Robin Doss
Selwyn Piramuthu
Wei Zhou (Eds.)

Communications in Computer and Information Science 878

Future Network Systems and Security

4th International Conference, FNSS 2018
Paris, France, July 9–11, 2018
Proceedings

Springer

Communications in Computer and Information Science 878

More information about this series at http://www.springer.com/series/7899

Robin Doss · Selwyn Piramuthu
Wei Zhou (Eds.)

Future Network Systems and Security

4th International Conference, FNSS 2018
Paris, France, July 9–11, 2018
Proceedings

Springer

Editors
Robin Doss
Deakin University
Burwood, VIC
Australia

Selwyn Piramuthu
Information Systems and Operations Management
University of Florida
Gainesville, FL
USA

Wei Zhou
Information and Operations Management
ESCP Europe
Paris
France

ISSN 1865-0929 ISSN 1865-0937 (electronic)
Communications in Computer and Information Science
ISBN 978-3-319-94420-3 ISBN 978-3-319-94421-0 (eBook)
https://doi.org/10.1007/978-3-319-94421-0

Library of Congress Control Number: 2018947313

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 the registered company Springer International Publishing AG
part of Springer Nature
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

Welcome to the proceedings of the Future Network Systems and Security Conference 2018 held in Paris, France!

The network of the future is envisioned as an effective, intelligent, adaptive, active, and high-performance Internet that can enable applications ranging from smart cities to tsunami monitoring. The network of the future will be a network of billions or trillions of entities (devices, machines, things, vehicles) communicating seamlessly with one another and it is rapidly gaining global attention from academia, industry, and government. The main aim of the FNSS conference series is to provide a forum that brings together researchers from academia and practitioners from industry, standardization bodies, and government to meet and exchange ideas on recent research and future directions for the evolution of the future Internet. The technical discussions are focused on the technology, communications, systems, and security aspects of relevance to the network of the future.

We received paper submissions by researchers from around the world including Australia, New Zealand, Germany, India, Portugal, Italy, Sweden, UK, USA, UAE among others. After a rigourous review process that involved each paper being single-blind reviewed by at least three members of the Technical Program Committee, 14 full papers and two short papers were accepted covering a wide range of topics on systems, architectures, security, and applications of future networks. The diligent work of the Technical Program Committee members ensured that the accepted papers were of a very high quality and we thank them for their hard work in ensuring such an outcome.

July 2018

Robin Doss
Selwyn Piramuthu
Wei Zhou

Organization

FNSS 2018 was held at ECSP Europe, Paris, France, during July 9–11, 2018.

Conference Chairs

Robin Doss	Deakin University, Australia
Selwyn Piramuthu	University of Florida, USA
Wei Zhou	ESCP Europe, France

Program Committee

Maythem Abbas	Universiti Teknologi PETRONAS, Malaysia
S. Agrawal	Delhi Technological University (DTU) Formerly Delhi College of Engineering (DCE), India
Rana Khudhair Ahmed	Al-Rafidain University College, Iraq
Adil Al-Yasiri	University of Salford, UK
Abdul Halim Ali	Universiti Kuala Lumpur - International College, Malaysia
Elizabeth Basha	University of the Pacific, USA
Aniruddha Bhattacharjya	Guru Nanak Institute of Technology (GNIT), India
David Boyle	Imperial College London, UK
Doina Bucur	University of Groningen, The Netherlands
Yue Cao	University of Surrey, UK
Arcangelo Castiglione	University of Salerno, Italy
Sammy Chan	City University of Hong Kong, SAR China
Kesavaraja D.	Dr. Sivanthi Aditanar College of Engineering, India
Eleonora D'Andrea	University of Pisa, Italy
Soumya Kanti Datta	EURECOM, France
Safiullah Faizullah	Hewlett-Packard, USA
Stephan Flake	Redknee Germany OS GmbH, Germany
Felipe Garcia-Sanchez	Universidad Politecnica de Cartagena (UPCT), Spain
Razvan Andrei Gheorghiu	Politehnica University of Bucharest, Romania
Mikael Gidlund	Mid Sweden University, Sweden
Shweta Jain	York College CUNY, USA
Hussain Mohammed Dipu Kabir	Samsung Bangladesh R&D Center, Bangladesh
Mounir Kellil	CEA LIST, France
Piotr Korbel	Lodz University of Technology, Poland
Lambros Lambrinos	Cyprus University of Technology, Cyprus
Yee Wei Law	University of South Australia, Australia
Albert Levi	Sabanci University, Turkey

Hang Li	Texas A&M University, USA
Li Liu	Chongqing University, P.R. China
Jorge López Benito	CreativiTIC Innova SL, Spain
Marius Marcu	Politehnica University of Timisoara, Romania
Rakesh Nagaraj	Amrita School of Engineering, India
Nagendra Kumar Nainar	CISCO, USA
Shashikant Patil	SVKMs NMiMS Mumbai India, India
Yang Peng	University of Washington Bothell, USA
Umar Raza	Manchester Metropolitan University, UK
Nihar Roy	G. D. Goenka University, India
Hussain Saleem	University of Karachi, Pakistan
Rui Santos Cruz	Universidade de Lisboa, Portugal
Pasquale Scopelliti	University Mediterranea of Reggio Calabria, Italy
Vartika Sharma	GSSSIETW Mysuru, India
Paulus Sheetekela	University of Namibia, Namibia
Lei Shu	Guangdong University of Petrochemical Technology, P.R. China
Shailendra Singh	University of California, Riverside, USA
Tripty Singh	Amrita Vishwa Vidyapeetham, India
Koushik Sinha	Southern Illinois University, USA
Houbing Song	West Virginia University, USA
Dimitrios Stratogiannis	National Technical University of Athens, Greece
David Sundaram	University of Auckland, Australia
Carlo Vallati	University of Pisa, Italy
Neelanarayanan Venkataraman	VIT University, India
Chih-Yu Wen	National Chung Hsing University, Taiwan
Hui Wu	University of New South Wales, Australia
Keun Soo Yim	Google, Inc., USA

Contents

Secure Design and Verification

Security, Privacy and Ethics

Big Data and Applications of Future Network Systems

Secure Design and Verification

Formal Verification of RGR-SEC, a Secured RGR Routing for UAANETs Using AVISPA, Scyther and Tamarin

Houssem E. Mohamadi[1](✉), Nadjia Kara[2], and Mohand Lagha[1]

[1] Institute of Aeronautics and Spatial Studies, University of Blida 1, Blida, Algeria
{h.mohamadi,mlagha-aerospatiale}@univ-blida.dz
[2] Department of Software Engineering and Information Technologies,
École de Technologie Superieure, Montreal, Canada
nadjia.kara@etsmtl.ca

Abstract. Designing an adaptive routing protocol for Unmanned Aeronautical Ad-hoc Networks (UAANETs) is very challenging. UAANET routing protocols are vulnerable to several attacks and threats. Thus applying security mechanisms is crucial to ensure the authentication, data integrity and confidentiality. Moreover, when applying formal verification methods to analyze protocols, it is necessary to define a model that formalizes their semantics and security requirements. In this paper, we focus on a hybrid routing protocol, called the Reactive-Greedy-Reactive (RGR), which combines the mechanisms of reactive routing and Greedy Geographic Forwarding (GGF). Our main contribution is to enhance the reactive mode of RGR protocol by incorporating three security mechanisms: a node-to-node authentication approach, a keyed-hash message authentication code and an aggregate designated verifier signature scheme. The results of our formal analysis are validated via three automated verification tools (AVISPA, Scyther and Tamarin).

Keywords: UAANETs · RGR · Security mechanisms
Reactive mode · Formal verification · Automated verification tools

1 Introduction

Unmanned Aerial Vehicles (UAVs) or drones are aircrafts that are flown without a human operator onboard. They are further classified into different categories according to several parameters, such as level of autonomy, aerodynamic configuration, size, endurance, etc. Regarding their application domain, drones can either be used in various civil or military applications [1].

In order to decrease mission delay and increase reliability, UAVs must cooperate with each other using wireless links [2]. When some UAVs communicate, they form a temporary self-organizing multi-hop network, composed of several UAVs and ground control station (GCS), called Unmanned Aeronautical Ad-Hoc Network (UAANET) [3] as shown in Fig. 1.

R. Doss et al. (Eds.): FNSS 2018, CCIS 878, pp. 3–16, 2018.
https://doi.org/10.1007/978-3-319-94421-0_1

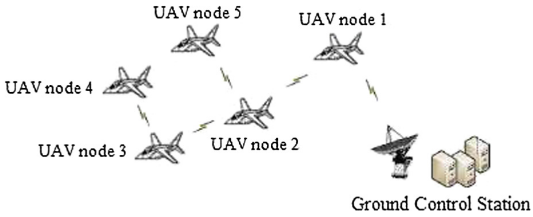


Fig. 1. UAANET network architecture.

Compared to other ad-hoc networks like MANETs (Mobile Ad-hoc Network) and VANETs (Vehicular Ad-hoc network), UAANETs have some specific characteristics, for instance, they are used for real time applications, their node mobility model is usually predictable, but due to some parameters such as environmental conditions, UAVs velocity and formations or mission updates, it can be dynamically modified [4].

UAANETs are characterized by a weak node density. The low number of nodes and their fast mobility enable them to cover a large area rapidly in order to improve the reliability of the network and ensure the scalability [4]. Network connectivity within UAANETs is generally intermittent due to UAVs movements or failures. Numerous link breaks in the network tend to cause frequent topology changes [4].

UAVs and GCS must transmit control and data traffic. Accordingly, an improved routing protocol is needed in order to find routes between nodes, and to accomplish UAANET missions [5]. UAANETs routing protocols can be classified according to their routing strategy into [5]:

- **Proactive routing:** Anticipates the topology changes and establishes a route based on prospected routes. Every node maintains fresh network state information with all nodes.
- **Reactive routing:** A route is created only when a source node wants to send data to a destination node and does not know initially if this node exists in the network.

  Compared to proactive protocols, reactive protocols are much more efficient, faster and adaptive as they change their routing decision according to the actual network conditions.
- **Geographic routing:** The establishment of a route from a source node to a destination is based on node positions rather than IP addresses.
- **Hybrid routing:** Combines two routing mechanisms. The RGR is an example of hybrid routing, which combines the mechanisms of reactive and geographical routings.

1.1 Motivation and Structure of Paper

Since UAANETs are prone to attacks performed by unauthorized entities, mainly due to the cooperativeness between all nodes without a previous security association, and the use of wireless links (e.g. WiFi and cellular networks), which are vulnerable to attacks such as eavesdropping, message replay or denial of service [5]. Thus, applying some security schemes is needed to satisfy UAANETs security requirements (authentication, data integrity, confidentiality).

Additionally, formal verification of security protocols has turned out to be a key issue [6]. Using mathematical techniques, formal verification techniques aim to detect flaws and evaluate the correctness, validity and reliability of a protocol model built from given specifications [6].

