research fields of recommender and reputation systems, which is why new hybridization approaches are needed and expected as well. On the other hand, this paper also serves as an important basis for the further exchange of ideas between both communities. For example, future research efforts could investigate the opposite of our approach: how recommender systems may be used to enhance reputation systems.

Acknowledgments. The research leading to these results was supported by the “Bavarian State Ministry of Education, Science and the Arts” as part of the FORSEC research association.

References

1. Abdel-Hafez, A., Tang, X., Tian, N., Xu, Y.: A reputation-enhanced recommender system. In: Luo, X., Yu, J.X., Li, Z. (eds.) ADMA 2014. LNCS, vol. 8933, pp. 185–198. Springer, Cham (2014). doi:[10.1007/978-3-319-14717-8_15](https://doi.org/10.1007/978-3-319-14717-8_15)
2. Abdel-Hafez, A., Xu, Y., Tian, N.: Item reputation-aware recommender systems. In: Proceedings of the 16th International Conference on Information Integration and Web-based Applications & Services (iiWAS), pp. 79–86 (2014)
3. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. *IEEE Trans. Knowl. Data Eng.* **17**(6), 734–749 (2005)
4. Alotaibi, S., Vassileva, J.: Trust-based recommendations for scientific papers based on the researcher’s current interest. In: Lane, H.C., Yacef, K., Mostow, J., Pavlik, P. (eds.) AIED 2013. LNCS, vol. 7926, pp. 717–720. Springer, Heidelberg (2013). doi:[10.1007/978-3-642-39112-5_96](https://doi.org/10.1007/978-3-642-39112-5_96)
5. Arazy, O., Sana, I., Shapira, B., Kumar, N.: Social relationships in recommender systems. In: Proceedings of the 17th Workshop on Information Technologies & Systems (WITS) (2007)
6. Artz, D., Gil, Y.: A survey of trust in computer science and the semantic web. *Web Semant. Sci. Serv. Agents World Wide Web* **5**(2), 58–71 (2007)
7. Bedi, P., Agarwal, S.K.: SRPRS: situation-aware reputation based proactive recommender system. *J. Inf. Assur. Secur.* **8**(4), 220–229 (2013)
8. Bedi, P., Agarwal, S.K., Jindal, V., Richa: MARST: Multi-agent recommender system for e-tourism using reputation based collaborative filtering. In: Madaan, A., Kikuchi, S., Bhalla, S. (eds.) Proceedings of the 9th International Workshop on Databases in Networked Information Systems (DNIS 2014). LNCS, vol. 8381, pp. 189–201. Springer, Heidelberg (2014). doi:[10.1007/978-3-319-05693-7_12](https://doi.org/10.1007/978-3-319-05693-7_12)
9. Bedi, P., Agarwal, S.K., Sharma, S., Joshi, H.: SAPRS: situation-aware proactive recommender system with explanations. In: Proceedings of the 3rd International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp. 277–283 (2014)

10. Burke, R.: Hybrid recommender systems: survey and experiments. *User Model. User-Adap. Interact.* **12**(4), 331–370 (2002)
11. Cimini, G., Medo, M., Zhou, T., Wei, D., Zhang, Y.C.: Heterogeneity, quality, and reputation in an adaptive recommendation model. *Eur. Phys. J. B* **80**(2), 201–208 (2011)
12. Constantinov, C., Mocanu, A., Popescu, E.: Online auctioning and recommendations: the eBidLand platform. In: *Proceedings of the 16th International Conference on System Theory, Control and Computing (ISCTCC)*, pp. 1–6 (2012)
13. Cui, L., Ou, P., Lu, N., Zhang, G.: A comprehensive trust-based item evaluation model for recommendation in social network. In: *Proceedings of the 21st IEEE Symposium on Computers and Communication (ISCC)*, pp. 1090–1096 (2016)
14. Dellarocas, C.: Reputation mechanisms. In: Hendershott, T. (ed.) *Economics and Information Systems. Handbooks in Information Systems*, pp. 629–660. Elsevier, Amsterdam (2006)
15. Goldberg, D., Nichols, D., Oki, B.M., Terry, D.: Using collaborative filtering to weave an information tapestry. *Commun. ACM* **35**(12), 61–70 (1992)
16. Guo, F., Li, S., Lin, K.: An auto-regulative document recommendation system based on P2P networks. In: *Proceedings of the 3rd International Conference on Natural Computation (ICNC 2007)*, pp. 467–471 (2007)
17. Herbig, P.A., Kramer, H.: The effect of information overload on the innovation choice process. *J. Consum. Mark.* **11**(2), 45–54 (1994)
18. Jøsang, A., Guo, G., Pini, M.S., Santini, F., Xu, Y.: Combining recommender and reputation systems to produce better online advice. In: Torra, V., Narukawa, Y., Navarro-Arribas, G., Megías, D. (eds.) *MDAI 2013. LNCS*, vol. 8234, pp. 126–138. Springer, Heidelberg (2013). doi:[10.1007/978-3-642-41550-0_12](https://doi.org/10.1007/978-3-642-41550-0_12)
19. Jøsang, A., Ismail, R., Boyd, C.: A survey of trust and reputation systems for online service provision. *Decis. Support Syst.* **43**(2), 618–644 (2007)
20. Jøsang, A., Quattrociochi, W., Karabeg, D.: Taste and trust. In: Wakeman, I., Gudes, E., Jensen, C.D., Crampton, J. (eds.) *IFIPTM 2011. IAICT*, vol. 358, pp. 312–322. Springer, Heidelberg (2011). doi:[10.1007/978-3-642-22200-9_25](https://doi.org/10.1007/978-3-642-22200-9_25)
21. Ku, Y.C., Tai, Y.M.: What happens when recommendation system meets reputation system? The impact of recommendation information on purchase intention. In: *Proceedings of the 46th Hawaii International Conference on System Sciences (HICSS)*, pp. 1376–1383 (2013)
22. Levy, Y., Ellis, T.J.: A systems approach to conduct an effective literature review in support of information systems research. *Informing Sci. J. (Informing Sci. Int. Journal Emerg. Transdiscipline)* **9**, 181–212 (2006)
23. Liu, D.R., Chen, Y.H., Huang, C.K.: QA document recommendations for communities of question-answering websites. *Knowl. Based Syst.* **57**, 146–160 (2014)
24. Liu, D.-R., Huang, C.-K., Chen, Y.-H.: Recommending QA documents for communities of question-answering websites. In: Selamat, A., Nguyen, N.T., Haron, H. (eds.) *ACHIIDS 2013. LNCS*, vol. 7803, pp. 139–147. Springer, Heidelberg (2013). doi:[10.1007/978-3-642-36543-0_15](https://doi.org/10.1007/978-3-642-36543-0_15)
25. Liu, D.R., Liou, C.H., Peng, C.C., Chi, H.C.: Hybrid content filtering and reputation-based popularity for recommending blog articles. *Online Inf. Rev.* **38**(6), 788–805 (2014)
26. Loepp, B., Herrmann, K., Ziegler, J.: Blended recommending: integrating interactive information filtering and algorithmic recommender techniques. In: *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI)*, pp. 975–984 (2015)

27. Lops, P., de Gemmis, M., Semeraro, G.: Content-based recommender systems: state of the art and trends. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds.) *Recommender Systems Handbook*, pp. 73–105. Springer, Boston (2011)
28. Massa, P., Avesani, P.: Trust-aware collaborative filtering for recommender systems. In: Meersman, R., Tari, Z. (eds.) *OTM 2004. LNCS*, vol. 3290, pp. 492–508. Springer, Heidelberg (2004). doi:[10.1007/978-3-540-30468-5_31](https://doi.org/10.1007/978-3-540-30468-5_31)
29. McNally, K., O'Mahony, M.P., Smyth, B.: A comparative study of collaboration-based reputation models for social recommender systems. *User Model. User-Adap. Interact.* **24**(3), 219–260 (2014)
30. Okoli, C., Schabram, K.: A guide to conducting a systematic literature review of information systems research. In: *Sprouts: Working Papers on Information Systems*, vol. 10(26) (2010)
31. Prassas, G., Pramataris, K.C., Papaemmanouil, O.: Dynamic recommendations in internet retailing. In: *Proceedings of the 9th European Conference on Information Systems (ECIS)* (2001)
32. Prat, N., Comyn-Wattiau, I., Akoka, J.: A taxonomy of evaluation methods for information systems artifacts. *J. Manage. Inf. Syst.* **32**(3), 229–267 (2015)
33. Pu, P., Chen, L., Hu, R.: Evaluating recommender systems from the user's perspective: survey of the state of the art. *User Model. User-Adap. Interact.* **22**(4–5), 317–355 (2012)
34. Qian, F., Zhao, S., Tang, J., Zhang, Y.: SoRS: social recommendation using global rating reputation and local rating similarity. *Physica A Stat. Mech. Appl.* **461**, 61–72 (2016)
35. Sanger, J., Richthammer, C., Pernul, G.: Reusable components for online reputation systems. *J. Trust Manage.* **2**(5), 1–21 (2015)
36. Smyth, B., Briggs, P., Coyle, M., O'Mahony, M.: Google shared. A case-study in social search. In: Houben, G.-J., McCalla, G., Pianesi, F., Zancanaro, M. (eds.) *UMAP 2009. LNCS*, vol. 5535, pp. 283–294. Springer, Heidelberg (2009). doi:[10.1007/978-3-642-02247-0_27](https://doi.org/10.1007/978-3-642-02247-0_27)
37. Su, K., Xiao, B., Liu, B., Zhang, H., Zhang, Z.: TAP: a personalized trust-aware QoS prediction approach for web service recommendation. *Knowl. Based Syst.* **115**, 55–65 (2017)
38. Su, X., Khoshgoftaar, T.M.: A survey of collaborative filtering techniques. *Adv. Artif. Intell.* **2009**(12), 1–19 (2009)
39. Tang, J., Hu, X., Gao, H., Liu, H.: Exploiting local and global social context for recommendation. In: *Proceedings of the 23rd International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 2712–2718 (2013)
40. Tserpes, K., Aisopos, F., Kyriazis, D., Varvarigou, T.: A recommender mechanism for service selection in service-oriented environments. *Future Gener. Comput. Syst.* **28**(8), 1285–1294 (2012)
41. Victor, P., de Cock, M., Cornelis, C.: Trust and recommendations. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds.) *Recommender Systems Handbook*, pp. 645–675. Springer, Boston (2011)
42. Vom Brocke, J., Simons, A., Niehaves, B., Reimer, K.: Reconstructing the giant: on the importance of rigour in documenting the literature search process. In: *Proceedings of the 17th European Conference on Information Systems (ECIS)* (2009)
43. Wang, Y., Tong, X., He, Z., Gao, Y., Wang, K.: A task recommendation model for mobile crowdsourcing systems based on dwell-time. In: *Proceedings of the IEEE International Conferences on Big Data and Cloud Computing (BDCloud), Social Computing and Networking (SocialCom), Sustainable Computing and Communications (SustainCom)*, pp. 170–177 (2016)

44. Wang, Y., Yin, G., Cai, Z., Dong, Y., Dong, H.: A trust-based probabilistic recommendation model for social networks. *J. Netw. Comput. Appl.* **55**, 59–67 (2015)
45. Webster, J., Watson, R.T.: Analyzing the past to prepare for the future: writing a literature review. *MIS Q.* **26**(2), 13–23 (2002)
46. Whitby, A., Jøsang, A., Indulska, J.: Filtering out unfair ratings in bayesian reputation systems. In: Proceedings of the 7th International Workshop on Trust in Agent Societies at the 3rd International Joint Conference on Autonomous Agents and Multi Agent Systems (AAMAS), pp. 106–117 (2004)
47. Yan, Z., Zhang, P., Deng, R.H.: TruBeRepec: a trust-behavior-based reputation and recommender system for mobile applications. *Pers. Ubiquit. Comput.* **16**(5), 485–506 (2012)
48. Yazidi, A., Granmo, O.C., Oommen, B.J., Gerdes, M., Reichert, F.: A user-centric approach for personalized service provisioning in pervasive environments. *Wirel. Pers. Commun.* **61**(3), 543–566 (2011)
49. Yu, Z., Song, W.W., Zheng, X., Chen, D.: A recommender system model combining trust with topic maps. In: Ishikawa, Y., Li, J., Wang, W., Zhang, R., Zhang, W. (eds.) APWeb 2013. LNCS, vol. 7808, pp. 208–219. Springer, Heidelberg (2013). doi:[10.1007/978-3-642-37401-2_22](https://doi.org/10.1007/978-3-642-37401-2_22)

Self-reported Verifiable Reputation with Rater Privacy

Rémi Bazin¹(✉), Alexander Schaub¹, Omar Hasan², and Lionel Brunie²

¹ Department of Computer Science, École Polytechnique, 91120 Palaiseau, France
bazin.remi@gmail.com

² University of Lyon, CNRS INSA-Lyon, LIRIS, UMR5205, 69621 Lyon, France

Abstract. Reputation systems are a major feature of every modern e-commerce website, helping buyers carefully choose their service providers and products. However, most websites use centralized reputation systems, where the security of the system rests entirely upon a single Trusted Third Party. Moreover, they often disclose the identities of the raters, which may discourage honest users from posting frank reviews due to the fear of retaliation from the ratees. We present a reputation system that is decentralized yet secure and efficient, and could therefore be applied in a practical context. In fact, users are able to retrieve the reputation score of a service provider directly from it in constant time, with assurance regarding the correctness of the information obtained. Additionally, the reputation system is anonymity-preserving, which ensures that users can submit feedback without their identities being associated to it. Despite this anonymity, the system still offers robustness against attacks such as ballot-stuffing and Sybil attacks.

1 Introduction

Reputation systems are very common on the Internet as they help the users learn about the quality of a product, document or other items of interest. Examples of reputation systems include the systems used on [Amazon](#) or [Taobao](#). All these examples are based on centralized reputation systems, which implies that their security relies on the assumption that the underlying server is honest and secure.

Decentralized protocols (e.g. BitTorrent [1], Bitcoin [20]) have emerged for mainly two reasons: releasing the central server from resource consuming tasks to distribute these among the peers, or getting rid of the security dependency on the central server. Indeed, should a reputation protocol be centralized, a privacy disclosure such as the AOL search data leak in 2006 always remains a possible threat [4]. Neither are we safe from sponsoring i.e. increasing a certain entity's reputation in exchange for some fee – be it a public practice or a hidden activity. Although we usually trust well known entities such as the ones quoted above to behave honestly, we want to get rid of these trust requirements for a wider range of systems. These reasons justify our need for a decentralized scheme.

Another feature that we wish to provide is to preserve the anonymity of the raters. This choice is motivated by studies, such as the one on eBay [25], that show how sellers might discriminate against customers based on their previous feedback. Two solutions arise to achieve this goal. The first one is to preserve the confidentiality of the rating values while making the list of raters for a specific vendor public. The other one is to hide everything but the aggregated reputation score by making the feedback entries unlinkable with the transactions and the identities of the customers. We will choose the latter proposition.