In this paper, we apply a formal verification technique on a modified RGR routing protocol. We focus on its reactive mode, which is mainly used in route discovery and data forwarding phases. Furthermore, to fulfill the security requirements of UAANETs such as confidentiality, authentication and data integrity, we have incorporated three security mechanisms, a node-to-node authentication approach, a keyed-hash message authentication code (HMAC) and an aggregate designated verifier signature scheme (Ag_DVS) based on asymmetric key encryption. The security properties of our model are validated via three automated formal verification tools, AVISPA, Scyther and Tamarin-prover.

The rest of this paper is organized as follows. In Sect. 2, we briefly present the existing research works that address the issues of RGR, the hybrid routing protocol and the formal analysis of security protocols. We provide an overview of the RGR protocol in Sect. 3. In Sect. 4, we describe the security properties of our proposed scheme. In Sect. 5, a presentation of the three automated verification tools is given, as well as the results of the formal verification of our model. Finally, in Sect. 6 we conclude the paper.

2 Related Work

The existing tests in literature focus only on improving the execution of RGR protocol, and do not address the enhancement of security and authentication issues.

Since RGR can be seen as a mixture of AODV (Ad-hoc On-demand Distance Vector) with GGF (Greedy Geographic Forwarding) [2,7], several upgrades have been proposed in order to enhance its performances. We can cite the RGR with scoped flooding, delayed route request, and mobility prediction [2]. The authors in [2] have implemented the RGR with scoped flooding and RGR with random way-point (RWP) mobility model using OPNET, the results showed that the protocol overhead, packet delivery ratio (PDR) and packet latency could significantly be reduced. In [8], a number of enhancements of RGR were simulated using NS2, the results showed that the capability of an intermediate node to decide how to forward data packets, whether via reactive mode or switch to GGF, is beneficial to decrease message overhead and improve PDR. The authors in [7] introduced some improvements and proposed a realistic mobility model for simulation in contrast to the unrealistic RWP model.

Additionally, some researches in the field of formal verification have been carried out to analyze certain security proprieties in AODV-based reactive protocols using automated verification tools, like AVISPA, Scyther, Tamarin-prover, ProVerif, Athena, NRL analyzer [9].

In order to analyze the correctness of AODV, a dynamic topological fuzzy timing high level Petri Nets (DT-FPNs) approach has been applied in [10], the results showed that this formal approach is efficient for the verification of routing protocols in MANETs. Bhargavan et al. demonstrated through PROMELA/SPIN that AODV protocol is not loop-free and proposed an enhancement to the protocol. The results of their formal analysis using SPIN and HOL (High Order Method) showed that this improved version is loop-free [11].

The formal analysis of Secure AODV (SAODV) via PROMELA and SPIN model checker in presence of an external attacker showed a serious security issue, which is routing loops, because of not protecting the sender address field [12]. In another study, a simulation-based framework has been developed by Ács et al. to verify the security requirements of distance vector protocols like SAODV. The authors evaluated the performance of such protocol vis-à-vis DoS attacks. They outlined two attacks, the first one aims to deceive the destination by sending a RREQ without updating its hop count field, and the second involves failing to deliver the data packets (spoofing attack) or causing energy consumption due to routing loops [13]. The authors in [14] analyzed and modeled a basic version of SAODV using a calculus and static analysis technique. They proved that their technique showed a similar spoofing attack like in [13].

The authors in [3] provided a formal verification of SUAP (Secure Uav Ad-hoc routing Protocol), a new secure routing protocol for UAANET based on SAODV, to analyze security properties by applying three cryptography techniques (public key cryptography, hash chains and geographical leashes) and using the AVISPA tool. As for our model, we use two more model-checking tools, and apply other cryptography techniques used mainly in MANETs and they are accommodated for UAANETs as detailed in Sects. 4 and 5.2.

3 The RGR Routing Protocol (Overview)

Based on the idea of merging two or several mechanisms in order to leverage their individual schemes, the RGR routing protocol combines the mechanisms of a reactive routing protocol with a greedy geographic forwarding [2]. HELLO messages containing a node's ID and position (altitude, latitude and longitude) are periodically broadcasted in order to keep track of the existence of each neighbor [2]. The major drawback of RGR is that network congestion may happen because of the overhead size and the increased number of control packets due to the frequent topology changes [5].

As in AODV, during the route discovery process when a source node needs to send data packets to a destination which is missing in its routing table [15]. It broadcasts route request packets (RREQ) to all its neighbors via flooding. RREQ packets contain mutable fields (such as hop count) and non-mutable fields (such as IP addresses). Upon receiving the RREQ packet, the destination node sends a route reply packet (RREP) to the source by using unicast messages thorough the precursor nodes (see Fig. 2).

In case of link breakage, the reactive mode can no longer be used until the route has been repaired. RGR switches to greedy geographic forwarding (GGF), in which data packets will be sent to the nearest neighbor toward the destination. In the same time, a RREQ is launched to establish new route [2,5]. The packet may be dropped if there is no reactive route nor a geographically closer neighbor to the destination.

4 RGR-SEC Protocol Scheme

Many solutions have been proposed in literature to secure ad-hoc networks and ensure their consistency [5,16]. In order to ensure the authentication so that only authorized entities are allowed to participate in the execution of our modified protocol model, which is named RGR-SEC, as well as confidentiality and integrity, we apply the following security mechanisms.

It is assumed initially that all members share some password that will serve as a weak shared secret and then used to derive a stronger secret [17]. Prior to the deployment, all parties request time stamped certificates from a trusted server (T). These certificates, that will bind node identities to their public keys, have an expiration time and are signed by the secret key of (T) [5]. Every node is deployed with a public/private key pair and the public key of (T) in order to be able to decrypt the certificates of other nodes.

To ensure data integrity and data origin authentication, we add a keyed-hash message authentication code (HMAC) to the transmitted messages. HMAC is a special one-way hash function, based on the idea of concatenating a message with a shared secret key known only by the sender and the recipient and hashing the result with a cryptographic hash function [18].

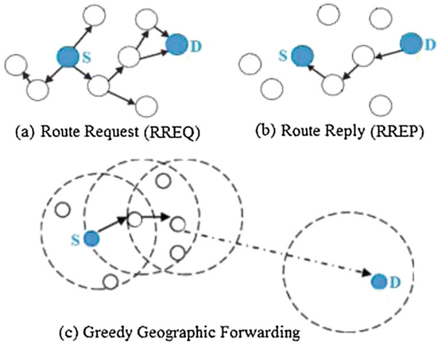


Fig. 2. The reactive and GGF modes of RGR.

By applying hashing twice, HMAC is considered to be more secure and efficient to provide message authentication and data integrity at the same time [18].

A HMAC for a message m with a shared secret key K is created as shown below:

$$HMAC = h\Big((K \oplus opad)||h\Big((K \oplus ipad)||m\Big)\Big) \tag{1}$$

Where:

- h denotes a cryptographic hash function.
- ipad and opad are different padding constants.
- || represents the concatenation.

We also include an aggregate designated verifier signature scheme (Ag_DVS), which is based on asymmetric key encryption, and which has been demonstrated to be efficient for the authentication of routes in reactive protocols [19]. In this scheme, a pair of keys is generated to sign the RREQ/RREP packets. One key is used by the signer and the other by the verifier [20]. The signer's aggregate signature key σ_{SV} is formed by combining the signer's secret key (SK_S) and the designated verifier's public key (PK_V).

$$\sigma_{SV} = h(SK_S.PK_V) \tag{2}$$

Equally, the designated verifier form the same combination of keys and combine it with the message to check whether the signer's aggregate signature key prevented the intruder from tampering with the message contents or not.
Upon receiving a signed route request/reply, nodes in this route can append their own aggregate signatures and XOR them to the message.

However, this proposed approach for securing the RGR protocol is not appropriate for small-sized UAVs that have limited storage and processing capacities nor for much larger networks (generally, UAV missions need the collaboration of 3 to 4 drones [3,5]), because it will induce more computational overhead compared with the normal AODV. Hence, adding nodes positions (altitude, latitude and longitude) and applying both symmetrical and asymmetrical cryptography techniques such as hash functions/chains and private/public key based certificates and signatures is highly resource and time consuming.

5 Results

Formal verification techniques fall into three categories: model checking, equivalence checking, and theorem proving [6]. The first approach tends to confirm whether a system satisfies a given desired or undesired property by using a dedicated tool, whereas the second one verifies if two system models, at different abstraction levels, are equivalent, and the latter applies mathematical methods to prove the correctness of a system [6]. In this paper, the model-checking method will be considered. Figure 3 illustrates its basics.

In order to formally verify the efficiency of our proposed scheme on the reactive part of RGR, we consider four UAVs, denoted as: a source node (A), a destination node (D) and two intermediate nodes (B, C). Then we specify the

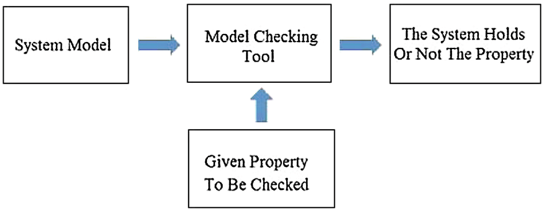

System Model
Model Checking Tool
The System Holds Or Not The Property
Given Property To Be Checked

Fig. 3. The model-checking method.

security proprieties of our formal model in three files: HLPSL (High-Level Protocol Specification Language), SPDL (Security Protocol Description Language) and SPTHY (Security Protocol Theory), which are going to be used by AVISPA, Scyther and Tamarin respectively.

5.1 AVISPA/Scyther/Tamarin Tools

Many automated verification tools have been developed to analyze security proprieties in routing protocols. These tools use the so-called Dolev-Yao intruder model, in which the network is supposed to be under the control of an intruder who can perform any operation, namely it can creates, reads, alters or destroys messages. However, it is assumed, at least initially, that the intruder does not know any information that should be kept secret [21]. In this paper we have used AVISPA, Scyther and Tamarin tools.

AVISPA Tool. AVISPA (Automated Validation of Internet Security Protocols and Applications) is an automated model checker for the verification of security protocols. In order to model and analyze a protocol, AVISPA provides its own role-based High-Level Protocol Specification Language (HLPSL). As depicted in Fig. 4, the given HLSPL is converted to the intermediate format IF, which is verified by four different automatic protocol analysis techniques. OFMC (On-the-Fly Model-Checker), CL-AtSe (Constraint-Logic based Attack Searcher), SATMC (SAT based Model-checking), and TA4SP (Tree Automata based on Automatic Approximations for the Analysis of Security Protocols) [22].

Currently, the only AVISPA's back-end tools that can deal with algebraic properties like the Exclusive-Or and Diffie-Hellman exponentiation are OFMC and CL-AtSe [9].

OFMC. The On-the-Fly Model-Checker (OFMC) uses symbolic analysis techniques to represent the state-space in a demand-driven way (On-the-Fly). It can be employed mutually for the verification of protocols (proving their correctness) and for efficient falsification (fast detection of attacks) [22].

CL-AtSe. The CL-based Model-Checker (CL-AtSe) can translate any security protocol specification written in the IF into a set of constraints (on the adversary's knowledge) [22], which can be effectively analyzed automatically to find attacks by using redundancy elimination and heuristics techniques [9].

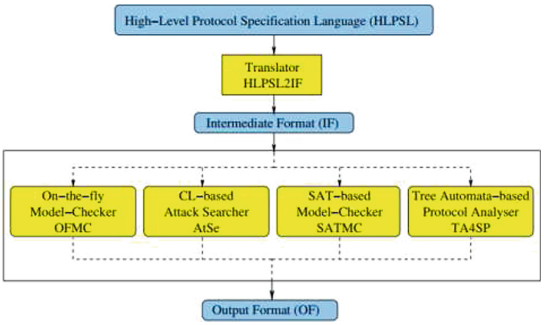

High–Level Protocol Specification Language (HLPSL)
Translator HLPSL2IF
Intermediate Format (IF)
On–the–fly Model–Checker OFMC
CL–based Attack Searcher AtSe
SAT–based Model–Checker SATMC
Tree Automata–based Protocol Analyser TA4SP
Output Format (OF)

Fig. 4. AVISPA tool v.1.1 architecture [22].

Scyther Tool. Scyther is a model checking tool for security protocols and their potential vulnerabilities, which is based on the perfect cryptography assumption and relays on backward search algorithm on trace patterns [23]. Scyther uses an unbounded model checking approach, which aims to demonstrate the soundness of a protocol for all possible behaviors, even in the presence of an adversary [18]. Scyther has its own specification role-based language to describe protocols, roles, sending/receiving events and provide expressions for encryption and hashing, which is called Security Protocol Description Language (SPDL) [23].

Tamarin Tool. Tamarin-prover is a model checker tool, written in Haskell programming language, which extends Scyther's backwards search algorithm, and offers two different modes to construct proofs and verify protocols, namely full automated and interactive modes. It also supports Diffie-Hellman exponentiation and bilinear pairing. Tamarin takes a security protocol theory file (.spthy) as an input to describe protocols and security proprieties, which are specified respectively via multiset rewriting rules and lemmas [24].

5.2 Formal Verification of RGR-SEC Protocol

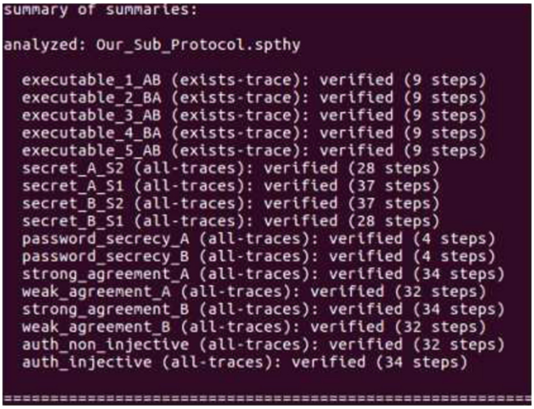
Our protocol is divided into two phases: A node-to-node authentication phase that precedes the real execution of the protocol, where two peers authenticate themselves to each other and agree beforehand on a long-term key from a shared password. Two nodes having a public/private key pair for encryption/decryption respectively can derive a strong key from a weak shared secret [17].

The initiator (A) broadcasts a first message signed by the public key of the responder (B) concatenated with the password P, which contains its IP address and a time-stamp, along with its certificate. (B) decrypts the message and randomly choose a secret S_B to be re-broadcasted to (A). Afterwards, (A) chooses a secret S_A and a random $Challenge_{\text{-}A}$ encrypted by the shared strong secret K_{AB} derived from (S_A, S_B). Equally, (B) sends a $Challenge_{\text{-}B}$ and $Challenge_{\text{-}A}$ to convince (A) that it knows S_A. (A) proves to (B) that it also knows S_B by resending the $Challenge_{\text{-}B}$.

1. $A \rightarrow B$: $\{A,\ N_A\}(PK_B.P),\ Cert_A$
2. $B \rightarrow A$: $\{B,\ N_A,\ S_B\}(PK_A.P)$
3. $A \rightarrow B$: $\{A,\ S_A,\ \{Challenge_{\_A}\}K_{AB}\}(PK_B.P)$
4. $B \rightarrow A$: $\{Challenge_{\_A},\ Challenge_{\_B}\}K_{AB}$
5. $A \rightarrow B$: $\{Challenge_{\_B}\}K_{AB}$

In order to prove the security goals of our sub-protocol specification via Tamarin, we specify three types of goals: secrecy, non-injective agreement and injective agreement. We also add executability lemmas to verify that the model can run to completion. Similarly, for Scyther, we use predefined claim events to specify the security requirements in SPDL specification, namely secrecy, aliveness, Niagree (non-injective agreement), Nisynch (non-injective synchronization) and weak agreement.

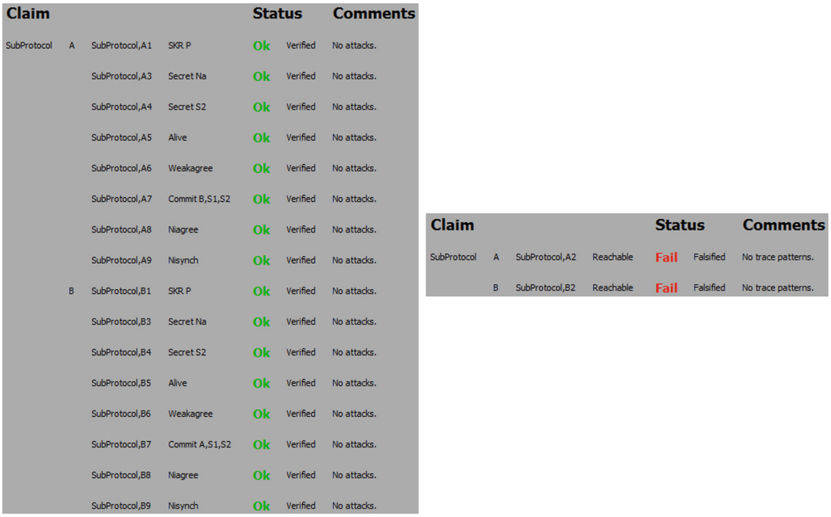
It can be observed from Fig. 5 that all security goals are proved by Tamarin. We can also see that the protocol specifications successfully guarantee all Scyther claims as in Fig. 6(a). A complete characterization of the sub-protocol roles is established to determine representatives (trace patterns) for all possible behaviors. The results show that no trace pattern indicating a possibility of attacks has been found as illustrated in Fig. 6(b).



```
summary of summaries:

analyzed: Our_Sub_Protocol.spthy

  executable_1_AB (exists-trace): verified (9 steps)
  executable_2_BA (exists-trace): verified (9 steps)
  executable_3_AB (exists-trace): verified (9 steps)
  executable_4_BA (exists-trace): verified (9 steps)
  executable_5_AB (exists-trace): verified (9 steps)
  secret_A_S2 (all-traces): verified (28 steps)
  secret_A_S1 (all-traces): verified (37 steps)
  secret_B_S2 (all-traces): verified (37 steps)
  secret_B_S1 (all-traces): verified (28 steps)
  password_secrecy_A (all-traces): verified (4 steps)
  password_secrecy_B (all-traces): verified (4 steps)
  strong_agreement_A (all-traces): verified (34 steps)
  weak_agreement_A (all-traces): verified (32 steps)
  strong_agreement_B (all-traces): verified (34 steps)
  weak_agreement_B (all-traces): verified (32 steps)
  auth_non_injective (all-traces): verified (32 steps)
  auth_injective (all-traces): verified (34 steps)

==============================================================
```

Fig. 5. Tamarin-prover analysis results.

For the second phase, it is assumed initially that each drone is equipped with GPS so that it can obtain its position and all drones have synchronised clocks. The source node (A) initiates the route request process by broadcasting a RREQ packet signed by the shared secret K_{AB} that contains its IP address and current sequence number, $RREQ_{ID}$, destination node's IP address and sequence number. It also appends its position and the time of sending the packet in order to build a geographical leash. And then encrypt them along with the first part by the aggregate signature σ_{AD}. The RREQ also contains hop count field, nodes certificates field and HMAC.



Claim				Status		Comments
SubProtocol	A	SubProtocol,A1	SKR P	Ok	Verified	No attacks.
		SubProtocol,A3	Secret Na	Ok	Verified	No attacks.
		SubProtocol,A4	Secret S2	Ok	Verified	No attacks.
		SubProtocol,A5	Alive	Ok	Verified	No attacks.
		SubProtocol,A6	Weakagree	Ok	Verified	No attacks.
		SubProtocol,A7	Commit B,S1,S2	Ok	Verified	No attacks.
		SubProtocol,A8	Niagree	Ok	Verified	No attacks.
		SubProtocol,A9	Nisynch	Ok	Verified	No attacks.
	B	SubProtocol,B1	SKR P	Ok	Verified	No attacks.
		SubProtocol,B3	Secret Na	Ok	Verified	No attacks.
		SubProtocol,B4	Secret S2	Ok	Verified	No attacks.
		SubProtocol,B5	Alive	Ok	Verified	No attacks.
		SubProtocol,B6	Weakagree	Ok	Verified	No attacks.
		SubProtocol,B7	Commit A,S1,S2	Ok	Verified	No attacks.
		SubProtocol,B8	Niagree	Ok	Verified	No attacks.
		SubProtocol,B9	Nisynch	Ok	Verified	No attacks.

(a)

Claim				Status		Comments
SubProtocol	A	SubProtocol,A2	Reachable	Fail	Falsified	No trace patterns.
	B	SubProtocol,B2	Reachable	Fail	Falsified	No trace patterns.

(b)

Fig. 6. Scyther verification results. (a) Security claims. (b) Characterization.

The notion of hop count update is specified using hash fields, which are always modified before packet forwarding. H_{max} represents the maximum hop count estimated in the network.

When the intermediate node (B) receives the RREQ, it verifies that the certificate has not expired and applies hash function to the present hash field, it XORs its own aggregate signature σ_{BD} to the previous signature, adds its own certificate, and forwards the packet to node (C). The RREQ is uniquely identified by the pair (source address, $RREQ_{ID}$), through which the receiving nodes identify the RREQ packet through which the receiving nodes identify the RREQ packet and discard the message if they receive a duplicate packet or already processed that message [15], as well as preventing replay attacks since certificates containing timestamps of when they were generated, and a time at which they expire are sent alongside.

Otherwise, the destination node (D) processes the packet and generates a RREP packet signed by the shared secret K_{CD}. Similar to RREQ, RREP contains $RREP_{ID}$, a destination sequence number that should be at least equal to the one contained in RREQ in order for the RREP to be forwarded and an expiration time. (D) includes its position, as well as the source node's position, the time of receiving the request and the time of sending the reply. So that the source node compares these values to its current position and computes the distance between itself and the destination [25]. This mechanism has been demonstrated to be efficient against the Wormhole attack [5,25]. Table 1 summarizes the different notations used in the following tuples representing the RREQ and RREP messages.