The protocol that we propose is also resistant against Sybil attacks [14]. These attacks consist of multiple fake identities or bots controlled by a single malicious user acting like legitimate clients in order to do ballot stuffing and send a high amount of either positive feedback values (self-promotion) or negative ones (bad-mouthing). We rely on blind signatures in our proposed protocol to achieve resistance against bad-mouthing attacks. Our scheme will also incorporate *tokens* as a way to prevent self-promotion.

A key contribution of the protocol is that the ratee himself stores the reputation values, yet the integrity of the reputation score is maintained. The querier is able to verify the integrity of the reputation score. This enables constant time retrieval.

The target application of our reputation system will be e-commerce: we will consider Service Providers (SPs) who want to sell goods, and clients who wish to buy the goods. The SPs will be the ratees i.e. the ones who receive ratings, whereas the clients will be the raters.

Our protocol fits into this e-commerce environment, while being both anonymity preserving and decentralized. It is based on Merkle trees [19], blind signatures [10] and non-interactive zero-knowledge proofs, and will be efficient (constant-time) when retrieving reputation.

The rest of the paper is organized as follows. Section 2 provides an overview of the state of the art concerning privacy-preserving reputation systems. Section 3 illustrates the model for the environment in which our protocol is to be used, while Sect. 4 highlights the objectives of our work. Our construct of *tokens* is described in Sect. 5. Section 6 describes the core protocol. An analysis of the protocol with regards to the previously defined objectives is presented in Sect. 7. Finally, we conclude in Sect. 8.

2 Related Work

Many privacy-preserving reputation systems have already been proposed in the literature. However, some of the papers in this domain use theoretical adversarial models that may not be appropriate for the real-world: for instance, the assumption that there will be no collusion among malicious peers is not realistic (e.g. [23]). Some other works are nonetheless more secure and resistant to small groups of malicious peers: the StR^M algorithm by Dimitriou et al. [12] and the Malicious k-shares protocol by Hasan et al. [17] are examples of such schemes. However, these protocols are rather confidentiality-preserving than privacy-preserving in the sense that they do not hide the list of users who participated in the rating.

Hence, we will focus on anonymity-preserving methods that completely hide the identity of the raters. Protocols of such type do already exist, but each one of them has some attributes that we want to avoid. The works of Androulaki et al. [3] and Petrlc et al. [24], for example, are instances of pseudonym based schemes. Nonetheless, these two require a centralized Trusted Third Party (TTP), and are thus not truly decentralized. The works of Anceaume et al. [2] and Lajoie-Mazenc et al. [18] on the other hand are more decentralized, but they rely on properties that we want to avoid: the first one prevents Sybil attacks by charging a fee, and in the second one, accredited signers are required to make resource heavy calculations for each rating of each SP. Even though this last recent contribution is very close to what we are looking for, we believe that our protocol succeeds better in distributing the computational costs among the different peers, notably by assigning the feedback records management to the specific service provider that is concerned. The work of Schaub et al. [26] is also decentralized and uses a blockchain to attain some similar objectives, but ballot-stuffing is still possible should the service provider be willing to pay fees for some additional custom feedback. Finally, the paper by Bethencourt et al. [6] illustrates a promising scheme based on signatures on published data. While this protocol is very interesting and secure, it has monotonic feedback, which allows an attacker to take advantage of his old good reputation without being affected by any new dissatisfaction that his recent activity might cause. We do however take inspiration from this work and use the same kind of zero knowledge proofs.

3 Model

The model we choose for our protocol is consistent with that of an e-commerce environment: we will consider a simple two-sided model where there are Service Providers (SPs) who sell goods and clients who buy them. We will only consider ratings provided by clients and destined for SPs.

Each transaction between a SP and a client should provide the client with a way to later post a feedback about the SP. The triggering event that enables a feedback to be sent should be the financial transaction itself. Moreover, only a single feedback may be valid per user per SP to prevent ballot-stuffing.

In our scheme, to maintain unlinkability between the client and the feedback, the feedback record would need to be sent by the client a certain amount of time after the transaction. This time-out may vary with the pace at which other clients' feedback is sent to the corresponding SP. Each user may be able to change this time-out privacy parameter according to his needs.

4 Objectives

Our objectives are to design a reputation system that is efficient, anonymity-preserving, decentralized and robust. The main novelty we propose is to ensure all of these contrasting properties in a single protocol. In the literature, we only find protocols that fulfill a subset of these attributes ([2, 3, 6, 12, 17, 18, 23, 24, 26]), as discussed in the related work section.

4.1 Efficiency

Clients may need to browse through the list of a large number of SPs before choosing to transact with a specific SP. Therefore, the ability to quickly retrieve reputation values without overwhelming the network nor requiring excessive computation and latency is an essential requirement in a reputation system. The protocol must therefore ensure that it is efficient for the clients to retrieve the reputation value of a SP. As a matter of fact, we want to have a constant-time reputation retrieval procedure, which is uncommon in decentralized systems in the literature. Efficiency on the user side is a key advantage of the protocol that we aim to propose.

4.2 User Anonymity Preservation

Anonymity is achieved by maintaining two types of unlinkability:

1. **Transaction – rating unlinkability.** The transaction itself may disclose the identity of the client, because of his shipping address for instance. This first kind of unlinkability consists in separating the transaction and the rating, which should be anonymous. However, we still want the transaction to enable the rating.
2. **Rating – rating unlinkability.** It has been shown ([4,21]) that this second kind of unlinkability – between several ratings of a unique user – is also primordial to preserve the anonymity of the users.

We do not aim to hide the identities of the SPs in our protocol though. This means that they will be linkable to all their previous ratings. In the e-commerce context, this behavior is indeed often desirable.

4.3 Decentralization

Our objective is also to design a decentralized scheme. Security and privacy are better preserved in a decentralized environment in the sense that one does not have to rely on a single central entity that can become a single point of failure.

We do not exclude a Certification Authority if we want to use it as a way to prevent Sybil attacks. This authority shall nonetheless not have any other role in the protocol than giving certificates. Moreover, it may be offline most of the time since the only requirement is that it correctly delivers certificates.

4.4 Robustness

In our scheme, we place ourselves in a situation where peers may be malicious and colluding together. However, a majority of the peers that we call trackers is considered to be non-colluding honest-but-curious.

5 Tokens – Security Against Sybil Attacks

In this section, we give a highlight of what the tokens are - a key building block that we use in our protocol. Their utility is to prevent Sybil attacks, and more precisely self-promotion, as highlighted in the introduction. They might be used in other contexts for protection against Sybil attacks in general. In that sense, their goal is to distinguish bots from real users.

In our protocol, the tokens are to uniquely identify a couple client/SP. Only the corresponding client should be able to generate such a token, and yet this token should not disclose any information related to his identity. One should therefore be able to tell if two tokens are issued by the same client or not, even though the anonymity of that client is preserved.

We also don't want other people to be able to reuse the token once the corresponding feedback record has expired. To achieve this, we include in each token a commitment to the one-time public key K that is used in the feedback records (see Sect. 6.2).

We design our tokens with a certificate-based implementation that is described below.

5.1 Certificate-Based Method

For this method of generating tokens, we assume the existence of a Certification Authority (CA), at least at some point. This authority might however go offline after delivering the certificates since only these ones are used. We leave the criteria required for admission up to the implementation.

In order to have identity-based tokens that are unlinkable with the identity of the user himself, we will use non-interactive zero-knowledge proofs of knowledge (NIZKs). The role of the NIZK proof is to check the hidden credentials of the client (both their integrity and the validity of the certificate from the CA) and to assert that the plaintext value *value* that is included in the token is uniquely identifying the client and the SP. Additionally, it should also contain a signature by the client on the one-time public key K of the feedback records (see Sect. 6.2) so as to prevent any subsequent use of the token.

5.2 Formalization

We denote $\text{CRED.VERIFY}(pk, sk)$ to refer to the verification of a public and private key pair, $\text{CERT.VERIFY}(cert, pk)$ for the verification of a certificate *cert* about a public key pk , and $\text{SIG.VERIFY}(sign, M, pk)$ for the verification of a signature *sign* on the data M using the public key pk . $\text{TOKENIZE}(v_{SP}, sk)$ will be a procedure that creates a token value uniquely identifying the SP v_{SP} and the client whose private key is sk .

Using the notation introduced by Camenisch and Stadler [9], we want to construct the following proof:

$$\text{NIZK} \left\{ \begin{array}{l} pk_C, sk_C, cert : \\ \text{CRED.VERIFY}(pk_C, sk_C) \\ \wedge \text{CERT.VERIFY}(cert, pk_C) \\ \wedge \text{SIG.VERIFY}(sign, K, pk_C) \\ \wedge [value = \text{TOKENIZE}(v_{SP}, sk_C)] \end{array} \right\} \quad (1)$$

where the hidden variables pk_C , sk_C and $cert$ would respectively be the public key of the client, his private (secret) key, and the certificate from the CA validating his public key. The external values $sign$ and $value$ should be given alongside with the zero-knowledge proof. They are respectively a signature on the one-time public key K and the unique identifier for the couple SP/client. v_{SP} should be a unique and publicly-known identifier for the SP that is involved.

An implementation of such a NIZK proof is detailed in the technical report that extends this paper [5]. The NIZK model proposed by Groth et al. [16] is at the core of that implementation.

6 Description of the Protocol

6.1 Outline

Our protocol involves three kinds of nodes, as listed below and as shown in Fig. 1.

- **Clients:** They are the ones who buy goods. Every user can be a client, assuming that they can produce tokens.
- **Service Providers (SPs):** They are the ones who sell goods. They are publicly registered. A SP is in charge of saving all the feedback records that are related to it. These form a “local blockchain” which is signed by the trackers, and made publicly available by the SPs.

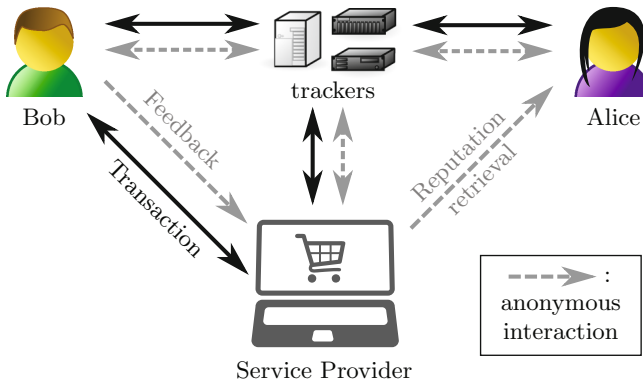


Fig. 1. Overview of the different entities and some primitive operations

- **Trackers:** They are a group of servers who are in the system mainly to control the good behavior of the SPs. Their role is therefore limited to the security and robustness of the scheme. We will minimize their involvement in the protocol in terms of resource usage.

The term *peer* will denote a computing unit that may be either of the three kinds of nodes above.

Below are brief summaries of the primitive operations in our reputation protocol, which are described in more detail in Sect. 6.3

- **Reputation retrieval:** The client obtains the reputation value of a SP from the SP itself, and verifies the trackers' signature.
- **Transaction:** The client asks for a blind signature from the SP while paying for his purchase. This will enable him to post a feedback record later. He also anonymously declares his purchase.
- **Sending feedback:** The client generates a feedback record from a token, the previous blind signature and his feedback value – which may also contain a comment. He sends it to the concerned SP who is required to include it in his next *block* of records (see Sect. 6.2).
- **Feedback aggregation:** The trackers periodically (e.g. once a day) sign the header of the next block, containing the current aggregated reputation value, so that the SPs can distribute it directly to the peers without any trust requirement between peers and SPs.

6.2 Setup

Trackers. What we will call *trackers* are a group of several servers whose aim is to guarantee the security of the scheme. They fulfill this task by providing the following public information, which they can provide along with a time-stamp and a signature:

- The list l_t of all the current trackers.
- A hash table b_t^1 containing proofs of malicious behavior and/or proofs of intentional withdrawal of old trackers.
- A hash table b_t^2 containing for each SP a list – which is possibly empty – of proofs of bad behavior.

Anyone should not be able to become a tracker since the corruption of a majority of them threatens the security of the scheme. However, even if a few become corrupted, the security is still upheld as long as a majority of them is honest. Assuming that they have divergent interests (to avoid collusion), competition and fear of fraud discovery are good safeguards.

Feedback Records. A feedback record is comprised of a tuple $(d, v, c, \mathbf{t}, K, s_1, s_2)$ containing:

- d : Date of publication
- v : Feedback value
- c : Feedback comment (optional, may be empty)
- t : Token (see Sect. 5)
- K : A one-time public key (part of a signature key pair)
- s_1 : (Blind) Signature of the SP on K
- s_2 : Signature on (d, v, c, t, K, s_1) , verifiable with K

SP – Persistence of the Records. The SP is in charge of maintaining the data of its records, meaning the records that rate him. This is a fair task allocation since: the more feedback records a SP has, the more known he is and therefore the more computational resources we may reasonably ask him to deliver.

The records are to be kept in a special list of data blocks where each block contains the records data for a given time period T . T must not be too long (for adaptability to new feedback) nor too short (for efficiency reasons). We will take the compromise $T = 1$ day to simplify the description. Another parameter also drives the temporal aggregation function: the number n_t of periods – days – during which a given feedback record is valid. For a living duration of the feedback records of one year, for instance, we would have something like $n_t = 365$. In other words, it is the number of blocks that account for the current overall reputation value. The length of the list of blocks that the SP should save and publish should be $n_t + 1$ for verification purposes. Once a new block is added, the oldest one is discarded from the list, provided that the SP is at least $n_t + 1$ days old.

Each block is a tuple $(d, v_T, v_{tot}, h, s_3, data)$ where:

- d : Date of publication
- v_T : Aggregated reputation value over the latest period T
- v_{tot} : Aggregated reputation value over the period $n_t T$
- h : Hash of (SP, h', r_1, r_2, r_3) with SP being the identity of the SP, h' the hash of $data$ and r_1, r_2 and r_3 the root labels of the three Merkle trees $\mathcal{T}_1, \mathcal{T}_2$ and \mathcal{T}_3 that are detailed below
- s_3 : Signature of the trackers on (d, v_T, v_{tot}, h)
- $data$: All the feedback records for this period T

This block is designed so that it can be sent without its $data$ element for any client to be able to retrieve and verify the current aggregated reputation value (with v_{tot}, s_3 and d).

In addition to these blocks of data, the SPs are required to maintain three Merkle trees. The first one \mathcal{T}_1 is to contain all the one-time public keys K that have been used so far in the blind signature scheme and published in feedback records. The second one \mathcal{T}_2 gathers all the currently used identities (i.e. token values). The third and last one \mathcal{T}_3 contains the identities of all the clients who made at least one financial transaction with the SP, regardless of whether or not they posted a feedback record. These identities are tokens that have been generated based on a derived version of the SP domain name identifier SP , along

with the date of the last transaction, committed inside the token. In this last Merkle tree, each node will also contain the number of leaves – i.e. identities – beneath it in addition to the usual hash of its children. That way, any peer can quickly retrieve the total number of buyers from the root of the tree, and verify it with r_3 . The inspectors for the updates of this total number also benefit from the structure of this Merkle tree, because only the updated branches need to be verified.

6.3 Primitive Operations

Reputation Retrieval. Each peer who wants to know the reputation of a SP just has to ask this SP for its reputation and the SP is expected to send back the signed data. The querying peer can then check the signatures of the trackers and retrieve the aggregated reputation value, as well as ask the SP for the rest of the block which contains the feedback records, i.e. the comments and individual feedback values.

As anybody can ask the SP for his reputation, clients have the choice to either ask him directly or use an anonymous connection such as Tor [13]. If he is asked directly, the query is no longer anonymous, but it is faster.

The main reputation retrieval procedure on the client side is detailed in Algorithm 1 below, which aims at retrieving the reputation of a SP SP at date d . The returned value is a tuple $(v_{tot}, header, s_4)$ where v_{tot} is the aggregated reputation we want, while $header$ and s_4 may be used for further analysis and data retrieval.

Algorithm 1. Retrieve the reputation of a SP

```

procedure REPRET( $SP, d$ )
  if ( $d > \text{today}()$ )  $\vee$  ( $d < \text{today}() - n_t T$ ) then
    fail with Wrong date  $d$ 
  if  $\neg$ CONNECTTO( $SP$ ) then
    fail with Unable to connect to  $SP$ 
  ( $header, s_4$ )  $\leftarrow$  ASKBLOCKHEADER( $SP, d$ )
  ( $d', v_T, v_{tot}, h, s_3$ )  $\leftarrow$   $header$ 
  if ( $d \neq d'$ )  $\vee$   $\neg$ CHECKSIG( $SP, s_4, header$ ) then
    fail with Non-cooperative SP
  if  $\neg$ CHECKSIG(trackers,  $s_3, (d', v_T, v_{tot}, h)$ ) then
     $p \leftarrow$  (REPRET, ( $header, s_4$ ),  $\emptyset$ )
    Send ( $SP, p$ ) to the trackers
    fail with Bad behavior
  return ( $v_{tot}, header, s_4$ )

```

Transaction. The transaction proceeds in three steps:

1. The client generates a one-time couple of public and private keys for a signing scheme, the public key being called K . He asks the SP to blindly sign his public key K during the financial transaction (see Sect. 6.4).

2. He gives a token generated with his identity and a derived version \widetilde{SP} of the SP identifier SP to the SP, so that it is included in the Merkle tree \mathcal{T}_3 .

The client memorizes the keys and the blind signature so that he might use them later to publish a feedback value. In the corresponding Algorithm 2 executed by the client, SP is the SP with whom the client is to pay for a specific good, and $context$ contains information about this good and the purchase in general. It uses the blind signature Algorithm 4: BLINDSIG to do the financial transaction in itself (see Sect. 6.4). It also uses the token creation scheme CREATETOKEN that takes the identifier of the Service Provider and the date (to prevent reuse of the token) as arguments.

Algorithm 2. Make a transaction (client side)

```

procedure CTRANSACTION( $SP, context$ )
  ( $sk, K$ )  $\leftarrow$  KEYGEN()
   $s_{SP} \leftarrow$  BLINDSIG( $SP, K, context$ )
   $d \leftarrow$  today()
   $t \leftarrow$  CREATETOKEN( $v_{SP}||$ "transaction",  $d$ )
   $SP$ .TRANSACTIONTOKEN( $t, d$ )
return ( $sk, K, s_{SP}$ )

```

Sending Feedback. When a client wants to rate a SP, after having done a transaction with it, he proceeds as follows:

1. The client waits until the anonymity set of the SP satisfies him, which means until there are enough buyers for this client to remain sufficiently anonymous (k -anonymity with sufficiently large k).
2. He fills a feedback record with the public key K and the blind signature that were generated during the transaction, the value and the comment of the feedback itself, and a token (see Sect. 5). He then signs the record so that it can be verified with K .
3. He anonymously gives the record to the SP, and asks a signed commitment from the SP stating that he will include this feedback record in his next block.
4. He checks for its effective inclusion later on.

For the whole publication part, it is assumed that the client uses an anonymous connection. The wait duration before sending the feedback record may be randomized in order to prevent any relevant statistical analysis that would break anonymity. A deadline may be set for feedback dispatch as to limit the effect of rosy retrospection and prevent any "reputation lag attack".

Should the record not be included in the next block, the client can then send the signed commitment he received from the SP as well as the signed block which should have contained the record to the trackers. This data is in itself a proof of bad behavior which would then be appended inside the corresponding hash map entry in each tracker.

Should the SP even refuse to deliver the signed commitment when being sent the feedback record, the client also has the possibility to send his feedback record to the trackers so that they try themselves to get this commitment and send it back to the client. If the SP also refuses to them, the trackers build a proof of bad behavior based on that fact, which is signed by all the trackers – or at least a majority of them.

In the following Algorithm 3 that describes this procedure, the client calls SENDFB with the identity of the concerned SP, the tuple that was returned by a previous call to the procedure CTRANSACTION, the feedback value v to submit and a comment c that is possibly empty.

Algorithm 3. Send a feedback record

```

procedure SENDFB( $SP, (sk, K, s_{SP}), v, c$ )
  Wait if necessary before proceeding.
   $d \leftarrow \text{today}()$ 
   $\mathbf{t} \leftarrow \text{CREATETOKEN}(v_{SP}, K)$ 
   $s_2 \leftarrow \text{CREATESIG}(sk, (d, v, c, \mathbf{t}, K, s_{SP}))$ 
   $rec \leftarrow (d, v, c, \mathbf{t}, K, s_{SP}, s_2)$  ▷ Feedback record
   $\alpha \leftarrow \text{ANONYMOUSCONNECTION}(SP)$ 
   $\alpha.\text{SEND}(rec)$ 
   $C \leftarrow \alpha.\text{RECEIVECOMMITMENT}()$ 
  if  $\neg \text{CHECKSIG}(SP, C, (\text{RECEIVED}, rec))$  then
    Send  $rec$  to a few trackers
    if They don't send back some valid  $C$  then
      fail with Bad behavior
  Wait (schedule the following) for the next day or later
  repeat
     $\alpha \leftarrow \text{ANONYMOUSCONNECTION}(SP)$ 
     $(v_{tot}, header, s_4) \leftarrow \alpha.\text{REPRET}(SP, d + 1)$ 
     $hinfo \leftarrow \text{CHECKHASH}(SP, header, s_4)$ 
     $data \leftarrow \alpha.\text{DATRET}(SP, header, s_4, hinfo)$ 
  until Reputation retrieved or too many fails
  if  $rec \notin data$  then
     $p \leftarrow (\text{SFB}, (rec, C, header, s_4, hinfo, data), \emptyset)$ 
    Send  $(SP, p)$  to the trackers
    fail with Bad behavior
  
```

Feedback Aggregation. In order to minimize the workload of the trackers, we want them to collectively only sign the header of each SP's new block for each period, so that the clients asking for the reputation of a SP directly ask the SP instead of asking the trackers. Since they only sign one block per period per SP, it is possible to check its integrity afterwards, and maybe create a proof of bad behavior that will be validated thanks to this signature. Of course, we could also decide that the trackers verify it, in full or in part, before giving their signature.

The deterrence that are the proofs of bad behavior make it possible to increase the efficiency of the computation. Indeed, only a partial verification of the data should be sufficient to dissuade malicious SPs from misbehaving.

The trackers do need to check the hash h however, to ensure that this block header won't be used by another SP.

This scheme is designed so that the computation and verification of the reputation v_{tot} is made easier thanks to the daily values v_T . Indeed, to check a new v_T , one needs to go through all the records of the day. To check a new v_{tot} however, one should only need to take into account the previous total aggregated reputation (v_{tot}) of the day before, and the values v_T for the incoming and expiring days. Being able to calculate a reputation based on a previous reputation and aggregated new and old feedback values is the only requirement that we want for the aggregation formula. We leave the choice of this formula up to the implementation. A large number of frequently used aggregation formulas are consistent with the previous prerequisite, as is detailed in the extended version of this paper [5].

6.4 Blind Signatures

Many algorithms exist for blind signatures: from the most well-known and simple one based on RSA cryptography [15] to other more complex ones [8, 22]. Some anonymous e-cash schemes may also be derived to be used as blind signature schemes. This is the case for the untraceable electronic cash by D. Chaum et al. [11], which is actually only a singular case of the RSA blind signature.

We face the following issue in implementing blind signatures for our scheme: how can we ensure that the blind signature is executed simultaneously with the payment? Indeed, should one of the two procedures terminate before the other, the one or the other of the two parties can stop the trade in the middle and use the half-trade to his advantage. If the blind signature finishes before the payment for instance, the user is able to rate the SP without even doing the trade (blind signature rendered useless). If it happens after the payment, the SP would be able to refuse the signing, and therefore the feedback. Even though it might not be much of a problem in this case, since refusing signatures means less feedback and less reputation for the SP, we still want to propose an alternative solution.

We detail below the implementation of a modified BLS blind signature that fulfills our requirements. The protocol, formalized in Algorithm 4, comprises of the following steps:

1. The client asks the SP to give a signature over the commitment that, should the signature of the client over the financial transaction be published, he is to blindly sign a specific masked message $-G^r M$.
2. Once he does so, the client can then safely sign the financial transaction.
3. The client asks the blind signature, and uses the signed commitment as well as the system of the trackers in case of bad behavior.

Algorithm 4. RSA Blind signature (client side)

```

procedure BLINDSIG( $SP, K, context$ )
   $T \leftarrow SP.GETTRANSACTIONTOSIGN(context)$ 
   $(\mathcal{G}' = \mathcal{G}^e \in \mathbb{G}_1, \mathcal{H}' = \mathcal{H}^e \in \mathbb{G}_2) \leftarrow$  public key of  $SP$  for the blind signatures
   $r \leftarrow \text{random}(), \mathcal{M} \leftarrow \text{HASH}(K) \in \mathbb{G}_1$ 
   $\mathcal{M}' \leftarrow \mathcal{G}'^r \mathcal{M}$ 
   $\tilde{T} \leftarrow SP.GETCOMMITMENT(T, \mathcal{M}')$ 
  if  $\neg \text{CHECKSIG}(SP, \tilde{T}, (\text{BLINDCOMMIT}, \mathcal{M}', T))$  then
    fail with Wrong commitment from  $SP$ 
   $s_T \leftarrow \text{SIGNTRANSACTION}(T)$ 
   $\bar{m} \leftarrow SP.ASKBS(s_T)$ 
  if  $\bar{\mathcal{M}}^e \bmod n \neq \bar{m}$  then
    Send  $(T, \mathcal{M}', \tilde{T}, s_T)$  to a few trackers
    if They don't send back some valid  $\bar{m}$  then
      fail with Bad behavior
  return  $\mathcal{M}\mathcal{G}'^r$ 

```

7 Analysis

Below is a brief analysis covering each one of the objectives we set in Sect. 4. A detailed analysis may be found in the extended version of the paper [5].

7.1 Efficiency

Our main goal regarding the efficiency aspects was to have a quick and light way of retrieving the reputation of a SP. This objective is achieved in our protocol because this operation operates in constant time in both network usage and computing power. Table 1 highlights how small the computations are to retrieve the reputation of a SP and to perform the other client-side operations, in the likely case that the SP does not misbehave. The only downside is the linearity with N – the number of feedback records of the day – for sending feedback, but this may be reduced to a logarithmic complexity by taking advantage of the Merkle trees.

Assumptions for Table 1: Table 1 assumes SHA1 hashes (256 bits), ECDSA signatures based on the `ASN1::secp160r1` curve (336 bits) and AES-128 security level for pairings. The computational time has been measured on a computer with an Intel Core i3-5005U CPU, using the MIRACL and Crypto++ libraries.

Table 1. Computational cost of the different client operations

| | Computation time | Network payload | Network messages |
|----------------------|------------------|-----------------|------------------|
| Reputation retrieval | 2.7 ms | 140B | 2 |
| Transaction | 46.3 ms | 1574B | 4 |
| Sending feedback | 33 ms | 2443B+N*1735B | 12 |

A back-and-forth interaction is counted as two network messages. All the costs of the different operations have been measured in the best-case scenario where the SP does not misbehave. Also, the interactions that are not part of the protocol – e.g. getting to know the SP or doing the financial transaction in itself – are not accounted for. Finally, the computation time overhead generated by the network communication, be it anonymous or not, is not included.

The SPs are the ones who are required to do most of the computations, but that is not problematic since the amount of work delegated to them is proportional to their number of feedback values, that is to say to their popularity and presumably to their computational capacity.

7.2 User Anonymity Preservation

We see in the extended version of this paper [5] that under some reasonable assumptions on the security of the building blocks, we have:

Property 1. *The client can choose his anonymity set (or a lower bound of it) for a future feedback submission.*

Property 2. *The client/feedback unlinkability remains in agreement with the anonymity set defined by the client.*

Property 3. *The feedback/feedback unlinkability is guaranteed within the anonymity set chosen by the client.*

7.3 Decentralization

The only potentially centralized entity in this protocol is the CA. The presence of a CA to create the tokens is a necessity to prevent Sybil attacks (and more precisely self-promotion) if we want to avoid the use of fees. However, this CA could be comprised of several entities and thus decentralized, using a multi-signature scheme such as [7].

The trackers are also comprised of several distributed entities. Even though there has to be a limited number of trackers for the protocol to remain efficient, it is still to be considered as decentralized.

7.4 Robustness

Assumption 1. *If the peers have a list of N running trackers, at least $\lfloor \frac{N}{2} \rfloor + 1$ of them are honest.*

Assumption 2. *The CA only delivers certificates to individual and unique users.*

As explained in the long version of this paper [5], the following property holds given the two previous assumptions:

Property 4. *The reputation score that is retrieved by clients is the aggregated feedback from all the buyers and the few identities Ω_{SP} colluding with the SP, if this SP does not undertake any detectable malicious behavior.*

Two main deterrents that protect the protocol against any misbehavior of SPs are as follows:

- Misbehaviors such as always being offline, not providing reputation or not allowing transactions are detrimental to the SP itself.
- Misbehaviors such as refusing to publish negative feedback allow the client to forge a verifiable proof of bad behavior that can be escalated to the trackers.

8 Conclusion

In this article, we have presented a reputation system that is consistent with our objectives: efficient, anonymity-preserving, decentralized, and robust against various known attacks against reputation systems, such as ballot-stuffing and Sybil attacks. To the best of our knowledge, this is the only scheme in the state-of-the-art that achieves these attributes concurrently in a single protocol. We use Merkle trees and signed blocks of data to minimize the workload on the clients and trackers and to fairly distribute the record maintenance tasks to the service providers. Clients are able to retrieve the reputation of a given service provider in constant time. Despite the fact that the SPs are in charge of maintaining their own reputation records, the proofs of malicious behavior provided by the protocol deter them from acting maliciously. For future work, some improvements can be considered to further minimize the role of the trackers.