- $A \rightarrow B$: $\{RREQ_{ID}, A, SEQ_A, SEQ_D, D, HOP_{cnt}, H_{max}, \{RREQ_{ID}, A, RREQ_{Leash}, D\}\sigma_{AD}\}K_{AB}, RREQ_{cert}, HMAC$

- $B \rightarrow C$: $\{RREQ_{ID}, A, SEQ_A, SEQ_D, D, HOP_{cnt}, H_{max}, \{RREQ_{ID}, A, RREQ_{Leash}, D\}\sigma_{AD} \oplus \sigma_{BD}\}K_{BC}, RREQ_{cert}, HMAC$

- $D \rightarrow C$: $\{RREP_{ID}, D, SEQ_D, A, T_{out}, HOP_{cnt}, H_{max}, \{RREP_{ID}, D, RREP_{Leash}, A\}\sigma_{DA}\}K_{CD}, RREP_{cert}, HMAC$

- $B \rightarrow A$: $\{RREP_{ID}, D, SEQ_D, A, T_{out}, HOP_{cnt}, H_{max}, \{RREP_{ID}, D, RREP_{Leash}, A\}\sigma_{DA} \oplus \sigma_{CA} \oplus \sigma_{BA}\}K_{AB}, RREP_{cert}, HMAC$

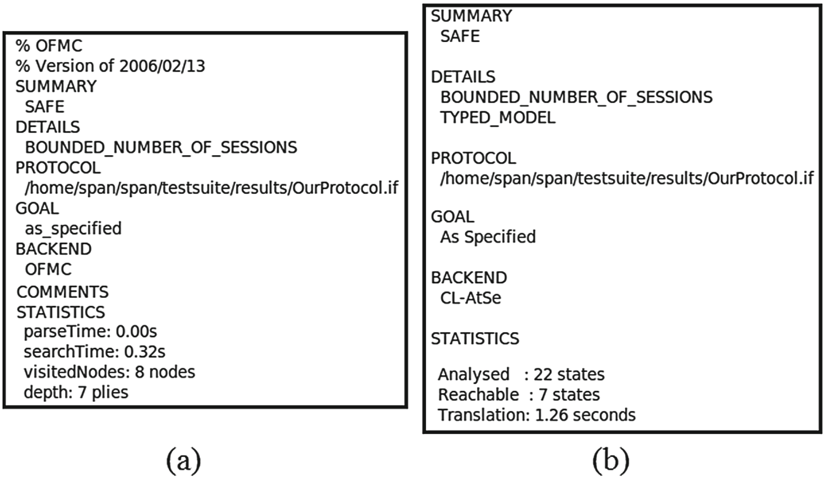
In HLPSL specification, we add a *witness* statement, in which the sender declares that it is witness for a specific information of the message and it is used for a weak authentication property. Additionally, a *wrequest* statement is sent by the receiver, which also serves as a weak authentication of a specific information. Both statements are uniquely identified by their ID in the goal section.

Figure 7 shows that both AVISPA back-ends, OFMC and CL-AtSe could not find any attack (e.g. man-in-the-middle attacks or replay attacks).

Table 1. Terminology table

Notation	Description
A, B, C, D	IP addresses of communicating nodes
$RREQ/RREP_{ID}$	The unique identifier of RREQ/RREP packets
SEQ_A, SEQ_D	Sequence numbers of source and destination nodes
HOP_{cnt}	The distance traveled by RREQ/RREP packets
H_{max}	The highest hop count estimated in the network
$RREQ/RREP_{Leash}$	Geographical leashes field
$RREQ/RREP_{cert}$	Certificates field
T_{out}	The duration wherein the RREP packet is valid
K_{ij}	The shared secret key
$\{Message\}K$	Message encryption with the key K
σ_{ij}	The aggregate signature key
$HMAC$	Keyed-hash message authentication code

The summary of the protocol specification showed that the security goals have been validated, since OFMC and CL-Atse are conceived to find attacks on specified properties and stop once they violated one, for example when an adversary replays or blocks a message.



(a)

```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  /home/span/span/testsuite/results/OurProtocol.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.00s
  searchTime: 0.32s
  visitedNodes: 8 nodes
  depth: 7 plies
```

(b)

```
SUMMARY
  SAFE

DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
  TYPED_MODEL

PROTOCOL
  /home/span/span/testsuite/results/OurProtocol.if

GOAL
  As Specified

BACKEND
  CL-AtSe

STATISTICS

  Analysed   : 22 states
  Reachable  : 7 states
  Translation: 1.26 seconds
```

Fig. 7. AVISPA output. (a) OFMC. (b) CL-AtSe.

6 Conclusion and Future Work

In order to satisfy UAANETs security requirements, an adaptive routing protocol is required. Among these protocols, the RGR that combines the mechanisms of the Greedy Geographic Forwarding and reactive routing is discussed in this paper. We have performed a formal verification of the reactive mode which is primarily used in RGR, by incorporating three security mechanisms, namely a node-to-node authentication approach to establish a secret shared key between two nodes, a keyed-hash message authentication code and an aggregate designated verifier signature scheme based on asymmetric key encryption. The formal verification has been carried out using AVISPA, Scyther and Tamarin tools.

We have run Scyther and Tamarin to check the correctness of the node-to-node key agreement sub-protocol since both of them cannot support the algebraic operator XOR used in HMAC and Ag_DVS. Moreover, we have run two AVISPA back-ends (OFMC and CL-AtSe) to formally verify the security proprieties during the route discovery.

Finally, we can conclude that our protocol specifications are secure regarding the attacks that these model checker tools are designed to find. We have carried our formal verification through an example of 4 nodes. However, the same observations would have been obtained if we added more nodes despite the model being more complex particularly in calculating the hop count hash field and the aggregate key for signing.

As a future work, we aim to formally verify the different properties of other UAANETs routing protocols using other formal verification techniques and tools as well as comparing their energy consumption in terms of both communication costs, computation complexity and thereby the processing power needed to execute them.

References

1. ElKholy, H.M.N., Habib, M.-K.: Dynamic modeling and control of a Quadrotor using linear and nonlinear approaches. M.S. thesis, The American University in Cairo (2014)
2. Li, Y., St-Hilaire, M., Kunz, T.: Enhancing the RGR routing protocol for unmanned aeronautical ad-hoc networks. Technical report SCE-12-01, Systems and Computer Engineering, Carleton University (2012)
3. Maxa, J.-A., Ben Mahmoud, M.-S., Larrieu, N.: Extended verification of secure UAANET routing protocol. In: 35th Digital Avionics Systems Conference, Sacramento, United States (2016)
4. Maxa, J.-A.: Architecture de communication scurise d'une flotte de drones. Rseaux et tlcommunications. Universit Toulouse 3 Paul Sabatier, Français (2017)
5. Maxa, J.-A., Ben Mahmoud, M.-S., Larrieu, N.: Survey on UAANET routing protocols and network security challenges. Ad Hoc Sens. Wirel. Netw. (2017)
6. Câmara, D., Loureiro, A.A., Filali, F.: Formal verification of routing protocols for wireless ad hoc networks. In: Misra, S., Woungang, I., Chandra Misra, S. (eds.) Guide to Wireless Ad Hoc Networks. Computer Communications and Networks. Springer, London (2009). https://doi.org/10.1007/978-1-84800-328-6_8
7. Sharma, P., Yadav, I.: Improving reactive greedy reactive routing in flying ad hoc networks. Int. J. Sci. Eng. Technol. Res. **5**(7), 2276–2281 (2016)
8. Anisha, S.L.: Enhancing RGR routing in unmanned air vehicles networks. Int. J. Sci. Eng. Technol. Res. **5**(7), 2338–2343 (2016)
9. Lafourcade, P., Puys, M.: Performance evaluations of cryptographic protocols verification tools dealing with algebraic properties. In: Garcia-Alfaro, J., Kranakis, E., Bonfante, G. (eds.) FPS 2015. LNCS, vol. 9482, pp. 137–155. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-30303-1_9
10. Xiong, C., Murata, T., Leigh, J.: An approach for verifying routing protocols in mobile ad hoc networks using Petri Nets. In: Proceedings of 6th IEEE Circuits and Systems Symposium on Emerging Technologies: Frontiers of Mobile and Wireless Communication, vol. 2, pp. 537–540 (2004). https://doi.org/10.1109/CASSET.2004.1321944
11. Bhargavan, K., Obradovic, D., Gunter, C.A.: Formal verification of standards for distance vector routing protocols. J. ACM **49**(4), 538–576 (2002). https://doi.org/10.1145/581771.581775
12. Gürdag, A.-B., Çaglayan, M.-U.: A formal security analysis of secure AODV (SAODV) using model checking. In: Proceedings of 8th International Symposium on Computer Networks (2008)
13. Ács, G., Buttyán, L., Vajda, I.: Provable security of on-demand distance vector routing in wireless ad hoc networks. In: Molva, R., Tsudik, G., Westhoff, D. (eds.) ESAS 2005. LNCS, vol. 3813, pp. 113–127. Springer, Heidelberg (2005). https://doi.org/10.1007/11601494_10
14. Nanz, S., Hankin, C.: A framework for security analysis of mobile wireless networks. Theor. Comput. Sci. **367**(1–2), 203–227 (2006). https://doi.org/10.1016/j.tcs.2006.08.036
15. Perkins, C.-E., Belding-Royer, E., Das, S.: Ad hoc on-demand distance vector (AODV) routing. Internet RFCs, pp. 1–38 (2003)
16. Bekmezci, I., Senturk, E., Turker, T.: Security issues in flying ad-hoc networks (FANETS). J. Aeronaut. Space Technol. **9**(2), 13–21 (2016)

17. Liao, L.: Group key agreement for ad hoc networks. Master thesis, Ruhr-University Bochum, Germany (2005)
18. Münch, M.: Integration and verification of a keyed-hash of a message authentication scheme based on broadcast timestamps for NUTS. Master thesis, NTNU, Trondheim (2014)
19. Bhaskar, R., Herranz, J., Laguillaumie, F.: Efficient authentication for reactive routing protocols. In: 20th International Conference on Advanced Information Networking and Applications, vol. 2, pp. 57–61 (2006). https://doi.org/10.1109/AINA.2006.162
20. Vishesh, K., Verma, A.: Formal verification of authenticated AODV protocol using AVISPA. Int. J. Comput. Appl. **50**(19), 38–43 (2012). https://doi.org/10.5120/7914-1179
21. Meadows, C.A., Meadows, C.A.: Formal verification of cryptographic protocols: a survey. In: Pieprzyk, J., Safavi-Naini, R. (eds.) ASIACRYPT 1994. LNCS, vol. 917, pp. 133–150. Springer, Heidelberg (1995). https://doi.org/10.1007/BFb0000430
22. The AVISPA Team: AVISPA v1.1 User Manual (2006)
23. Pfeffer, K.: Formal verification of a LTE security protocol for dual-connectivity. M.S. thesis, KTH Royal Institute of Technology, Stockholm, Sweden (2014)
24. The Tamarin Team: Tamarin-Prover Manual: Security Protocol Analysis in the Symbolic Model (2017)
25. Perrig, A., Johnson, D.-B.: Packet leashes: a defense against wormhole attacks in wireless ad hoc networks. In: IEEE INFOCOM, pp. 1976–1986 (2003). https://doi.org/10.1109/INFCOM.2003.1209219

QoS-Based Sequential Detection Algorithm for Jamming Attacks in VANET

Fatma Salem(✉), Yassin Elhillali, and Smail Niar

University of Valenciennes, CNRS UMR 8201-LAMIH, 59313 Valenciennes, France
{fatma.salem,yassin.elhillali,smail.niar}@univ-valenciennes.fr

Abstract. As a key component of the future Vehicle-to-anything (V2X) communication technology, Vehicular Ad hoc Network (VANET) has a great potential of enabling real-time traffic safety and efficiency applications for people on roads. Therefore, attacking and misusing such network could cause destructive consequences. Wireless communication in VANET is based on IEEE 802.11p-based DSRC standard. Due to its inherited distributed contention resolution mechanism, the MAC protocol in IEEE 802.11p is more susceptible to jamming attacks. While preventing jamming attacks in VANET is not feasible, due to its unbounded scalability, detecting such attacks is primordial. First we develop optimization methodology for IEEE 802.11p MAC which defines its stability region under normal network conditions, this will allow us to determine detection threshold value to distinguish normal operation and attacks. Second, we implement the sequential detection of change method along with the developed methodology and we propose QoS-based Sequential Detection Algorithm (QoS-SDA). The important performance characteristics of QoS-SDA are accuracy and speed, while jamming attacks are detected with low probability of false alarms. Finally, we provide comprehensive analytical and simulation analyses to prove the validity of the develop methodology and the efficiency of the proposed algorithm.

Keywords: Security · Jamming · VANET · IEEE 802.11p · MAC
V2X

1 Introduction

Vehicle-to-anything (V2X) is emerging as a promising communication technology with great potential of supporting a variety of novel applications in Intelligent Transportation System (ITS) to improve traffic safety and efficiency for people on roads. As defined by the Third Generation Partnership Project (3GPP) group, V2X is aiming to enable Vehicle-to-Vehicle (V2V), Vehicle-to-Infrastructure (V2I) and Vehicle-to-Pedestrian (V2P) communications, which promises to eliminate 80% of the current road crashes [1]. As a key component of V2X, supporting V2V and V2I communications, Vehicular Ad hoc Networks

R. Doss et al. (Eds.): FNSS 2018, CCIS 878, pp. 17–32, 2018.
https://doi.org/10.1007/978-3-319-94421-0_2

(VANETs) play an important role in enabling life-critical safety applications, such as cooperative collision warning, collision avoidance and road conditions (e.g., slippery road) [2]. Hence, security is the most required feature for VANET since attacking and misusing such network could cause destructive consequences.

The communication in VANET is based on the Dedicated Short-Range Communications (DSRC) which is achieved over reserved radio spectrum band allocated in the upper 5 GHz range. The main enabling communication standard in DSRC is IEEE 802.11p. In the Medium Access Control (MAC) layer of IEEE 802.11p, different Quality of Service (QoS) classes are obtained by prioritizing the data. Therefore, application messages are categorized into different Access Classes ACs, with AC_0 has the lowest and AC_3 the highest priority [3]. IEEE 802.11p MAC layer possess various vulnerabilities to Denial of Service (DoS) attacks. Such attacks are a vexing problem in all wireless networks, but they are particularly threatening in VANET. Jamming is one kind of DoS in which the jammer can fully or partially prevent legitimate nodes from accessing the network.

1.1 Related Works

In order to detect jamming in conventional wireless networks (i.e., mobile, sensor networks), different detection methods have been proposed in the literature. They are mainly based on observing packet/network measures, such as packet delivery ratio (PDR), signal strength (SS) and carrier sensing time (CST) in normal and abnormal (jamming) network conditions. In [4,5], jamming attacks are evaluated at the packet level utilizing packet send/delivery ratio, while [6,7] proposed network level detection methods for arbitrary jamming attacks. However, packet/network level measures do not directly reflect Qos imposed in time-critical applications. For instance, a high packet delivery ratio does not necessarily guarantee that the required latency for the delivery of time-critical messages is satisfied. Hence, these methods are not feasible for VANETs in which time-critical applications are of a fundamental importance.

Unfortunately, while research on jamming attacks in conventional wireless networks is active and prolific, we have seen very few efforts specifically targeting vehicular networks. Radio Frequency jamming in VANETs were studied experimentally in indoor and outdoor in [8]. The authors of [9] proposed a MAC-based detection method targeting only safety applications. By imposing verification check, an algorithm to detect the malicious node in VANET is proposed in [10]. However, the reported jamming detection methods in VANET focus on a particular jamming attack. In addition, they investigate only the case of communication between two nodes and no medium access contention is considered. Obviously, in order to achieve a high detection efficiency, the reference value (i.e., detection threshold) that is used in differentiating a jamming attack and normal network conditions should be accurate. However, this can not be achieved without the consideration of medium access contention mechanism which is generally ignored in existing studies.

1.2 Motivations and Contributions

While preventing jamming attacks in VANET is not feasible due to its unbounded scalability, the detection of such attacks is of paramount importance. In this work, we propose a jamming detection algorithm that is able to detect different types of jammers. There are two key observations that drive our detection method.

1. The QoS requirements imposed by different classes ACs and how they are intimately interwoven with the contention mechanism of IEEE 802.11p MAC. This leads to develop an *optimization methodology* which allows us to decide on the detection threshold value.
2. Vehicles which are moving in free-flow conditions form clusters. The stability in the clusters' links provide support to the QoS requirements in each class AC, and consequently the threshold value in (1) can be obtained accurately.

Motivated by the two observations, we propose a QoS-based Sequential Detection Algorithm (QoS-SDA) that can effectively detect jamming attacks, while false detections occur infrequently. Our main contributions in this work include

1. We develop an *optimization methodology* for IEEE 802.11p MAC which ties QoS requirements of ACs with the contention mechanism design parameters.
2. Utilizing the methodology in (1), we define IEEE 802.11p stability region from which we determine an accurate detection threshold value.
3. We implement the sequential detection of change method along with the developed methodology and we propose the QoS-SDA.
4. We provide analytical and simulation analyses to prove the validity of the methodology and the efficiency of the algorithm.

2 Jamming Attacks in VANET

2.1 Vulnerability of IEEE 802.11p MAC to Jamming Attacks

Due to its inherited distributed contention resolution mechanism, the MAC protocol in IEEE 802.11p is more susceptible to jamming attacks. IEEE 802.11p uses the enhanced distributed channel access (EDCA) as MAC method. The EDCA uses CSMA with collision avoidance (CSMA/CA) while traffic prioritization is provided by four Access Classes with AC_0 has the lowest priority and AC_3 the highest priority [3]. For each newly generated packet, the vehicle senses the channel before it starts the transmission. If the channel is sensed idle for a time period greater than or equal to an arbitration interframe space (AIFS), the packet can be directly transmitted. If the channel is busy or becomes busy during AIFS, the vehicle must wait until its backoff timer decreases to zero to be able to transmit again, as shown in Fig. 1. If a malicious attacker deliberately transmits interfering random packets during AIFS, the vehicle will sense the channel busy and then starts the backoff process. The Backoff Timer (BT)

Table 1. EDCA contention parameters

AC	ACI	CW_{min}	CW_{max}	$AIFS$
Background	AC_0	15	1023	9
Best effort	AC_1	15	125	6
Safety	AC_2	7	15	3
Safety-of-life	AC_3	3	7	2

is chosen randomly from a discrete uniform distribution and drawn from a Contention Window (CW) in the interval $(0, CW_{min})$. Moreover, the MAC protocol of 802.11p is a stop-and-wait protocol and therefore the sender vehicle awaits an acknowledgment (ACK) from the receiver. If no ACK is received due to being deliberately discarded by a malicious attacker, the sender will falsely believe that there exists congestion and then enter a retransmission state. For every retransmission attempt, the BT value will be doubled from its initial value until reaching CW_{max} and the packet who reached the maximum number of attempts will be rejected out of the system. Hence, if the attacker launched successive constant jams, the vehicle being jammed would experience an increased rejection rate due to constantly sensing a busy channel and virtually stop transmissions. The contention parameters as adapted from [3], is shown in Table 1. While AC_3 and AC_2 have strict Qos requirements, such requirements are loosened for AC_1 and AC_0, as we shall discuss further in Sect. 4. Therefore, we only consider three types of messages in a vehicle; AC_3 for Safety-of-life, AC_2 for Safety messages, and we lump both AC_1 and AC_0 into one class AC_1 for Non-safety messages.

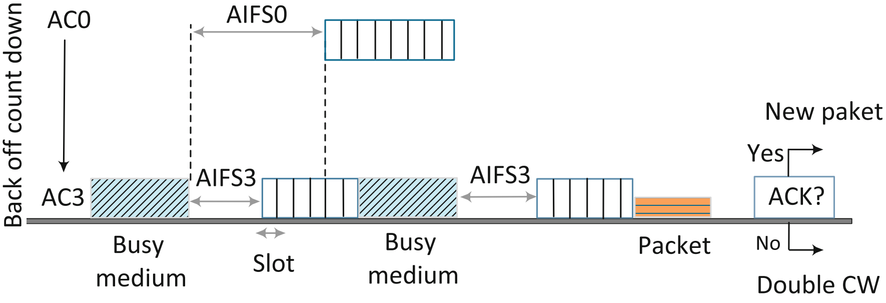


Fig. 1. EDCA backoff procedure

2.2 Attacker Model

There are three common types of jammers in wireless networks [11]:

- *Constant jammer:* transmits random interfering packets to make the channel always busy. Whenever a legitimate node attempts to access the channel, it finds the channel busy then enters to backoff process. If the constant jammer launches successive attacks it could lead to a completely denial of service.

- *Random jammer:* operates according to a random sequence of active (ON) and sleep (OFF) periods. It sleeps regardless of any activity on the network, and during the jam interval, it acts as a constant or intelligent jammer.
- *Intelligent jammer:* is a protocol-aware jammer that conforms to legitimate transmissions, hence it is activated when it senses activity in the channel which makes it less likely to detect.