References

1. Bittorrent protocol. <http://www.bittorrent.com/>
2. Anceaume, E., Guette, G., Lajoie Mazenc, P., Prigent, N., Viet Triem Tong, V.: A privacy preserving distributed reputation mechanism, October 2012
3. Androulaki, E., Choi, S.G., Bellovin, S.M., Malkin, T.: Reputation systems for anonymous networks. In: Borisov, N., Goldberg, I. (eds.) PETS 2008. LNCS, vol. 5134, pp. 202–218. Springer, Heidelberg (2008). doi:[10.1007/978-3-540-70630-4_13](https://doi.org/10.1007/978-3-540-70630-4_13)
4. Barbaro, M., Zeller Jr., T.: A face is exposed for AOL searcher no. 4417749, August 2006
5. Bazin, R., Schaub, A., Hasan, O., Brunie, L.: A decentralized anonymity-preserving reputation system with constant-time score retrieval (technical report). Cryptology ePrint Archive, Report 2016/416 (2016). <http://eprint.iacr.org/2016/416>
6. Bethencourt, J., Shi, E., Song, D.: Signatures of reputation. In: Sion, R. (ed.) FC 2010. LNCS, vol. 6052, pp. 400–407. Springer, Heidelberg (2010). doi:[10.1007/978-3-642-14577-3_35](https://doi.org/10.1007/978-3-642-14577-3_35)
7. Boldyreva, A.: Threshold signatures, multisignatures and blind signatures based on the Gap-Diffie-Hellman-Group signature scheme. In: Desmedt, Y.G. (ed.) PKC 2003. LNCS, vol. 2567, pp. 31–46. Springer, Heidelberg (2003). doi:[10.1007/3-540-36288-6_3](https://doi.org/10.1007/3-540-36288-6_3)

8. Camenisch, J., Koprowski, M., Warinschi, B.: Efficient blind signatures without random oracles. In: Blundo, C., Cimato, S. (eds.) SCN 2004. LNCS, vol. 3352, pp. 134–148. Springer, Heidelberg (2005). doi:[10.1007/978-3-540-30598-9_10](https://doi.org/10.1007/978-3-540-30598-9_10)
9. Camenisch, J., Stadler, M.: Proof systems for general statements about discrete logarithms. Technical report 260, Institute for Theoretical Computer Science, ETH Zurich, March 1997
10. Chaum, D.: Blind signatures for untraceable payments. In: Chaum, D., Rivest, R., Sherman, A. (eds.) *Advances in Cryptology*, pp. 199–203. Springer, US (1983)
11. Chaum, D., Fiat, A., Naor, M.: Untraceable electronic cash. In: Goldwasser, S. (ed.) CRYPTO 1988. LNCS, vol. 403, pp. 319–327. Springer, New York (1990). doi:[10.1007/0-387-34799-2_25](https://doi.org/10.1007/0-387-34799-2_25)
12. Dimitriou, T., Michalas, A.: Multi-party trust computation in decentralized environments in the presence of malicious adversaries. *Ad Hoc Netw.* **15**, 53–66 (2014)
13. Dingledine, R., Mathewson, N., Syverson, P.: Tor: the second-generation onion router. In: *Proceedings of the 13th Conference on USENIX Security Symposium, SSYM 2004*, vol. 13, pp. 21–21. USENIX Association, Berkeley (2004)
14. Douceur, J.R.: The sybil attack. In: *Proceedings of 1st International Workshop on Peer-to-Peer Systems (IPTPS) (2002)*
15. Goldwasser, S., Bellare, M.: *Lecture notes on cryptography*, p. 235 (2001)
16. Groth, J., Sahai, A.: Efficient non-interactive proof systems for bilinear groups. In: Smart, N. (ed.) EUROCRYPT 2008. LNCS, vol. 4965, pp. 415–432. Springer, Heidelberg (2008). doi:[10.1007/978-3-540-78967-3_24](https://doi.org/10.1007/978-3-540-78967-3_24)
17. Hasan, O., Brunie, L., Bertino, E., Shang, N.: A decentralized privacy preserving reputation protocol for the malicious adversarial model. *IEEE Trans. Inf. Forensics Secur.* **8**(6), 949–962 (2013)
18. Lajoie-Mazenc, P., Anceaume, E., Guette, G., Sirvent, T., Viet Triem Tong, V.: Efficient distributed privacy-preserving reputation mechanism handling non-monotonic ratings, January 2015
19. Merkle, R.C.: A certified digital signature. In: Brassard, G. (ed.) CRYPTO 1989. LNCS, vol. 435, pp. 218–238. Springer, New York (1990). doi:[10.1007/0-387-34805-0_21](https://doi.org/10.1007/0-387-34805-0_21)
20. Nakamoto, S.: Bitcoin: a peer-to-peer electronic cash system (2008)
21. Narayanan, A., Shmatikov, V.: Robust de-anonymization of large sparse datasets. In: *IEEE Symposium on Security and Privacy, SP 2008*, pp. 111–125, May 2008
22. Okamoto, T.: Efficient blind and partially blind signatures without random oracles. In: Halevi, S., Rabin, T. (eds.) TCC 2006. LNCS, vol. 3876, pp. 80–99. Springer, Heidelberg (2006). doi:[10.1007/11681878_5](https://doi.org/10.1007/11681878_5)
23. Pavlov, E., Rosenschein, J.S., Topol, Z.: Supporting privacy in decentralized additive reputation systems. In: Jensen, C., Poslad, S., Dimitrakos, T. (eds.) *iTrust 2004*. LNCS, vol. 2995, pp. 108–119. Springer, Heidelberg (2004). doi:[10.1007/978-3-540-24747-0_9](https://doi.org/10.1007/978-3-540-24747-0_9)
24. Petrlic, R., Lutters, S., Sorge, C.: Privacy-preserving reputation management. In: *Proceedings of the 29th Annual ACM Symposium on Applied Computing, SAC 2014*, pp. 1712–1718. ACM, New York (2014)
25. Resnick, P., Zeckhauser, R.: Trust among strangers in internet transactions: empirical analysis of eBay’s reputation system, chap. 6, pp. 127–157
26. Schaub, A., Bazin, R., Hasan, O., Brunie, L.: A trustless privacy-preserving reputation system. *IFIP SEC - Privacy* (2016)

**William Winsborough Commemorative
Address and Award 2017**

Strong Accountability and Its Contribution to Trustworthy Data Handling in the Information Society

Siani Pearson^(✉) 

Malvern, UK

siani.pearson@btinternet.com

Abstract. Accountability has long been the subject of discussion within public administration. Especially given the potential privacy and security risks arising from rapidly changing usage of information technology (IT), it can be useful to apply this notion also in the commercial world, relating to the actions of private organisations. However, accountability may be neither a necessary nor a sufficient condition for trust. In order to provide an improved basis for trustworthiness via enhancing accountability, certain conditions need to be met. In this paper we elucidate what these conditions are and explain the related notion and importance of strong accountability. Further, we ground this analysis within the wider context of organisational ethical decision making. As a topical case in point we focus on the data protection area and the protection of personal data.

Keywords: Accountability · Data protection · Ethics · Trust

1 Introduction

Recent changes in information technology (IT) such as the shift to hybrid computing, increase in mobile connectivity and big data explosion are giving rise to a rapid transformation of enterprise IT. Adopting the new style of IT across all industry sectors has distributed our data everywhere, increasingly connecting different types of objects, collecting data in new ways, creating new exposures and attack surfaces. Concerns continue to grow around what has been called the ‘darker side’ of the Information Society.

From a societal perspective, this new IT can be used in ways that undermine social values and citizens’ expectations [1]. Not only the privacy of the world’s citizens is challenged, but there are unprecedented implications for their safety, for example concerning the reliability of critical infrastructure. The relationship between online privacy and security is actually quite complex, but online privacy goes beyond just confidentiality and encompasses a range of personal data handling mechanisms. There is a major difference between protecting data and using data, and in the past privacy and security have too often been considered as a zero sum gain. This tension between privacy and security that was discussed in the 1990s has now given way to more complex tensions such as privacy and autonomy versus open data and the free flow of information. These tensions have been exacerbated by highly publicised cases such as the Snowden revelations about mass surveillance by the United States (US) intelligence

services, the Schrems vs Facebook ruling by the European Court of Justice which ruled the European Commission's US Safe Harbour decision to be invalid in view of the Snowden revelations and the US State Court ruling about Microsoft not having to reveal emails held in Dublin to the US Justice Department. Citizens tend to cooperate with corporate surveillance because it offers convenience, and submit to government surveillance because it promises protection, and the result is a mass surveillance society underpinned by the new IT [2].

From the business perspective, associated risk has to be managed in a way that takes account of increasingly sophisticated cyber-attacks as well as potentially costly and complex regulatory pressures. Organisations face a trust challenge in which innovation and potential customer and societal benefits have to be weighed against legal obligations and customer and societal expectations. Not only do they need to decide which actions to take, but they may need to justify those to others [3]. Several recent cases highlighted in the press illustrate how, in order to increase trust with their customers, certain corporations are trying to protect the privacy of their customers' data from unwanted government surveillance, or at least be as transparent about what is revealed as they can, even though they face legal constraints about what they can do or reveal [2, 4], and governments are trying to counter this in the name of national security [2, 5]. Yet new services and practices typically involve multiple parties, some of whom are invisible to the data subject (DS) (individual whose data is being processed), and often their rewarding potential is proportionate to the potential risks in terms of privacy. More and more this data processing drives new business intelligence, helping innovation of new services and products. Moreover, dynamic and fierce competition can bring business practices that have not been tested from a privacy or data protection side.

Due to the way these services and networks tend to be borderless, addressing concerns around security is not only a national priority but is inherently global, and traditional legal frameworks are struggling to cope [6]. A multitude of different regulatory approaches, variable interpretations and academic visions generate uncertainty, complexity and risks for both companies and DSs as it may become difficult to ensure efficient protection of people's private life as well as to comply with applicable national laws and frameworks. Due to technological development, data flows can be dynamically changing, fragmented and global. The data of a specific DS may move from one day to the other in different places and be split into chunks requiring processing by different entities in different places of the world. Data creation and collection is increasing exponentially, in ways that may or may not provide new or improved services for the benefit of the DSs involved. In dynamic contexts like cloud there are further problems due to potentially weak trust relationships with new providers and the time needed to set up contractual arrangements that allow transborder data flow of personal information [7].

In order to maintain social and commercial trust, ethical codes of practice can be used by organisations and these will have to forbid some uses of information technology that are legally compliant, commercially profitable, and technologically possible. As Angela Merkel [8] said in a speech influenced by her experience of being an object of US surveillance, *"When we proceed as if the ends justify the means, when we do everything that is technologically possible, we damage trust; we sow mistrust. In the end there is less,*

not more security.” Damaging trust may also result in the end in less profit and economic growth, and missed opportunities for improvement of lives by novel technologies.

In this paper we consider this problem of trustworthy organisational behaviour in our modern world and some solutions to it, focussing on the role of accountability. The following section considers briefly a number of ethical issues arising from recent changes in IT. Section 3 shows how ethical decision making can vary according to the framework adopted and considers how substance might be introduced into this process. In Sect. 4 the potential role of accountability within this process is assessed, particularly with regards to the central aspect of companies being able to show that they are behaving in an ethical way. Furthermore, in Sect. 5 the way in which accountability can be used in such a context in order to increase the trustworthiness of organisations to other parties, and especially to citizens, is discussed, and a case is made for *strong accountability* in order to satisfy this need. This is an important aspect in countering potential organisational behaviour in using notions of accountability, ethical frameworks and trust as a smokescreen for actions that ultimately decrease or attack universal human rights or social norms, decrease privacy, increase surveillance and the like, or ultimately do not take enough account of the summation of individual citizens’ interests as against the single corporate interest. Although we consider European data protection as a significant example throughout this paper, analogous arguments may well apply to other domains including environmental sustainability.

2 Ethical Issues Arising from Recent Changes in Information Technology

In this section we consider a number of ethical issues arising from recent changes in IT. Web 2.0 and the rise of social networks shifted the balance of generation of Internet content from service providers to users, and thereby blurred the distinction between the data controller (DC) (who determines the means and purposes of processing of personal data) and the DS. Furthermore, over time:

- metadata has become increasingly viewable as personal data
- de-anonymisation has been made much easier
- storage costs have decreased
- the dangers of profiling have become evident
- large-scale collection of personal data using opt out mechanisms has been carried out
- differences between legislation applying where the DC and DS are in different countries could cause difficulties or potential harm to either (particularly in the sense of solutions being either ineffective or difficult to implement).

Connected to these general trends, different social and ethical issues can be associated with specific business models and technologies [9]. For example:

- *cloud computing*: lack of control and transparency [10], increased risks due to de-localisation and subprocessing [7], changes in risk perception [11, 12], fears about surveillance by foreign governments [13]

- *big data*: secondary usage of customers' data, unwanted profiling, potential discrimination, easier de-anonymisation and mining of information from social networks
- *mobile*: unwanted collection of personal and location information by apps and issues with the readability of privacy policies
- *internet of things (IoT)*: increased surveillance and behavioural tracking, difficulties in obtaining consent and difficulties in providing remediability and redress

One way of approaching this topic (fitting especially well with the social and historical European context, although the values apply universally) is preservation of values of the Enlightenment. During the seventeenth and eighteenth centuries, groups of intellectuals and philosophers, such as the Lunar Society of Birmingham, held discussions that helped articulate the notion of individual rights, amongst other things. But are such values that have underpinned our modern, secular age now under threat? As Tim Berners Lee has been quick to point out, the Internet need not itself result in that, as it may support universality and new rights including access to the Internet [14]. In order to avoid sweeping away rights and values in our digital era that the classical enlightenment helped reinstate, we need a new type of governance in which we can avoid technological 'dark paths' and in which fairness and human autonomy and rights are important. Ethical behaviour and choices corresponding to this will help build trust.

Although some social norms may gradually evolve, when it comes to the real consequences and harms of privacy intrusion the concerns will be the same and protection will not be useless. They are in fact more useful than ever when we observe that new innovative business models such as the ones mentioned above become less obvious and understandable by DSs. More specifically, individuals' ethical judgements about the collection of personal data are distorted by a number of practical factors. Even if an individual is actively and willingly disclosing data, he/she may be doing so on the basis of a flawed, incomplete or misleading set of assumptions. So, fear and doubts are shaping perceptions and trust becomes a key requirement. As the Eurobarometer survey (June 2015) [15] found, protection of personal data remains a very important concern for citizens. For example, nine out of ten Europeans think that it is important for them to have the same rights and protection over their personal information, regardless of the country in which the public authority or private company offering the service is based, and 69% of people say their explicit approval should be required in all cases before their data is collected and processed [15].