3 System Model and Problem Formalization

3.1 Network Architecture

Due to different vehicular traffic scenarios, i.e., regular, dense or sparse, vehicles in VANETs which are moving on the same directed pathway and maintain V2V connectivity form clusters [12]. We consider clustered network architecture, the components in the architecture are Cluster Heads (CHs), Cluster Members (CMs) and a V2I backbone network of CHs and Roadside Units (RSUs). The neighboring vehicles within the range of a CH become a CM and directly communicate with their CH via IEEE 802.11p. The CH then aggregates the data from its local CMs and send it to the RUS in its range in periodic time intervals. The RSU is then responsible for disseminating the received data to the other RUSs in the network. Even though the average speeds of vehicles in different paths vary, speeds of vehicles in the same directed path have almost identical mean and variance, moreover as these vehicles move across in the same path, the change in the neighborhood is small. This eliminates the effect of mobility and consequently reduces the need for periodic election of CHs which makes the links among vehicles within the same cluster to be more stable. Thus, links stability provides us an opportunity to define IEEE 802.11p stability region which will be further used to define a threshold value on the per cluster input rate.

3.2 Problem Formalization

Consider a cluster C comprised of a CH and a set S of CMs, $S = 1, 2, \ldots, n$. Then, a number $m : m < n$ of RSUs are displaced in the network. Within the cluster C each CM generates messages for the three classes $AC_i, i = 1, 2, 3$. Each class AC_i imposes different QoS requirements (e.g., delay and rejection rate) while presenting traffic rates $\lambda_i, i = 1, 2, 3$ which vary dynamically. These variations in the rate λ_i along with contention resolution in 802.11p induce changes in the rate accessing the CH, i.e., λ_{Ci}. The main problem can be formulated as follows:

- Given the rates $\lambda_i, i = 1, 2, 3$ and given the statistical descriptions of the data traffics, then there exists an upper and a lower bound on λ_{Ci} such that the QoS requirements for each access class AC_i is satisfied.

- If the changes in λ_{Ci} are within the cluster stability region, then the cluster data rate is maintained; meaning that the QoS of each AC_i is satisfied and the resulted rejection rate is due to normal operation (no jamming). Otherwise, the rate λ_{Ci} in no longer maintained and the rejections are highly probable due to a jamming attack.
- To detect the changes in λ_{Ci}, a proposed QoS-based Detection Algorithm will be devised that traces consecutive λ_{Ci} changes and declares a jamming attack whenever this change falls outside the predefined bounds interval.

4 Optimization Methodology for IEEE 802.11p Configuration

4.1 Traffic Classes and Quality of Service Criteria

Obviously, a jamming attack results in an increase in the number of packet collisions observed in the affected cluster and consequently leads to a rejection among the transmitted packets. An important measure which differentiates between the rejection under normal operation and the rejection due to jamming is the lower bound on the fraction of the successfully transmitted traffic L_s. The cluster stability region is obtained via an *optimization methodology* which relates the traffic rate maintainance, in terms of satisfaction of L_s and the other QoS, with the contention mechanism design parameters. Below, we identify QoS for each class as taken from [13].

QoS of Safety-of-life Class, AC_3: Each generated Safety-of-life message imposes a strict upper bound U_d of 100 *ms* on the transmission delay it may tolerate. The communication range is between 50 and 300 meters. In addition to have 99.9% probability of successful transmission, i.e., $L_s \geqslant 0.99$.

QoS of Safety Class, AC_2: An upper bound, U_d of 1000 ms on the delay per message along with 99.9% probability of successful transmission apply here as well with a communication range that may extend up to 1000 m.

QoS of Non-safety Class, AC_1: Non-safety messages do not impose constraints on transmission delays, while they have shorter range (up to 90 m). It is desirable, however, that delays be finite with high transmission reliability.

4.2 EDCA Quality of Service Support

The Contention Window CW, in Table 1, is a basic design parameter that is chosen so that the stability region of IEEE 802.11p is maximized while maintaining the QoS requirements of each traffic class even under congestion conditions. This leads to the following Lemma for which the proof is in the Appendix.

Lemma 1. *Given a lower bound on the successful transmissions L_s and an upper bound U_d on the transmission delay, the Contention Window size (CW) should be determined from the following constrained optimization problem: Find the CW value such that the input rate λ is maximized while the fraction of the* *successfully transmitted traffic S remains greater than or equal to L_s. That is, the system throughput, which attains the stability region, at this CW value is*

$$\lambda^*_{U_d,Ls}(CW) = \sup\left(\lambda\colon S \geq L_s\right) \tag{1}$$

We applied this optimization method for the three ACs each with its defined CW value. Table 2 reports the obtained results for Poisson input rate $\lambda \in [0.1, 0.4]$.

Table 2. Traffic stability region

CW_{AC}	U_d	L_s	λ	S
$CW_{AC_3} = 3$	100	0.99	0.10	0.9961
			0.15	0.9910
			0.20	0.9777
			0.25	0.9062
			0.30	0.8151
			0.35	0.7417
			0.40	0.6411
$CW_{AC_2} = 7$	1000	0.99	0.10	1.0000
			0.15	1.0000
			0.20	0.9971
			0.25	0.9931
			0.30	0.9679
			0.35	0.8114
			0.40	0.7260
$CW_{AC_1} = 15$	NA	0.99	0.10	1.0000
			0.15	1.0000
			0.20	1.0000
			0.25	1.0000
			0.30	0.9970
			0.35	0.9954
			0.40	0.7781

From Table 2 we observe that for a fixed CW, the attainable S value for $\{\lambda^*_{U_d,Ls}\}$ is an increasing function of U_d. An interesting observation is that only for small rates $(0.1, 0.15)$, 802.11p meets the required reliability when a high delay constraint is imposed. With less constraint and for small rates in AC_2, the fraction S meets the lower bound L_s. While, for AC_1 with no constraint and for rates $(0.1, 0.35)$, the fraction S almost equals one. Hence, we subsequently select on the upper and lower bounds on the cluster rate that to be monitored

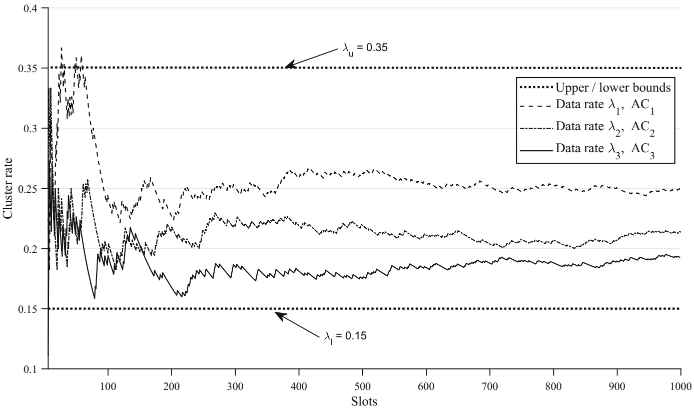


Fig. 2. Cluster data rate under normal network conditions; no jamming attack.

as, $\lambda_u = 0.35$ and $\lambda_l = 0.15$. The detection algorithm will monitor λ_u to λ_l shifts and declares a jamming attack whenever the cluster rate shifts below λ_l

Figure 2 results from simulating the clustered network with the basic EDCA parameters in Table 1. The figure illustrates the traffic rates $\lambda_i, i = 1, 2, 3$ for the classes ACs under normal network operation (no jamming). As shown, the rates are maintained within the stability bounds ($\lambda_u = 0.35, \lambda_l = 0.15$). These results corroborate Lemma 1 and validate the obtained analytical results in Table 2.

5 QoS-Based Sequential Detection Algorithm (QoS-SDA)

In the following we propose a QoS-based Squential Detection Algorithm (QoS-SDA) which implements the developed *optimization m* along with the sequential detection of change method that was first presented in Bansal and Papantoni-Kazakos [14]. The proposed algorithm operates sequentially on the cluster rate λ_{C_i} utilizing a reflective barrier at 0 and a decision threshold ζ. The algorithmic operational value T_n is updated only at the beginnings of frames $\{n_i\}_{i \geqslant 0}$. After initialization with T_0, the algorithm operates in the following three phases:

1. *Observation phase:* Given the rate $\lambda_i, i = 1, 2, 3$ of the access classes ACs with known distribution P_i, let $P_i(m_1^n)$, where $m_1^n = (m_1, \cdots, m_n)$ denote its nth dimensional distribution in frames. Then, the algorithm starts observing the data arrival sequence as follows; m_1 arrivals occur in the first frame, and so on, with m_n data arrivals lie in the nth frame.
2. *Updating phase:* Allowing adaptations only at the beginnings of frames and for stationary distribution P_i, the QoS-SDA takes the general form

$$T_n(m_1^n) = max\left[0, T_{n-1}(m_1^{n-1}) + log\left(\frac{P_l(m_n \mid m_1^{n-1})}{P_u(m_n \mid m_1^{n-1})}\right)\right] \tag{2}$$

3. *Detection Phase:* The algorithm continues updating its operational value until it stops the first time n when it crosses the decision threshold ζ. Then it is decided that a shift in the cluster rate is outside the bound interval, i.e., $\lambda_i < \lambda_l$ and a jamming attack has occurred.

5.1 QoS-SDA for Poisson Model

The traffic accessing the CH, while not Poisson it can be closely approximated by a Poisson process with rate λ_{Ci} packet/time unit. Since Poisson is stationary memoryless process, QoS-SDA utilizes no memory in this case, hence the conditional distributions in (2) then collapse. Let L denote the frame length in slot units and define N_n to be the number of data arrivals within the nth frame from the beginning of time. Then, QoS-SDA updates its operational value as follows

$$T(n) = max\left[0, T(n-1) + L\left((\lambda_u - \lambda_l) + N_n\, log\left(\frac{\lambda_u}{\lambda_l}\right)\right)\right] \tag{3}$$

Even though no memory is required in (3), the decision threshold value ζ may not be a positive integer. To circumvent this difficulty, we define the constant $\Lambda_{u,l} \triangleq (\lambda_u - \lambda_l)/log\left(\frac{\lambda_u}{\lambda_l}\right)$. Since the cluster rate is bounded by rational values, i.e., (λ_u, λ_l), we may further define this constant by two integers b and t

$$\Lambda_{u,l} = b/t, \quad \text{for two integers } b \text{ and } t : b < t \tag{4}$$

Using the expression in (4) with appropriate scaling by $\mathbb{1}(\lambda)$ where

$$\mathbb{1}(\lambda) = \begin{cases} 1, & \text{if } \lambda_l < \lambda_u \\ 0, & \text{if } \lambda_l > \lambda_u \end{cases} \tag{5}$$

and with initial $T(0) = 0$, the QoS-SDA in (3) then transforms to

$$T(n) = max\left[0, T(n-1) + (-1)^{\mathbb{1}(\lambda)}\left(N_n t - Lb\right)\right] \tag{6}$$

The following Pseudocode summarizes the three phases for QoS-SDA.

Algorithm. QoS-based Sequential Detection Algorithm (QoS-SDA)

```
    Initialization
Define the detection rate interval $(\lambda_u, \lambda_l)$
Based on $(\lambda_u, \lambda_l)$, define the integers $t, b$ such that $\lambda_l < \frac{t}{b} < \lambda_u$.
Select a decision threshold $\zeta$ , $\zeta > 0$
Set the operational value $T(0) = 0$
```

Phase 1 – Observation phase

```
procedure ObservationPhase
    Let at some slot, $\lambda_u$ be decided as just starting.
    Let the generating process of $\lambda_u$ to be Poisson.
    Observe the arrivals in frame length $L$.
    Count the number of arrivals in $nth$ frame $(N_n)$, from the beginning of time.
end procedure
```

Phase 2 – Updating phase

```
procedure UpdatingPhase
    for each frame do
        update $T(n)$ as follows
        $T(n) = max\left[0, T(n-1) + (-1)^{\mathbb{1}(\lambda)}\left(N_n t - Lb\right)\right]$
    end for
end procedure
```

Phase 3 – Detecting phase

```
procedure Detecting phase
    if $T(n) \geqslant \zeta$ at $n$ then
        Stop at time $n$
        Cluster rate $\lambda_{Ci} < \lambda_l$
        Declare a jamming attack detection
    end if
end procedure
```

5.2 Decision Threshold Selection

QoS-SDA induces correct detection decisions as well as false alarms whose relative relationship is controlled by the value of the selected threshold ζ. Thus, the performance of the algorithm is basically characterized by two probability measures:

1. $P_D(n)$, Detection probability: The probability that a shift in cluster rate below λ_l is decided before or at time n given that $\lambda_u \rightarrow \lambda_l$ shift has occurred.
2. $P_{FA}(n)$, False Alarm probability: The probability that before or at time n it is decided that cluster rate has shifted below λ_l, while $\lambda_u \rightarrow \lambda_l$ never changed.

As functions of n, these probabilities represent correct detection and false alarm curves. These curves can then be used for the appropriate selection of the decision threshold ζ. Qualitatively speaking, we are seeking a threshold value for relatively small n sample sizes such that the probability $P_D(n)$ is sufficiently large, while $P_{FA}(n)$ is be below a specified desirable level. Toward this end, we evaluate QoS-SDA for several given threshold values, then we compare the correct detection and false alarm curves induced by the algorithm at these threshold values to decide on the appropriate threshold value. Figure 3 shows the behavior of the detection and false alarm curves for four threshold values. From the figure we notice how the two curves are decreasing with increasing a threshold value. We also observe the better performance for the threshold value 200 for time values $n \leqslant 80$. This is true since larger differences in the false alarm set at different threshold values, i.e., $\{P_{FA}^{\zeta_i}(n) - P_{FA}^{\zeta_j}(n); i \neq j\}$ represent improved algorithmic performance.

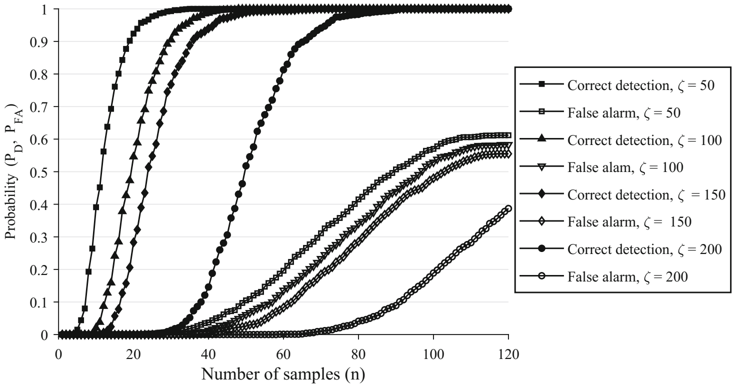


Fig. 3. Power and false alarm curves for different threshold values.

By selecting the decision threshold value (200), and utilizing the observed data for Non-safety class AC_1, the sequential operation of QoS-SDA is depicted in Fig. 4. The figure shows two cases; the first in the presence of Constant jamming attack which was successfully detected and the latter in the absence of attack in which the algorithm continues operating sequentially.

6 Simulation Model and Results

The simulations were performed using network simulator in MATLAB that we developed with the basic IEEE 802.11p MAC layer operation as previously explained in Sect. 2. The used EDCA parameters are those in Table 1 with the attempt limit set to seven attempts. We considered the Poisson Model and focused on a single cluster. In addition, we modeled the data rates $\lambda_i, i = 1, 2, 3$

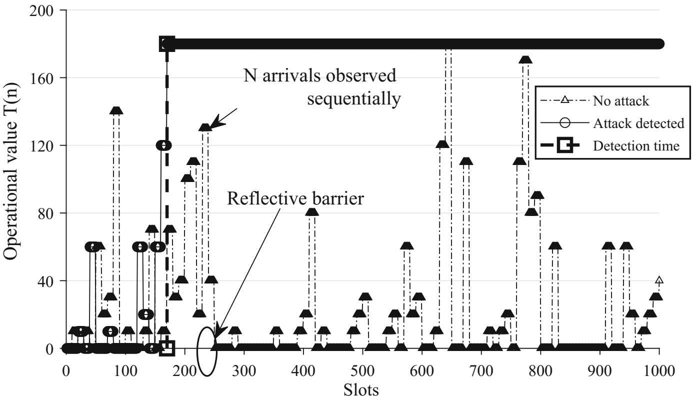


Fig. 4. Time evolution of QoS-based Sequential Detection Algorithm (QoS-SDA).

transmitted by each CM as exponentially distributed in frame lengths. We simulated QoS-SDA with monitored cluster rates within the detection interval ($\lambda_u = 0.35, \lambda_l = 0.15$), selected frame length L equals to 20 time slots and with the integers b and t values set to 11 and 50 respectively. In order to let the jammer have time to react, specifically Random jammer, we selected the average message length equals to a frame length. Following the threshold selection method explained in Sect. 5, algorithmic decision threshold was selected to be 200.

Figure 5 shows different manifestations of the jamming attacks mentioned in Sect. 2, in the three Sub-figures the attacker launches the attack in the first 100 slots with rate 0.1 packet/slot, the attack rate was set to this small value to measure the ability of the algorithm to track and detect very small changes in the normal network conditions. Sub-figure (a) shows the Constant jammer, where the attacker transmits continues random packets with rate 0.1 packet/slot. The three traffic rates have been affected with more noticeable rate drop for AC_3. This is mainly due to the small CW value used in AC_3 class, more discussion on CW values and their effect on the detection probability is in the sequel.

Sub-figure (b) shows the Random jammer where a packet arrives in each of the time slots of the ON state, following a Bernoulli (0.5) distribution. In Sub-figure (c), an intelligent jammer, who is aware of the target communications, attacks the legitimate transmissions of the messages in AC_2 and AC_3 utilizing the corresponding contention parameters to inject its interfering packets. As we can see, the algorithm only needed a few samples, at this small value of attacking rate for the three jamming types, to detect the cosponsoring attack. Almost all the detection times, where the detection threshold λ_l has been crossed, fall in the first 100 slots where the attack started.

Detection probability and the detection delay are used as metrics to evaluate QoS-SDA. To illustrate the influences on the attack rate and contention window size CW, we consider the Intelligent jammer as it conforms to the contention

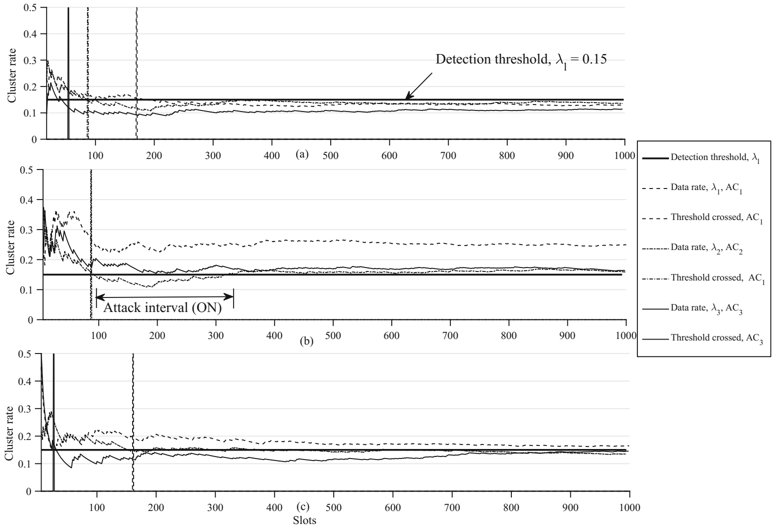


Fig. 5. The algorithmic performance under different attacker models. (a) Constant jammer (b) Random jammer (c) Intelligent jammer

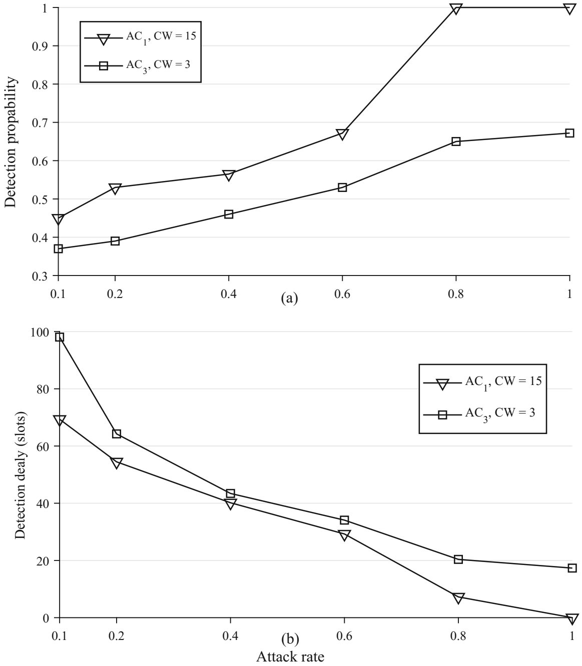


Fig. 6. (a) Detection probability, (b) detection delay as a function of the attack rate.

mechanism in IEEE 802.11p MAC. Figure 6 depicts the detection probability and the detection delay verses the attack rate for different contention window sizes. In Sub-figure (a), when $CW = 15$, QoS-SDA performs better as the attack rate increases comparing with the case when $CW = 3$. This decrease in detection probability for AC_3, as compared to AC_1, is due to the fact that shorter contention cycles and shorter AIFS times would produce more collisions and force the packets to reach the maximum number of attempts quickly and hence abandon the system early. As a result, this induces more rejections and the monitored rates (λ_u, λ_l) are becoming significantly close which affects the detection decisions in QoS-SDA. When the attack rate reaches 1, i.e., the malicious attacker jams all the activity in the transmission channel, QoS-SDA can detects up to 70% of attack cases for AC_3 class with $CW = 3$, while it detects almost all the attack cases for AC_1 with $CW = 15$. In Sub-figure (b), we plot the detection delay verses the jamming attack rate. As a consequence of the discussion in Sub-figure (a), the detection delay is a decreasing function of the attack rate with obviously improved detection performance for traffic class AC_1 with $CW = 15$, primarily due to the larger CW value and long AIFS time comparing with AC_3.