In general, ethical issues in IT include the following [16]:

1. one should have *no surprises* about data usage, or put too much emphasis on legal rather than what is legitimate, or overly make use of exemptions
2. *ethical dilution* may occur for example because harm can be difficult to quantify (and it could be potential and not just physical or financial)
3. *different stakeholders* are involved who could have competing interests, be unequal in terms of influence, or speak different languages
4. there can be changing and *complex contexts* magnifying risks of re-identification, lack of consent and lack of transparency

After having considered in this section some social and ethical implications in the information society, next we look at how these are being addressed in the form of ethical frameworks, what the impact of this is on businesses and how organisations can take active steps that include being accountable.

3 Ethical Decision Making

In Sect. 1, we introduced the organisational trust challenge in which innovation and potential societal benefits are balanced against societal expectations and legal obligations. The business context and risk appetite of the organisation will affect how much it wants to risk non-compliance and various forms of backlash from users and from supervisory authorities; there are a number of potential risks including reputational damage, business continuity impact and fines. In order to avoid getting the balance wrong, ethical frameworks have an important role to play, and this will be considered further in this section. As we shall see in later sections, privacy by design, accountability and security are all aspects that need to be taken into account when organisations deploy the resultant solutions, as well as embedding ethical decision making into their operations and culture.

3.1 Different Approaches to Ethics

Much more broadly than the IT domain, different ethical approaches can be taken. Broadly speaking, these divide into teleological approaches (an ethics of what is good – for example, utilitarianism) and deontological approaches (an ethics of what is right – for example, using Kant’s categorical imperative) [17]. Depending on which ethical approach you take, you might get a different answer about what you should do. A teleological decision looks at the rightness of wrongness based on the results or the outcomes of the decision. A deontological decision instead considers the moral obligations and/or duties of the decision maker based on principles and rules of behaviour. More information about the various different sub-approaches and philosophers in such a taxonomy of commercial ethics is given in [17]. The ethical dimensions of productive organisations and commercial activities have been studied since the 1970s within the field of business ethics, and a number of different approaches can be taken corresponding to this, as summarised for example within [18], ranging from Milton Friedman’s [19] view of corporate executives’ responsibility generally being to maximise profits while conforming to basic rules of the society to the opposing idea of corporate social responsibility (actions by businesses that are not legally required and intended to benefit other parties) [20].

3.2 Ethical Frameworks

In order to use these ethical approaches in a practical perspective by embedding ethics within business operations, one approach is to try alternative approaches and see the extent to which there is agreement.

This may look simple, but actually it is not necessarily an easy process. Let us consider comparing just one form of deontological judgment with one form of teleological judgment. If the result were that you would be doing the wrong thing and getting the wrong results (poor outcome), it might seem fairly obvious that a project fitting in that category should not go ahead, just as it needs little thought that a project doing the right thing and getting a good outcome is perfectly fine to go ahead. However, if you regularly deliver highly on the deontological spectrum but poorly on the teleological spectrum, you may well go out of business as it just might not be sustainable financially to continue. Conversely, if delivering highly on the teleological spectrum but low on the deontological spectrum, the drive for profit is taking precedence over consideration about what is (or is not) the right thing to do. In particular there is a zone of ethical nuances (especially along the boundaries between these) where the conclusion is not clear. Furthermore, when there is an economic slump, things can be perceived to be ethically questionable that would not have been before, so this ethical nuances zone can change [17].

Moreover, there tend to be different kinds of ethical perspectives for different types of organisations. For instance, guardian roles (such as regulators) seem to favour a deontological culture, whereas commercial institutions seem to favour a teleological culture and other actors (such as activists and technologists) may favour virtue ethics roles. Broadly speaking, governmental policy makers have outcome-based ethics, like commercial organisations, but are interested in economic and developmental outcomes at the national or regional level rather than the organisational level. Individuals who may be DSs have their own ethical framework. These different ethical frameworks and potentially conflicting objectives can make designing ethical codes of practice for the configuration and commercial use of new technologies difficult [17]. The code of practice could be a failure if it is unacceptable to any of these types of stakeholder. It must provide adequate protection of individuals' rights and interests. It must also give guidelines and assist with compliance to laws and regulations, as well as being practical for information technologists to comply with, and allowing new innovative mechanisms to achieve their potential for driving socially and economically beneficial applications.

In addition, as we considered in the previous section, it is beneficial to take into account a more nuanced understanding of "harm" including risk, potential harm, and forms of harm other than just physical and financial. In the data protection sphere, this is somewhat accounted for within the notion of Data Protection Impact Assessments (DPIAs) [3], which extend the standard practices of security risk analysis to examine also harms to the DS with regard to a proposed activity. In carrying out this assessment the summation of the harm across society needs to be properly evaluated and justified, as otherwise there is a risk that the potential harm to a single individual, as measured by an organisation that has a particular activity in mind, will typically tend to be over-ridden by other concerns.

3.3 Examples Addressing Technological Change

There are a range of different examples of ethical frameworks for decision making in contexts particularly influenced by recent technological development from different countries, most of which are still under development. In particular:

1. **British Computer Society (BCS) DIODE [21]**: a five stage ethical meta-framework (with iteration), within which different ethical approaches can be utilised.
2. **US Department of Homeland Security's Menlo Report [22]**: this framework for ethical guidelines for computer and information security research centres around four ethical principles: respect for persons; beneficence; justice; respect for law and public interest (which includes transparency and accountability).
3. **Information Accountability Foundation (IAF)'s Unified Ethical Frame for Big Data Analysis [23]**: an ethical framework for big data based on five values: beneficial; progressive; sustainable; respectful; fair.
4. **UK Government Cabinet Office's Data Science Ethical Framework [24]**: this focuses on six principles to stimulate ethical action when conducting data science: start with clear user need and public benefit; use data and tools which have the minimum intrusion necessary; create robust data science models; be alert to public perceptions; be as open and accountable as possible; keep data secure.
5. **European Data Protection Supervisor's (EDPS) opinions on ethics [1, 25, 26]**: freedom and dignity underpin the proposed approach, with user control, transparency, privacy by design and accountability being key aspects of the ethical solutions. In addition, EDPS has formed an ethics board to provide advice about ethical approaches to data protection in Europe.

Of course, there has been a substantial body of research in ethics for quite some time that is relevant to making ethical judgments relating to technology [17, 27]. Of particular interest is a proposal by Gary Marx [28] that the ethics of a surveillance activity must be judged according to the means, context and conditions of data collection and the uses/goals. Furthermore, he has defined 29 questions related to this – the more one can answer these questions in a way that affirms the underlying principle, the more ethical the activity [28]. This provides a substantive basis for ethical judgment that appears to be currently lacking from many ethical frameworks – instead the latter often just present a number of key values as a basis for discussion by groups of experts and/or interested parties, and a process for the results to be reported back to other parties [23, 29].

Accountability is part of all of the above frameworks, but it is only one aspect of the proposed ethical code or approach. Other aspects that should be considered include for example: data minimisation; strong constraints around re-identification and (very) harmful uses of data; special treatment of sensitive data; constraints on sources used and recipients of data produced. Since this paper focuses on accountability, we will not consider those aspects further here. However, accountability is not only a way of ascribing ethical considerations beyond the DC, but also contributes to solutions: *“Transparency and accountability towards the range of stakeholders in business – including employees, customers, suppliers, shareholders, local communities, society at*

large and the environment – are both a standard that is expected, and a mechanism for securing compliance with codes of conduct designed to meet society’s expectations.” [30]. We consider this aspect further in the following sections.

Many ethical frameworks aim to take a wider range of aspects into account than just data protection [28]. However, in this paper, we will look in particular at one example that is a current hot topic, namely data protection.

European Data Protection. Security is a very strong requirement for data protection but it is not enough. The Organisation for Economic Co-operation and Development (OECD) privacy principles [31] have formed the basis for most data protection and privacy laws worldwide. These are privacy principles that should apply regardless of the institution or technology and are a rules-based (deontological) approach. Since the introduction of the legislative framework for protection of personal data in the European Union (EU) in 1995 (in the form of Directive 95/46/EC) which largely reflects these principles, there has been a fast pace of technological change. In 2003 this was complemented by the E-Privacy Directive (2002/58/EC), which, amongst other things, placed traffic and location data into the category of personal DS to the regime. As a result of further technological change (as discussed in Sect. 2), there has been a major revision of European data protection legislation, called the General Data Protection Regulation (GDPR) [3], which was agreed upon by the European Parliament and Council in December 2015 and will introduce uniform requirements in all Member States, with the corresponding enforcement (and penalties of up to 4% of global turnover) starting in 2018. Within this regulation accountability is an important concept, that we now consider further.

4 How Accountability Can Contribute to These Solutions

4.1 The Concept of Accountability

Accepting responsibility, providing accounts and holding to account are central to what is meant by *accountability*. In the data protection context, the concept encompasses an end to end data stewardship regime in which the enterprise that collects personal and business confidential data is responsible and liable for how the data is shared and used, including onward transfer to and from third parties.

Accountability (for complying with measures that give effect to practices articulated in data protection guidelines) has been present in many core frameworks for privacy protection, starting with OECD’s privacy principles in 1980 [31]. More recently, not only have regulators increasingly been requiring that companies prove they are accountable, but organisations themselves are seeing the benefits of taking an accountability-based approach. Legislative authorities have been developing frameworks such as the EU’s Binding Corporate Rules [32] and APEC’s Cross Border Privacy Rules [33] to try to provide a cohesive and more practical approach to data protection across disparate regulatory systems, and these can be regarded as an operationalisation of accountability.

From an analysis of the usage of the term ‘accountability’ in different fields [34], we propose the following definition:

Accountability: *State of accepting allocated responsibilities, explaining and demonstrating compliance to stakeholders and remedying any failure to act properly. Responsibilities may be derived from law, social norms, agreements, organisational values and ethical obligations.*

Thus, accountability relationships reflect legal and business obligations, and also can encompass ethical attitudes of the parties involved. Our analysis actually combines and extends two aspects, based upon ideas coming from the social sciences [35] such that both commitment and enforcement are involved in accountability. Thus, the concept of accountability includes a normative aspect, whereby behaving in a responsible manner is perceived as a desirable quality and laid down in norms for the behaviour and conduct of actors. This can be applied to steer accountable behavior of actors *ex ante*. Accountability also encompasses institutional mechanisms in which an actor can be held to account by a forum, that involve an obligation to explain and justify conduct and ensure the possibility of giving account *ex post facto* (via accountability tools).

We broaden the notion of a forum to that of an accountee in a service provision chain, or more broadly a business ecosystem of interacting organisations and individuals – the actors of the ecosystem – who provide and consume IT-based services. These actors are controlled not only by internal factors of the system, such as codes of conduct and existing relations, but also by external factors such as regulations, the wider environment or even required skills.

Our approach is towards further operationalisation of the way accountability should be embedded in the ecosystem's norms, practices and supporting mechanisms and tools. First, we steer accountability behavior of actors including service providers *ex ante*. Second, we allow for a mechanism that entails the social relation between the accountant and accountee that involves an obligation to explain and justify conduct and ensures the possibility of giving account *ex post facto* (via accountability tools, such as the ones described in [36]).

Our model is that an *accountor* is accountable to an *accountee* for the following objects of accountability:

- **Norms:** the obligations and permissions that define data practices; these can be expressed in policies and they derive from legislation, contracts and ethics.
- **Behaviour:** the actual data processing behaviour of an organisation.
- **Compliance:** entails the comparison of an organisation's actual behaviour with the norms.

By the accountant exposing the norms it subscribes to and the things it actually does, an external agent can check compliance. For more analysis on accountability obligations, especially those owed by cloud service providers, and organisations that use cloud services, to regulators, stakeholders and society, see [6, 36].

4.2 What Organisations Need to Do

Organisations operate under many norms, reflecting obligations and stakeholder expectations, and more broadly reflecting the various ethical, social and legal obligations that apply to their business situation. For example, in a cloud computing context, these could be regulations that apply to that organisation's provision or usage of cloud services (such as US Health Insurance Portability and Accountability Act, or HIPAA), as well as individual service level agreements (SLAs) that are in place. Accountable organisations need to implement appropriate measures to comply with these norms, which includes managing risks, adopting appropriate security controls, employing privacy by design and planning for remediation. Accountability does not typically itself directly address these requirements, other than providing information about mechanisms used or helping deal with breaches. In addition, a central part of accountability that increases transparency is to demonstrate how the norms are met and risks managed [30]. This risk assessment should include not only the standard organisational security risk assessment but also an assessment of the potential harm to individuals/DSs. It is possible indeed to incorporate the latter into the former or carry out a separate assessment (such as a DPIA or environmental assessment).

Accountability needs to be embedded into the culture and practices of the organisation. In moving to an accountability culture, decisions are made based on a set of ethics- and value-based criteria in addition to liability. So, for example, an organisation should not relocate operations to a country with a weaker legal framework in an effort to reduce its privacy protections.

The Global Accountability Project started by privacy regulators and privacy professionals [37] gives five essential elements of data protection accountability: (i) organisation commitment to accountability and adoption of internal policies consistent with external criteria, (ii) mechanisms to put privacy policies into effect, including tools, training and education, (iii) systems for internal ongoing oversight and assurance reviews and external verification, (iv) transparency and mechanisms for individual participation, and (v) means for remediation and external enforcement. Guidance has also been produced from Canada (and from other regulators around the world) about the expected form of comprehensive accountability programs that organisations should put in place [38]. In addition to such organisational practices, a variety of accountability tools may be utilised in support of accountability: see [36] for further details.

If these elements are already in place for data protection accountability, it makes sense to achieve accountability for ethical use of new technologies (such as big data collection and analysis) by extending the existing elements for data protection accountability to cover these considerations as well, so that the ethical code of practice is integrated into the existing elements. In any case, there should not just be a separate part of the organisation that deals with ethical issues, but the practices must be more integrated. For example, senior leadership should articulate and communicate an internal organisational policy consistent with the ethical code(s) and the policy should be part of mandatory data protection training for employees engaged in those activities and audited against.

Data Protection Example. The OECD principles [31] lead fairly directly to a number of practices that organisations acting as DCs need to take: organisations should be open about their policies and practices; personal information should only be collected for defined and relevant purposes; that information should only be used and disclosed in ways that are consistent with those purposes; access and correction rights should be granted to individuals; the data should be kept secure. However, there is a general movement globally towards less prescriptive approaches by regulators with organisations being allowed more control over which mechanisms to use, so long as they can show that they are meeting higher level goals [6]. In Europe, as mentioned in the previous section, GDPR [3] will include a new data protection principle: the principle of accountability. DCs will be compelled to adopt policies, organisational and technical measures to ensure and be able to demonstrate compliance with the legal framework.

5 The Relationship Between Accountability and Trust

Accountability can play an important role in enhancing trust in any information society; however, the relationship between the two concepts is complex because:

- *Accountability is not a necessary condition for trust:* deployment of certain security or privacy techniques (such as strong encryption with the keys controlled by the user) may engender trust without the need to trust the service provider, although trustworthiness is a much broader notion than security as it includes subjective criteria and experience, among other factors. It could be argued that if technologies were deployed where the trust model involves minimal trust in service providers and other associated actors – that is to say, if a combination of privacy enhancing techniques and encryption were used – there would be no need for accountability, and accountability is only needed to fill the gap where some trust in the service provider is needed.
- *Accountability may increase trust:* there is a paucity of such ‘minimal trust’ cases occurring in practice and indeed potential for re-anonymisation using additional information and meta-information even in such cases, thus creating a role for accountability. A good accountability deployment into an organisation might indeed increase its trustworthiness for potential clients: for example, a recent International Data Corporation survey [39] found accountability to be a key aspect of improving trust in cloud adoption.
- *Accountability is not a sufficient condition for trust:* it might be claimed that an accountability-based approach was being adopted, but this could be a smokescreen for weak privacy, perhaps even compounded by collusion in the verification process and the downplaying of DS expectations, wishes and involvement in the service provision. Indeed, verification is needed to encourage trust within an environment of market compliance, and trust issues will arise if levels of verification are perceived to be low.

From a societal perspective, an objection to accountability is that it could be a means to produce harmful effects for society [40]: big data and accountability can be regarded as two cycles of policy manoeuvre to try to accomplish the abolition of

purpose limitation in pseudonymous data [41]. This objection relates to the effects on both individual and society of a transition to continuous and ubiquitous data collection. Irrespective of the rules or algorithms governing how that data is used, this obviously would have legal effects on universal privacy rights, as well as a general “panoptic” effect of knowing that a record of individual behaviour exists inescapably. This is an entirely different social, political, and phenomenological situation that is incomparable with life without such (involuntary) life-logging.

Even if this wider context is ignored or disputed, other routes to potential harm to society, and DSs, may be considered. The trustworthiness of the process of verification of accounts produced *ex post facto* by that actor, and any associated remediation and penalties, are extremely important in affecting the strength of the accountability that evolves within a system. Moreover, there is a danger that individuals’ viewpoints might be overlooked and their choice and control reduced.

In order to strengthen the link between accountability and trust by providing stronger grounds for trustworthiness, we argue for the notion of *strong accountability*, which encourages ethical characteristics (such as high transparency in balance with other interests) and trustworthy mechanisms for producing and verifying logs as well as adequate enforcement. In the following sections therefore we examine some ethical considerations associated with accountability, and then consider the nature of strong accountability itself.

5.1 Ethical Considerations

Accountability can provide trust in fair behaviour, detect issues when they occur and provide effective support for remediation while calling for explanation if something goes wrong. This latter aspect is discussed by Dubnick in relation to public administration and debates concerning responsibility and professional integrity, who summarises the discussion with: “*Ethical behaviour, in short, required the presence of external accountability mechanisms in all their various forms*” [42].

Referring back to the discussion about accountability given within Sect. 4.1, one aspect involves development of ethical guidelines for behaviour of actors, aimed at certain types of ‘good willing’ actors and reflecting best practice in stewardship of data. When it comes to protecting personal and confidential data, ethical principles such as ‘do no harm’ and ‘respect for others’ are clearly relevant. Yet in making a decision about what would be ethical, in some cases, the agents, actions or purpose might not be known, and possible results or outcomes might be highly uncertain. There are also a number of ethical principles that come into play arising from looking after valuable data. Personal data is often valuable to the person identified due to the harm that could come to that person if accessed, altered or anyhow misused by others. It is also valuable to an organisation for administrative purposes or for business activities. Confidential data has a value in terms of who may, or may not, have access to it.

With accountability as a mechanism [35], both good and bad actors are held to account for the consequences of their behaviour; following on from the discussion in Sect. 3.2, the evaluators (and actors) might be using different ethical frames.

A number of ethical questions can also be posed with regard to the objects of accountability presented in Sect. 4.1:

Norms. How can legal obligations keep pace with developments in technology? How can ethical norms be defined and assessed? How can the interests of weaker parties not be subsumed by those of stronger parties? Is a given means of data collection ethically acceptable?

Behaviour. In determining whether the performance of an action by the accountant is ethical (and legal), there is a need to specify criteria for judging ‘good’, ‘bad’, ‘right’, etc., to judge the ethical quality of an accountant’s actions according to these criteria, and provide reasons if there are shortcomings. A core part of accountability is to determine and clarify the rights and obligations of actors. To illustrate for a cloud service provision example, there is a corresponding need to clearly allocate privacy and security responsibilities across the various cloud supply chain actors. A closely related attribute of accountability is *responsibility* [43]: *the property of an organisation or individual in relation to an object, process or system of being assigned to take action to be in compliance with the norms.* There can also be a link with the ethical obligation to honour promises: personal and/or confidential data is given to a third party in exchange for some service, but only given on the premise (and condition) that the data given will be used in accordance with some agreement made between the data provider (whether it be a private individual, or a company) and the service provider, and that it will be adequately ‘looked after’. Breaking an agreement or promise through a lack of care and attention could be unethical (based on Kant’s Categorical Imperative). Agreements not met or broken promises lead to a breach of trust, a loss of trust and confidence in the organisation, and potentially an end to the working relationship.

Compliance. Form is important, in that the corresponding accounts provided by the accountant should be truthful. But what is an appropriate level of detail/content for a given context, and how can trust be provided in the verification process? Procedural ethics, in the form of the ethics of the reporting and enforcement process, is relevant here. There should be an element of dialogue and transparency without overwhelming the recipient. There should also be a willingness to admit error and to be honest about the facts and not bury bad news. In the account provision process, the focus is often on the consequences of behaviour (outcomes) but it might also take actions into account. Turelli and Floridi [44] put forward accountability as an ethical principle such that accountees must be capable of being aware of outcomes and able to know the ‘actions’ that led to the outcomes for which the accountant is responsible: “*an agent should be held accountable for the consequences of her/his actions or projects*”. The accountant’s story about the collection and use of other people’s personal data must be open to inspection, rebuttal and dialogue by everyone because information privacy is a common, social and public good, not only an individual right [45].

In order to address how accountees might assess and enforce the interests of individuals and society, a notion of *democratic accountability* [46] can be useful. This reflects the right of society to information about the extent to which a private organisation has complied with (minimum) standards of law and other regulation, as well as

the right to information about public domain matters of a social and ethical nature (which can be elicited via public opinion). To restore power to the *demos* democratic accountability, accounting arrangements are characterised by two essential elements that are core accountability attributes, namely transparency and responsiveness:

Transparency. This is a *property of a system, organisation or individual of demonstrating and/or providing visibility of its governing norms, behaviour and compliance of behaviour to the norms* [43]. Accountability implies a process of transparent interaction, in which the accountee seeks answers and possible rectification. In the data protection realm, the commitments of the DC need to be properly understood by the DS and others and the focus on transparency is mostly around the processes and procedures that the controller must implement to protect the data, rather than on the data as such. There is *ex post* transparency that informs about consequences if data already has been revealed (i.e. what data are processed by whom and whether the data processing is in conformance with negotiated or stated policies): for example giving an account of a data protection breach, with remediation options. But there is also *ex ante* transparency that should enable the anticipation of consequences before data are actually disclosed or processed (usually with the help of privacy policy statements). For example: does the organisation have an effective complaint handling process? Is there a responsible person, such as a chief privacy officer? Is there a privacy management framework? Is there staff training? In addition, transparency of operations helps counter the ‘invisibility factor’ [27], which is a key reason why computers raise ethical issues. For example, in cloud contexts data passes from the DS ultimately to a third party, where the location and involved processes may be invisible. Transparency is not however always a good idea because there are a number of tensions between transparency versus privacy, security, or usability – more information may lead to less understanding and may undermine trust [47]. In particular, there might be a conflict between maximal openness and the obligation to have appropriate technical and organisational security measures in place to protect personal data. Some of this conflict may be resolved by delegation of trust, in the sense that trusted third parties, such as auditors or supervisory authorities, may have privileged, yet verifiable, access to information that allows them to make assessments, of which only necessary information and conclusions are passed on to other parties (to avoid for example revealing specific security vulnerabilities or unnecessary personal data). This is in a sense a private accountability process, whereby there needs to be transparency between DCs and data processors, in such a way as to minimise security and privacy risks.

Responsiveness. This is a *property of a system, organisation or individual to take into account input from external stakeholders and respond to queries of these stakeholders* [43]. It could be argued that if the level of public and *ex-ante* accountability is not adequately high, there could be a lack of any role for individuals and public interest groups in the process, apart, maybe, for remediation mechanisms when negative impacts materialise. The provision of accounts is a process (see [48] for details) rather than being static. Ethical considerations include: Is a dialogue invited for people involved indirectly, for example people for whom the action reported in the account is consequential (such as cloud subjects [36])? Are an organisation’s ways of producing

an account open to testing? How is a sceptical search for alternative explanations accommodated? What procedures are in place for resolving disputes between accounts?

5.2 The Need for Strong Accountability

In order to address the above issues, we argue that an accountability-based approach should have the following characteristics, which together support a strong accountability approach [50]:

- **Support for externally agreed data protection approach:** Accountability should be viewed as a means to an end (i.e. that organisations should be accountable for the personal and confidential information that they collect, store, process and disseminate), not as an alternative to reframing basic privacy principles or legal requirements. In this way, the accountability elements proposed within GDPR are instrumental to provide a certain assurance of compliance with the data protection principles, but do not replace them, and DPIAs, codes of conduct and certifications are proposed to increase trust in service providers who adhere to them.
- **Clarity of responsibility:** The commitments of the DC need to be well defined – this is (part of) the aspect of responsibility, that is an element of accountability. Service provider responsibilities should be defined in contracts and the definition of standard clauses by the industry, as validated by regulators, will help service users (such as cloud customers) with lower negotiation capabilities. The commitments of the DC should include all applicable legal obligations, together with any industry standards (forming part of the external criteria against which the organisation’s policies are defined) and any other commitment made by the DC in privacy statements. In the cloud context, this is particularly important as entities may have multiple roles, e.g. they could be a joint controller and processor. Once again, the responsibilities of the entities along the service provision chain need to be clearly defined, including relative security responsibilities. On the other hand certain tasks will need to be jointly carried out to be effective, such as risk assessment and security management. In this case there is a clear need for cooperation and coordination.
- **Transparency:** This should be increased, in ways that do not decrease privacy or security. This includes the nature of accounts being public where possible, and the need for the commitments of the DC to be properly understood by the DSs (and other parties). In addition, the mechanisms used and relevant properties of the service providers in the provision chain need to be clarified as appropriate to cloud customers and regulators. Furthermore, DPIA/PIA is one form of verification for accountability (that should be used in conjunction with others) that can be used to help provide transparency about the nature of the risks, including the criteria used in the risk assessment, how decisions are made to mitigate risk, and whether the mechanisms to be used and implemented are appropriate for the context. Comprehensive obligations for controllers to inform supervisory authorities and DSs of personal data breaches would further increase transparency. It is not only customers and end users that might be affected by certain kinds of data processing, but society at large. Transparency should therefore also be aimed at the general public and the

regulator. This contributes to the maintenance of ethical standards, rather than stimulating a race to the bottom (of cost and privacy protection).

- **Trustworthy mechanisms for producing accountability evidence:** Trustworthy evidence needs to be produced and reflected in the account, for example using automated evidence gathering about non-compliance. Accountability evidence needs to be provided at a number of layers. At the organisational policies level, this would involve provision of evidence that the policies are appropriate for the context, which is typically what is done when privacy seals are issued. But this alone is rather weak; in addition, evidence can be provided about the measures, mechanisms and controls that are deployed and their configuration, to show that these are appropriate for the context. For higher risk situations continuous monitoring may be needed to provide evidence that what is claimed in the policies is actually being met in practice [49]; even if this is not sophisticated, some form of checking the operational running and feeding this back into an organisation's accountability management program in order to improve it is part of accountability practice.
- **Protection of evidence, assessments and accounts against tampering:** Technical security measures (such as open strong cryptography) can help prevent falsification of logs, and privacy-enhancing techniques and adequate access control should be used to protect personal information in logs and other accountability evidence [50]. Note, however, that data that is collected for accountability might be itself data that can be abused and hence also needs to be protected. The potential conflict of accountability with privacy is somewhat reduced as the focus in data protection is not on the accountability of DSs but rather of DCs, which need to be accountable towards DSs and trusted "intermediaries" such as the supervisory authorities.
- **Verifiability:** This is the extent to which it is possible to assess norm compliance [43]. Accounts must be adequately verified and collusion between the accountant, its partners and the accountee must be prevented. There needs to be a strong enough verification process to show the extent to which commitments have been fulfilled. Audits should be regular, in a similar way to Sarbenes-Oxley external audit, rather than one-off checks at the accountability programme level. Note however that missing evidence can pose a problem, and guarantees are needed about the integrity and authenticity of evidence supporting this verification and the account. In addition, the actor carrying out the verification checks needs to be trusted by the DS and to have the appropriate authority and resources to carry out spot checking and other ways of asking organisations to demonstrate accounts. That is why the data protection authorities will need to play a key role in the trust verification, for example in data protection certification. There are further related aspects supporting this approach in terms of responsibility and transparency, as listed above. In terms of external governance mechanisms, strong enforcement strategies, not only in terms of verification, but also in terms of increasing the likelihood of detection of unlawful practices and strong penalties if caught, seem to be a necessary part of accountability.

6 Conclusions

Accountability is not only an important aspect of ethical codes of conduct relating to business activities involving new technologies, but can also provide a mechanism for securing compliance with such codes of conduct. Centrally, it provides a mechanism for oversight within an organisation and enables external audit.

In the context of data protection accountability is particularly important, in that personal data has been given to an organisation for some stated purpose and it is expected by the provider of the data that it will be kept safe and used according to the established purposes. This is a legal obligation in many countries, and it is also a moral obligation. In our information society, personal data can be ‘under the charge’ of a variety of organisations in a way that is often not transparent or under the control of DSs, end users and customers. That data can also be transferred to many different locations under different legal jurisdictions.

In discussions about the laws that support data protection it is easy to get side-tracked from the most important issue. At the heart of data protection there is more than protection of the data – there is protection of the person to whom the data relates [3]. Transparency and accountability disclose satisfactory (or unsatisfactory) stewardship of data which to the originator – either DS or DC – is not just data but is information that has a value, either (for the DS) in terms of potential harm or possible benefit or (for the DC) through its business value or in costs incurred in collection and processing.

Enhancing accountability can be an improved basis for trustworthiness, and higher degrees of accountability, if appropriately advertised, could result in higher acceptance and trust by prospective customers. In order to be adopted, accountability must deliver effective solutions whilst avoiding where possible overly prescriptive or burdensome requirements. On the other hand accountability can also be used as a smokescreen for decreasing individual rights and for allowing businesses to give freer rein to non-paternalistic capitalist desires. The concept of ‘strong accountability’ is very important in helping demonstrate why (and indeed whether) an organisation should be trusted as well as in preventing the latter. An important aspect of this is that accountability should have democratic and ethical characteristics, in which transparency should be as high as possible in balance with other interests, and regulatory and supervisory authorities should have a primary role in the verification of the level of organisational compliance.

Acknowledgments. The author gratefully acknowledges input from Penny Duquenoy. It is partially based upon research carried out during EU A4Cloud project [51] and while employed at HPE Labs in Bristol, UK. However, it has been written subsequent to that period and does not represent any official position of HPE.

References

1. European Data Protection Supervisor (EDPS): Towards a New Digital Ethics. Opinion 4/2015, 11 September 2015
2. Schneier, B.: Data and Goliath: The Hidden Battles to Collect Your Data and Control Your World. W.W. Norton & Co., New York (2015)

3. General Data Protection Regulation (GDPR): Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (2016)
4. Ni Loideain, N.: EU law and mass internet metadata surveillance in the post-snowden era. *Media Commun.* **3**(2), 53–62 (2015). doi:[10.17645/mac.v3i2.297](https://doi.org/10.17645/mac.v3i2.297)
5. UK Investigatory Powers Act (2016). <https://www.gov.uk/government/collections/investigatory-powers-bill>
6. Charlesworth, A., Pearson, S.: Developing accountability-based solutions for data privacy in the cloud. *innovation, special issue: privacy and technology. Eur. J. Soc. Sci. Res.* **26**(1), 7–35 (2013). Taylor & Francis
7. Pearson, S.: Privacy, security and trust in cloud computing. In: Pearson, S., Yee, G. (eds.) *Privacy and Security for Cloud Computing*. CCN, pp. 3–42. Springer, London (2013). doi:[10.1007/978-1-4471-4189-1_1](https://doi.org/10.1007/978-1-4471-4189-1_1)
8. Shae, M.: English translation of German Chancellor Angela Merkel’s speech to the German Parliament, 29 January 2014
9. UK Information Economy Council: Addressing consumer confidence in the Digital Economy (2015). <http://www.digitalcatalapultcentre.org.uk/wp-content/uploads/2015/04/Information-Economy-Council-IEC-Principles-Consultation.pdf>
10. Article 29 Data Protection Working Party: Opinion 05/2012 on Cloud Computing (2012)
11. Cloud Security Alliance (CSA): The Treacherous Twelve: Cloud Computing Top Threats in 2016, Top Threats Working Group (2016)
12. CSA: The Notorious Nine: Cloud Computing Top Threats in 2013. Top Threats Working Group, February 2013
13. European Parliament (EP): Fighting Cyber Crime and Protecting Privacy in the Cloud. Directorate-General for Internal Policies (2012)
14. Berners-Lee, T., Halpin, H.: Defend the web. In: Bus, J., Crompton, M., Hildebrandt, M., Metakides, G. (eds.) *Digital Enlightenment Yearbook*, pp. 3–12. IOS Press (2012)
15. Jourova, V.: Data protection Eurobarometer. European Commission Factsheet 431 (2015)
16. Wilton, R.: Four ethical issues in online trust. In: *CREDS 2014* (2014)
17. Harris, I.: Commercial Ethics: Process or Outcome? Gresham Lecture, London, 6 November 2008
18. Moriarty, J.: Business Ethics. *Stanford Encyclopedia of Philosophy*, November 2016
19. Friedman, M.: The Social Responsibility of Business is to Increase Its Profits. *New York Times Magazine*, 13 September 1970
20. McWilliams, A., Siegel, D.: Corporate social responsibility: a theory of the firm perspective. *Acad. Manag. Rev.* **26**, 117–127 (2001)
21. Harris, I., Jennings, R.C., Pullinger, D., Rogerson, S., Duquenoy, P.: Ethical assessment of new technologies: a meta-methodology. *J. Inf. Commun. Ethics Soc.* **9**(1), 49–64 (2010). Emerald Group Publishing
22. Bailey, M., Dittrich, D., Kenneally, E., Maughan, D.: The Menlo Report. *IEEE Secur. Priv.* **10**, 71–75 (2012)
23. Information Accountability Foundation: A Unified Ethical Frame for Big Data Analysis. Big Data Ethics Project, v1.0 (2014)
24. UK Cabinet Office: Data Science Ethical Framework, v1.0, 19 May 2016
25. EPDS: Meeting the Challenges of Big Data. Opinion 7/2015, 19 November 2015
26. EDPS: Opinion on coherent enforcement of fundamental rights in the age of big data. Opinion 8/2016, 23 September 2016
27. Moor, J.H.: What is computer ethics? *Metaphilosophy* **16**, 266–275 (1985)

28. Marx, G.T.: An ethics for the new surveillance. *Inf. Soc.* **14**(3), 171–186 (1998)
29. Raab, C.D.: Information Privacy: Ethics and Accountability, Keynote Presentation for the Expert Workshop on ‘Cultures of Accountability’, KU Leuven, 13 November 2014
30. Lake, R.: Social Accountability, the OECD Guidelines for Multinational Enterprises and the OECD Principles of Corporate Governance (1999)
31. Organization for Economic Cooperation and Development (OECD): Guidelines on the Protection of Privacy and Transborder Flows of Personal Data (1980)
32. Information Commissioner’s Office (ICO): Binding corporate rules (2012)
33. APEC Data Privacy Sub-Group: Cross-border privacy enforcement arrangement, San Francisco (2011). http://aimp.apec.org/Documents/2011/ECSSG/DPS2/11_ecsg_dps2_010.pdf
34. Papanikolaou, N., Pearson, S.: A cross-disciplinary review of the concept of accountability. In: Proceedings of TAFC. IFIP, May 2013
35. Bovens, M.: Two concepts of accountability: accountability as a virtue and as a mechanism. *West Eur. Politics* **33**(5), 946–967 (2010)
36. Pearson, S.: Accountability in cloud service provision ecosystems. In: Bernsmed, K., Fischer-Hübner, S. (eds.) NordSec 2014. LNCS, vol. 8788, pp. 3–24. Springer, Cham (2014). doi:[10.1007/978-3-319-11599-3_1](https://doi.org/10.1007/978-3-319-11599-3_1)
37. Center for Information Policy Leadership (CIPL): Accountability: A compendium for stakeholders. The Galway/Paris Project (2011)
38. Office of the Information and Privacy Commissioner for British Columbia: Getting Accountability Right with a Privacy Management Program (2012)
39. International Data Corporation (IDC): Quantitative Estimates of the Demand of Cloud Computing in Europe (2012)
40. Bennett, C.J.: The accountability approach to privacy and data protection: assumptions and caveats. In: Guagnin, D., et al. (eds.) *Managing Privacy through Accountability*, pp. 33–48. MacMillan (2012)
41. Article 29 Data Protection Working Party: Opinion 03/2013 on purpose limitation (2013)
42. Dubnick, M.J.: Accountability and ethics: reconsidering the relationships. *Int. J. Organ. Theor. Behav.* **6**(3), 405–441 (2003)
43. Pearson, S.: Accountability in the cloud. In: Proceedings of the Trust in the Information Society, ITU Kaleidoscope Conference, Barcelona, Spain, pp. 5–16. IEEE, 9–11 December 2015
44. Turilli, M., Floridi, L.: The ethics of information transparency. *Ethics Inf. Technol.* **11**, 105–112 (2009). Springer
45. Raab, C.: Privacy, Security and Safety: Intelligence Services and National Security, IFIP Summer School 2016, Privacy and Identity Management – Facing Up To Next Steps, Karlstad, Sweden, 21–26 August 2016
46. Jaatun, M., Pearson, S., Gittler, F., Leenes, R., Niezen, M.: Enhancing Accountability in the Cloud. *Int. J. Inf. Manag.* (2016). Pergamon
47. Tsoukas, H.: The Tyranny of Light. *Futures* **29**(9), 827–843 (1997)
48. Gittler, F., Pearson, S.: Cloud Accountability Reference Architecture. D42.4a. A4Cloud Project Public Deliverable (2016). <http://www.a4cloud.eu/sites/default/files/D42.4%20Reference%20Architecture%20%28Final%29.pdf>
49. Pearson, S., Luna, J., Reich, C.: Improving cloud assurance and transparency through accountability mechanisms. In: Zhu, S.Y., Hill, R., Trovati, M. (eds.) *Guide to Security Assurance for Cloud Computing*. CCN, pp. 139–169. Springer, Cham (2015). doi:[10.1007/978-3-319-25988-8_9](https://doi.org/10.1007/978-3-319-25988-8_9)

50. Butin, D., Chicote, M., Le Métayer, D.: Strong accountability: beyond vague promises. In: Gutwirth, S., Leenes, R., De Hert, P. (eds.) *Reloading Data Protection: Multidisciplinary Insights and Contemporary Challenges*, pp. 343–369. Springer, Netherlands (2014)
51. Pearson, S., et al.: Accountability for cloud and other future internet services. In: *Cloud Computing Technology and Science*, pp. 629–632. IEEE (2012)

IFIPTM 2017 Graduate Symposium

Information Trust

Tosan Atele-Williams^(✉) and Stephen Marsh

Faculty of Business and Information Technology, University of Ontario Institute
of Technology, Oshawa, ON L1H 7K4, Canada
{tosan.atele-williams, stephen.marsh}@uoit.ca

Abstract. Information has been an essential element in the development of collaborative and cooperative models, from decision making to the attainment of varying goals, people are adept at making judgment on the trustworthiness of information, based on knowledge and understanding of a normative model of information. Contemporary narrative especially in high-impact contexts like politics, health, business, government and technology, is eroding trust in information, its source, its value and the ability to objectively determine the trustworthiness of a piece of information, a situation made more complex by social networks, social media have made the spread of information easier and faster irrespective of their trustworthiness, hence the need for judgment on the trustworthiness of a piece of information based on social cognitive construct, a trust model for information.

Keywords: Computational trust · Information trustworthiness · Decision support · Trust properties · Information value

1 Introduction

Various information behavior models, suggest a normative model of information as true, complete, valid, can be relied on as being correct and from a trusted source [1]; census data from statistics Canada can be regarded as valid and from a trusted source which can be reliably used for planning purposes, and as an economic tool, such data should carry more trusted weight than information sourced from a third party sources or social media platforms. Other normative information behavior prescribes trusted information as timely in the sense that it should be from a precise time period [1, 2], for example when analyzing census data for planning and developmental purposes it is paramount to look at current or the most recent figures. Information is of no value or worth the investment of time and money, especially in making business decisions if it is not relevant, does not have the right amount of details, cannot be easily stored in a way that it can be accessed effortlessly, or easily understood by the end user [1, 2, 4]. Other factors that add value and trustworthiness to information include but not limited to its accuracy, consistency, and completeness. Despite the best effort of information scientist on the nature of information, and work on information literacy behavior misinformation and disinformation still permeates social networks [1, 4], social media platforms like twitter and Facebook has helped in the spread of inaccurate information, a culture emboldened by need to share information even when the validity of the information

cannot be vouched for or when the person sharing such information does not believe it but regardless still goes ahead to share because it serves a narrative, a means to manipulate rather than to inform, as a source of social influence [3], as demonstrated by the recent political and business climate in the west that have added relatively new lexicons like fake news and alternate facts.

The consequences of deceptive and misleading information can be far-reaching for governments, citizens, business institutions, data professionals, and designers, it can create an atmosphere of mistrust, distrust, confusion, panic, and it can influence decisions and damage reputations. Information agents, brokers may find it difficult to use information, or seek alternate and less reliable sources of information because of the air of uncertainty, hence the need for an information model based on computational trust [5, 7], a paradigm drawn from a social, cultural, historical and psychological context and much more aspects of relationships [6], trying to model the best in these related milieus computationally.

Trust as a computational concept is important in understanding the thought process with regard to choice, options and decision-making process in human and computer interactions, especially in situations where there is a measure of risk [7, 8]. The goal is to formulate a theoretical framework; a socio-cognitive construct for the trustworthiness of information based on cues of credibility and deception, a model to assist judgment calls and an expectation of when, trust and its fulfillment can be expected.

Information does not exist in a vacuum, how it is perceived and used is influenced by a number of social, cultural, and historical factors, hence the need for an inclusive and context-aware information literacy behavior [1], our goal is to incorporate the characteristics of information; reliability, validity, and importance into a trust model, depending on the context, the model will also factor in the reputation of a source, the value of the information and cues to credibility and deception, with the aim of enabling agents to make judgments and situational decisions about the trustworthiness of information.

References

1. Karlova, N.A., Fisher, K.E.: A social diffusion model of misinformation and disinformation for understanding human information behavior. *Inf. Res.* **18**(1), 1–17 (2013)
2. Craig, S.: Lies, damn lies, and viral content. How news websites spread (and debunk) online rumors, unverified claims, and misinformation. Tow Center for Digital Journalism (2015)
3. Barber, K.S., Kim, J.: Belief revision process based on trust: agents evaluating reputation of information sources. In: Falcone, R., Singh, M., Tan, Y.H. (eds.) *Trust in Cyber-Societies*. LNCS, vol. 2246, pp. 73–82. Springer, Heidelberg (2001)
4. Marsh, S.: infoDNA (Version 2) Agent Enhanced Trustworthy Distributed Information. PST (2004)
5. Witkowski, M., Artikis, A., Pitt, J.: Experiments in building experiential trust in a society of objective-trust based agents. In: Falcone, R., Singh, M., Tan, Y.H. (eds.) *Trust in Cyber-Societies*. LNCS. vol, 2246, pp. 111–132. Springer, Heidelberg (2001)
6. Piotr, S.: *Trust: A Sociological Theory*. Cambridge University Press (1999)
7. Marsh, S., Briggs, P.: Examining trust, forgiveness and regret as computational concepts. *Computing with Social Trust*, pp. 9–43. Springer London (2009)
8. Golbeck, J. (ed.): *Computing with Social Trust*. Springer Science Business Media (2008)

Privacy and Trust in Cloud-Based Marketplaces for AI and Data Resources

Vida Ahmadi Mehri^(✉) and Kurt Tutschku

Blekinge Institute of Technology, Karlskrona, Sweden
{vida.ahmadi.mehri,kurt.tutschku}@bth.se

Abstract. The processing of the huge amounts of information from the Internet of Things (IoT) has become challenging. Artificial Intelligence (AI) techniques have been developed to handle this task efficiently. However, they require annotated data sets for training, while manual pre-processing of the data sets is costly. The H2020 project “Bonseyes” has suggested a “Market Place for AI”, where the stakeholders can engage trustfully in business around AI resources and data sets. The MP permits trading of resources that have high privacy requirements (e.g. data sets containing patient medical information) as well as ones with low requirements (e.g. fuel consumption of cars) for the sake of its generality. In this abstract we review trust and privacy definitions and provide a first requirement analysis for them with regards to Cloud-based Market Places (CMPs). The comparison of definitions and requirements allows for the identification of the research gap that will be addressed by the main authors PhD project.

Keywords: Privacy · Trust · Marketplace · IoT · Cloud · AI

1 A Market Place for Artificial Intelligence and Data

Bonseyes’ Market Place (MP) for AI [1–4] aims at engaging the various stakeholders, e.g. data providers, model, or application designer, into business among AI resources, i.e. data sets, models, training facilities, etc. The business around the resources may accelerate the model design and reduces the design costs. The MP will provide functions to offer, sell, pay or use AI resources and data sets. The proposed MP will be implemented by a cloud system in order to deal with the large size of data sets and to permit elasticity for the AI resources. This led to the notion of a CMP. As any MP, a CMP requires mechanisms to enforce *privacy* and *trust*. However, the separation between resource location (e.g., storage location) and resource availability (e.g. data availability) in cloud systems makes it more challenging to implement trustful mechanisms for these features as in non-virtualised systems.

2 Trust and Privacy Definitions for Network and Clouds

Trust and Trust Dimensions: a widely agreed definition for *trust* in networks, Clouds and systems is given in IETF Internet security glossary: as “... the extent to which someone who relies on a system can have confidence that the system meets its specifications, i.e., that the system does what it claims to do and does not perform unwanted functions” [5]. The view of applications, Clouds and networks as “systems” leads to the definition of multiple *trust dimensions* [6]. These dimensions comprise (a) “device trust”, i.e. the reliability of IoT devices to produce data correctly, (b) “operation trust”, refers to the combination of data traceability and analytics, (c) “communication trust”, builds on confidentiality, integrity, and authenticity in data transmission, (d) “infrastructure trust”, which aims at the transparency and predictability of processing.

Privacy and Privacy Dimensions: the IETF glossary also provides a definition for *privacy*: “... the right of an entity to determine the degree to which it will interact with its environment, including the degree to which the entity is willing to share information about itself with others” [5]. R.S. Poore defines privacy as a required context of personal Identifiable Information (PII) which have to be under control of the individual person who is the owner of it [7]. This view on privacy leads to *privacy dimensions* such as, cf. [6, 8]:

- Identity privacy: avoid the disclosure of users identity
- Location privacy: avoid the disclosure location information for specific user
- Device privacy: avoid the disclosure of device and security information
- Communication privacy: refers to encryption algorithm for confidentiality
- Access privacy: privilege levels for authorised data access
- Operation privacy: avoid the disclosure of data processing techniques
- Footprint privacy: avoid the identity disclosure by behavioural analysis
- Query privacy: avoid the identity disclosure by analysis of the origin of queries

3 Privacy and Trust Requirements for CMPs

In general, the privacy levels for an AI resource or a data set depend on the importance of the resource or of the type of PII stored in it. For example, medical records need very high levels of protection since a leakage of information may embarrass a specific person. Hence, if such data sets are traded then the specific levels of privacy needs to be maintained at the various locations where the data is accessed or processed, otherwise the users will lose their *trust* into the MP.

CMPs might host data sets with very different privacy levels at very different locations. As a result, they must enable a differentiated, transparent and even traceable handling of data. Some data sets may not be allowed to leave a certain physical premise due to regulation or provider policy, while others can do so. A CMP must support both modes of data availability at the required privacy levels for the sake of its generality. This feature is often denoted as the ability for “privacy by design” [9] of an architecture.

Since data sets are associated in a MP with a value, the infringement of this value by disclosing the data to other users needs to be avoided. Here, the privacy requirements, as seen from the data provider, turn into the problem of “Digital Right Management” that needs to be addressed by the MP architecture or its mechanisms and functions.

4 Conclusion



It is obvious that the definitions of trust and privacy do not directly address the virtualisation features of Cloud system. Particularly, the implications by separating between storage location and data availability in the Cloud are not clear yet. The privacy and trust dimensions might partly match with the application requirements of CMPs. Hence, their suitability for evaluation Cloud mechanisms needs to be investigated. These investigations as well as the design of mechanism for privacy and trust in CMPs will define the work in this PhD project.

Acknowledgment. This project has received funding from the European Unions Horizon 2020 research and innovation programme under grant agreement No 732204 (Bonseyes). This work is supported by the Swiss State Secretariat for Education Research and Innovation (SERI) under contract number 16.0159. The opinions expressed and arguments employed herein do not necessarily reflect the official views of these funding bodies.

References

1. BONSEYES - Artificial Intelligence Marketplace. <https://www.bonseyes.com/>
2. Bonseyes Consortium. Grant agreement number - 732204 bonseyes - annex 1 (part a): Description of action. Available on request from Bonseyes Consortium at <https://www.bonseyes.com/>, October 2016
3. Fricker, S., Maksimov, Y.: Pricing of data products in data marketplaces. In: 8th International Conference on Software Business (ICSOB) (2017, in submitted)
4. Llewellyn, T., Milagro, M., Deniz, O., Fricker, S., Storkey, A., Pazos, N., Velikic, G., Dahyot, R., Koller, S., Goumas, G., Leitner, P., Dasika, G., Wang, L.: Bonseyes: platform for open development of systems of artificial intelligence. In: ACM International Conference on Computing Frontiers (2017, in submitted)
5. RFC 2828. Internet security glossary, May 2000
6. Daubert, J., Wiesmaier, A., Kikiras, P.: A view on privacy and trust in iot. In: 2015 IEEE International Conference on Communication Workshop (ICCW), pp. 2665–2670, June 2015
7. Poore, R.S.: Anonymity, Privacy, and Trust. *Inf. Syst. Secur.* **8**(3), 16–20 (1999)
8. Cheng, Y., Naslund, M., Selander, G., Fogelström, E.: Privacy in machine-to-machine communications a state-of-the-art survey. In: 2012 IEEE International Conference on Communication Systems (ICCS), pp. 75–79, November 2012
9. Article 29 data protection working party and working party on police and justice, the future of privacy. Joint contribution to the Consultation of European Commission on the legal framework for the fundamental right to protection of personal data, WP168, December 2009

Psychological Evaluation of Human Choice Behavior in Socio-Technical Systems: A Rational Process Model Approach

Tim Schürmann  

Technische Universität Darmstadt, Darmstadt, Germany
schuermann@psychologie.tu-darmstadt.de

In the age of digital services and everyday smartphone usage, the issue of online privacy has gathered more and more interest for researchers, service providers and consumers. Assuming one's digital information is private is equivalent to trusting service providers to handle one's data in a certain way or ensuring protective measures against loss of privacy. When a consumer registers for an online service or installs a smartphone app, I assume an internal psychological process to relate the benefits of their decision to the risks associated with it. However, this process is considered to be subject of uncertainty. Therefore, decisions in a socio-technical environment can be viewed as decisions governed by a probabilistic amount of trust in an outcome, or, in other words, the amount of belief one holds that a hypothesis about future events will turn out to be true.

Previous research on human online behavior paints a fairly bleak picture of how we handle said uncertainty. It often adopts the paradigm of the Homo Heuristicus [1], relying on computational shortcuts rather than normatively rational inference. In a scenario as complex as online privacy, it also points out how unlikely it is for users to have a complete understanding of the capabilities and motives of involved parties [2].

However, psychological research on broader decision making processes includes evidence that humans are in fact able to combine information in a rational sense [3]. The Sampling Hypothesis [4] may provide the grounds for unifying research on heuristic or otherwise boundedly rational decision making on one hand with a rational account on the other. It does so by approximating Bayesian inference, sampling from probability distributions over possible hypotheses or parameter values instead of using these full distributions and creating implausibly complex computations. Its application shows that specific effects like the availability heuristic can actually be considered by-products of its sampling process [5]. Vul [4] provides evidence that in many situations, sampling only a very limited number of times provides a computationally similar result to using full yet analytically intractable probability distributions. Specifically, he links the benefits of sampling to the consumption of energy and time while arriving at a decision: why make one time- and energy-consuming decision perfectly maximizing my chance of success, when I can make many "good enough" decisions that approximate optimal results in the long run? This globally optimal solution however can produce seemingly irrational local behavior. Models that utilize such approximate Bayesian inference are termed rational process models [6].

It appears as though human subjects, while certainly limited in their cognitive resources and computational capabilities as laid out by the bounded rationality paradigm, may make use of this process: they operate by maximizing success chances and making rational choices, but on a global rather than local level. My work utilizes this type of model to investigate how humans make decisions online, and more importantly how to sensitize them to make more adequate decisions to protect their private information. A preliminary study [7] indicates that answers to these questions are not as simple as pointing to a specific heuristic approach or a systematic gap between privacy-related attitudes and behavior. When asked whether they wanted to install a travel-related smartphone app involving beneficial and non-beneficial features, subjects showed behavioral patterns that were predicted by a rational process model. Preference trade-offs for the app's features form the basis of the model prediction as a posterior distribution. Then, sampling from said individual posterior provides the model with an approximate probability of choosing to install the app. The model stochastically chooses the option with higher utility according to its probability. It therefore allows for a seemingly irrational decision on the local level when choosing the option with lower utility instead. The rate with which subjects chose their higher utility option or deviated from it was predicted by the model, with a deviation of approximately 5% between its prediction and the empirical data. This deviation is not significantly different from zero, as indicated by a Bayesian estimation of the difference parameter between the two.

The model seems to capture the process with which subjects combine preferences about features as well as their trade-off between utility maximization and cognitive resource management. It is based on subjective utility distributions, thus avoiding the assumption of complete situational knowledge proposed in previous research [2] to arrive at a rational decision. These subjective utility distributions in turn can be learned solely based on past experience [8]. It is worth noting that heuristic or probability-weighted alternatives of the model, following a cumulative prospect theory (CPT) approach, could possibly have resulted in a decent statistical fit as well. CPT's parameter estimation [9] would likely capture stochastic variations descriptively if it was retrospectively fitted for a specific individual and trial. It would not, however, explain the nature of the variation or the necessity of the sampling process on theoretical grounds. Meanwhile, the rational process model approach outlined here unites the idea of Bayesian computational rationality in human cognition with limitations on the algorithmic level [10]. Additionally, it allows for an explanation of other phenomena observed in decision making research, like probability matching.

Based on the preliminary study, I plan to first adapt the model to other interactions with socio-technical systems. Secondly, I will explore specific mechanisms of the model to apply them to privacy interventions. For example, increasing the number of samples drawn in the model increases the chance of choosing an option with higher utility, instead of sometimes choosing a lower utility option. This may be achieved by asking subjects to state their choice repeatedly. Assuming a privacy-protecting decision (not installing an app that requires permissions to access private data) is a subject's higher utility option, an intervention increasing their internal sample count should result in a higher probability of choosing that option. However, there is a chance that they favor a privacy-disclosing option. In that case, an intervention designed to increase sampling counts might reinforce the tendency to pick the disclosing option, resulting in

the opposite of the intended purpose. Future work will draw inspiration from how the mechanisms of sampling in decision making work to design privacy-protecting interventions tailored to individual preferences and thereby making use of the human tendency to operate on a globally rational level of information integration. Building on these rational process mechanisms, I aim to assess and direct user trust in the interaction with socio-technical systems as well as explain stochastic deviance from their expected behavior.

References

1. Gigerenzer, G., Brighton, H.: Homo heuristicus: why biased minds make better inferences. *Top. Cogn. Sci.* **1**, 107–143 (2009). doi:[10.1111/j.1756-8765.2008.01006.x](https://doi.org/10.1111/j.1756-8765.2008.01006.x)
2. Acquisti, A., Grossklags, J.: Privacy and rationality in individual decision making. *IEEE Secur. Priv. Mag.* **3**(1), 26–33 (2005). doi:[10.1109/MSP.2005.22](https://doi.org/10.1109/MSP.2005.22)
3. Griffiths, T.L., Tenenbaum, J.B.: Optimal predictions in everyday cognition. *Psychol. Sci.* **17**(9), 767–773 (2006). doi:[10.1111/j.1467-9280.2006.01780.x](https://doi.org/10.1111/j.1467-9280.2006.01780.x)
4. Vul, E., Goodman, N., Griffiths, T.L., Tenenbaum, J.B.: One and done? Optimal decisions from very few samples. *Cogn. Sci.* **38**(4), 599–637 (2014). doi:[10.1111/cogs.12101](https://doi.org/10.1111/cogs.12101)
5. Sanborn, A.N., Chater, N.: Bayesian brains without probabilities. *Trends Cogn. Sci.* **20**(12), 883–893 (2016). doi:[10.1016/j.tics.2016.10.003](https://doi.org/10.1016/j.tics.2016.10.003)
6. Sanborn, A.N., Griffiths, T.L., Navarro, D.J.: Rational approximations to rational models: alternative algorithms for category learning. *Psychol. Rev.* **117**(4), 1144–1167 (2010). doi:[10.1037/a0020511](https://doi.org/10.1037/a0020511)
7. Schürmann, T., Smirny, J., Zimmermann, S., Vogt, J.: Adoption behavior of smartphone apps gathering private data is explained by a sampling-based rational process model. Manuscript in preparation (2017)
8. Srivastava, N., Schrater, P.: Learning what to want: context-sensitive preference learning. *PLoS ONE* **10**(10), e0141129 (2015). doi:[10.1371/journal.pone.0141129](https://doi.org/10.1371/journal.pone.0141129)
9. Boos, M., Seer, C., Lange, F., Kopp, B.: Probabilistic inference: task dependency and individual differences of probability weighting revealed by hierarchical bayesian modeling. *Front. Psychol.* **7**, 755 (2016). doi:[10.3389/fpsyg.2016.00755](https://doi.org/10.3389/fpsyg.2016.00755)
10. Marr, D.: *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. The MIT Press (2010). doi:[10.7551/mitpress/9780262514620.001.0001](https://doi.org/10.7551/mitpress/9780262514620.001.0001)

Author Index

- Atele-Williams, Tosan 221
Au, Man Ho 152
- Basu, Anirban 12
Bazin, Rémi 180
Boender, Jaap 79
Brunie, Lionel 180
- Ceolin, Davide 49
Chen, Shuo 21
- de Lima Neto, Fernando Buarque 41
de Siqueira Braga, Diego 41
Dwyer, Natasha 110
- Fritsch, Lothar 3
Fukushima, Kazuhide 12
- Gal-Oz, Nurit 119
Gudes, Ehud 119
- Habib, Sheikh Mahbub 94
Hasan, Omar 180
Hellingrath, Bernd 41
- Kiyomoto, Shinsaku 12
- Lu, Rongxing 21
- Malik, Rabee Sohail 94
Mano, Ken 135
Marsh, Stephen 110, 221
Martin, Andrew 57
- Mehri, Vida Ahmadi 223
Meng, Weizhi 152
Milaszewicz, Pavlos 94
Mühlhäuser, Max 94
- Niemann, Marco 41
Nugraha, Yudhistira 57
- Othman, Hussien 119
- Pearson, Siani 199
Pernul, Günther 163
Potenza, Simone 49
Primiero, Giuseppe 79
- Rahman, Mohammad Shahriar 12
Richthammer, Christian 163
- Sakurada, Hideki 135
Schaub, Alexander 180
Schürmann, Tim 226
- Tsukada, Yasuyuki 135
Tutschku, Kurt 223
- Vasilomanolakis, Emmanouil 94
- Weber, Michael 163
- Xu, Rui 12
- Zhang, Jie 21