7 Conclusion

Jamming attacks are very serious of risk for Vehicular Ad hoc Networks (VANETs) which play an important role in enabling life-critical safety applications. In this work, we provided an in-depth study on the vulnerabilities of IEEE 802.11p MAC, the enabling communication standard in VANET, to jamming attacks and we proposed the QoS-based Sequential Detection Algorithm (QoS-SDA) to detect jamming attacks. The algorithm operates sequentially on observed data sequence in the stability region of IEEE 802.11p MAC and declares a jamming attack whenever a shift in this data sequence falls outside the predefined stability region. In order to define the stability region of IEEE 802.11p MAC, an optimization methodology for IEEE 802.11p MAC configuration under normal operation is developed. The simulation, for different jamming attacks, verified the accuracy and the efficiency of QoS-SDA to detect the attacks even under small attacking rate.

Appendix: Proof of Lemma 1

Let us define

- CW: The contention window size.
- $P_m(\mu, \sigma^2)$: The distribution of the traffic with mean μ and variance σ^2.
- P: The probability of successful transmission.
- λ: Poisson arrival rate.
- k: The expected number of packets that are successfully transmitted during a time interval l given that it started with the transmission of m packets.

When no constraints (i.e., delay or successful transmission bounds) on the transmitted traffic are imposed, the following definition of throughput is meaningful to capture IEEE 802.11p MAC algorithm stability

$$\lambda^* = \sup \big(\mu \colon \mu = \alpha\big) \tag{7}$$

Let us define the output rate α such that $\alpha = \frac{1}{n}\sum_{i=1}^{n}\beta_i$, in [15] it has been proven that the output process of IEEE 802.11p under stable conditions tends to follow a Bernoulli distribution. Hence

$$\{\beta_i\}_1^n \sim \text{Brn} = \begin{cases} P(\beta_i = 1) = P & \text{if the } i^{th} \text{ slot is a success slot} \\ P(\beta_i = 0) = 1 - P & \text{otherwise} \end{cases} \tag{8}$$

The only parameter of the Bernoulli output in (8) is P. To define P let us consider the following scenario: Let m represents the total number of arrivals (packets) in a time interval l with rate μ. If the contention resolution in IEEE 802.11p, induced only a single successful transmission, then the probability of this event is $P = CW/l$. However, under stable operation of the algorithm this value approaches the input rate i.e., $P = \mu$ and consequently (7) holds.

Now let m to increase and the quantities μ and σ^2 in the arrivals' distribution P_m simultaneously to decreases so that

$$m\mu = \lambda \quad \text{for } \lambda > 0, m \gg 1 \tag{9}$$

the latter expression is the Poisson theorem, then P_m converges in distribution to Poisson process, i.e., $P_m \longrightarrow Pois(\lambda)$. Give Poisson rate λ and in the presence of constraints, the expected number of packets transmitted in the first slot of a time interval l is λCW and therefore the fraction of packets that are successfully transmitted S during l is $S = k/\lambda CW$. In [15], recursions for computing the quantity k have been found. In Poisson process, the arrival points in an interval are uniformly distributed. Hence, if a fraction S of the packets are successfully transmitted it means that S is also the fraction of the interval resolved. Therefore, $(k/\lambda CW)CW = k/\lambda$ represents the average portion of the resolved interval, which takes on the average N slots to be resolved. Thus, the algorithm remains stable, even under congestion conditions, whenever it is able to resolve collisions at the rate in which the arrival process progresses in time, that is

$$N \leqslant \frac{k}{\lambda} \tag{10}$$

(10) defines the maximum value on the input rate λ at this specific CW value so that IEEE 802.11p throughput is maximized while the successfully transmuted packets are bounded by S; thus, the statement in Lemma 1 is a consequence of this.

References

1. 3GPP. Study on LTE-Based V2X Services (Release 14), Technical Specification Group Services and System Aspects (TSG SA), 3GPP TR (2016)
2. Vehicle safety communications-applications (VSC-A). Final report CAMP Vehicle Safety Communications 2 Consortium, Washington, D.C., USA (2011)
3. IEEE: IEEE Standard for Wireless Access in Vehicular Environments (WAVE)-Multi-Channel Operation. IEEE Std 1609.4, pp. 1–94 (2016)
4. Xu, W., Trappe, W., Zhang, Y., Wood, T.: The feasibility of launching and detecting jamming attacks in wireless networks. In: 6th ACM MobiHoc, pp. 4657, USA (2005)
5. Bayraktaroglu, E., et al.: On the performance of IEEE 802.11 under jamming. In: IEEE INFOCOM, Phoenix, AZ, USA, pp. 1265–1273, April 2008
6. Li, M., Koutsopoulos, I., Poovendran, R.: Optimal jamming attacks and network defense policies in wireless sensor networks. In: IEEE INFOCOM (2007)
7. Wood, A., Stankovic, J., Son, S.: JAM: a jammed-area mapping service for sensor networks. In: 24th IEEE Real-Time Systems Symposium (2003)
8. Punal, O., Pereira, C., Aguiar, A., Gross, J.: Experimental characterization and modeling of RF jamming attacks on vanets. IEEE Trans. Veh. Technol. **64**(2), 524540 (2015)
9. Benslimane, A., Nguyen-Minh, H.: Jamming attack model and detection method for beacons under multichannel operation in vehicular networks. IEEE Trans. Veh. Technol. **66**(7), 6475–6488 (2017)
10. Singh, A., Sharma, P.: A novel mechanism for detecting DOS attack in VANET using enhanced attacked packet detection algorithm (EAPDA). In: 2nd International Conference on Recent Advances in Engineering and Computational Sciences (RAECS), Chandigarh, pp. 1–5 (2015)
11. Sufyan, N., Saqib, N., Zia, M.: Detection of jamming attacks in 802.11b wireless networks. EURASIP J. Wirel. Commun. Netw. **1**, 118 (2013)
12. Roess, R., Prassas, E., McShane, W.: Traffic Engineering. Prentice-Hall, Englewood Cliffs (2004)
13. Mak, T., Laberteaux K., Sengupta, R.: A multi-channel VANET providing concurrent safety and commercial services. In: 2nd ACM International Workshop on Vehicular Ad Hoc Networks, pp. 1–9. ACM Press, New York (2005)
14. Bansal, R., Papantoni-Kazakos, P.: An algorithm for detecting a change in a stochastic process. IEEE Trans. Inf. Theory **32**, 227235 (1986)
15. Salem F., Elhillali, Y., Niar, S.: Efficient modeling of IEEE 802.11p MAC output process for V2X interworking enhancement. IET Netw. (2018). https://doi.org/10.1049/iet-net.2017.0228

Identifying Previously Requested Content by Side-Channel Timing Attack in NDN

Ertugrul Dogruluk(✉), Antonio Costa, and Joaquim Macedo

Centro Algoritmi, Universidade do Minho, Braga, Portugal
{d7474,costa,macedo}@di.uminho.pt

Abstract. NDN is a new name-based network paradigm. It is designed to keep the contents in the cache to increase the network efficiency. However, previously requested content may put the user privacy at risk. The time difference between cached and non-cached contents of interest responses can be used by an adversary to determine previously requested contents in cache. This attack is classified as side-channel timing attack. In NDN, it is used a signature to authenticate interests and data packets. However, signed packets does not affect the performance of side-channel timing attack. Independently of being signed or not, the adversary may identify both the sensitive and non-sensitive contents, recently cached by router. In order to mitigate side-channel attacks in NDN, there are several countermeasure methods proposed by other researchers. In this work, firstly we developed an attack scenario using ndnSIM simulator. Then we evaluated the scenario under attack and without attacks. We also proposed an adversary detection algorithm that combines three different defense countermeasures in order to maximize the cache availability.

Keywords: NDN · Content privacy · Side-channel timing attack

1 Introduction

Internet is being reshaped to handle content production and distribution, as nowadays users desire, for example, to watch movies and use social networking. Moreover, the number of IoT (Internet of Things) devices over the Internet is increasing enormously. However, such activities are not the most appropriate to be done over Internet, because this network was conceived for point-to-point communications. To overcome the problems raised by this communication paradigm, content centric networks (CCNs) have been proposed. According to this new paradigm, replicas of content(s) are generated and cached. The aim of caching is to reduce the latency and data loss, thereby improving the distribution quality of popular content(s). NDN (Named Data Network) is based on cache (buffer-memory) and name-based network that has been presented as a next version of CCN networks, as proposed by [16]. Nevertheless, caching, in spite of its benefits, may threat the privacy of NDN users. An adversary may know which content an user has requested and become motivated to proceed

R. Doss et al. (Eds.): FNSS 2018, CCIS 878, pp. 33–46, 2018.
https://doi.org/10.1007/978-3-319-94421-0_3

with a timing attack [7] It is based on the time differences between cached and not cached contents, as discussed later in this paper. Any type of cached content may be targeted by an adversary to cause a privacy threat. So, timing attack affecting the user privacy is a major problem in NDN, since blocking such attack is a challenging problem to solve. However, there are a few limited techniques to prevent timing attacks. The proposed countermeasure methods are based on artificial network delay [7,10], random caching [1], and request name encrypting [5]. Since these countermeasure techniques affect the cache performance, there is a trade-off between efficiency and privacy. For example, one technique is to delay the request content in the node to reduce the network throughput.

In this paper, an implemented attack topology is presented, side-channel timing attack measurements are analyzed and, based on primary findings, and is proposed an algorithm to identify the adversary node.

The rest of this paper is organized as follows. Section 2 summarizes the NDN architecture and its features. Section 3 demonstrates the operation of side-channel timing attack and how it can be used against the user privacy. Section 4 studies current countermeasures to mitigate side-channel timing attack. Section 5 shows the scenario implemented to simulate side-channel timing attacks. Section 6 summarizes attack scenario findings, and countermeasure results. Section 7 shows related researches that have been done so far. Finally, Sect. 8 presents the conclusions.

2 NDN Architecture

NDN is an ongoing project that proposes to transform the existing Internet design into a content-centric architecture, in order to improve the content distribution efficiency. NDN is based on human readable address names and keeps the contents in the cache, thus facilitating content distribution and providing low latency [17]. NDN is based on *interest* and *data* packets. Interest packets are produced by the consumers and data packets by the producers. The content name in the interest packet identifies the request of the consumer, for example, /pt/uminho/algoritmi. As shown in Fig. 1, the producer includes a signature in the data packet. Mechanisms for signing and verifying the integrity of the contents have been proposed for NDN, such as the one described in [8].

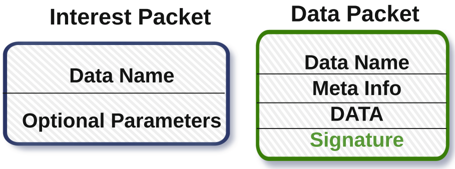


Fig. 1. Packet types used in NDN (adapted from [8])

2.1 NDN Forwarding Model

In a NDN router, the forwarding of interest and data packets is carried out by three engines: PIT(Pending Interest Table), FIB (Forwarding Information Base) and CS (Content Store) [17].

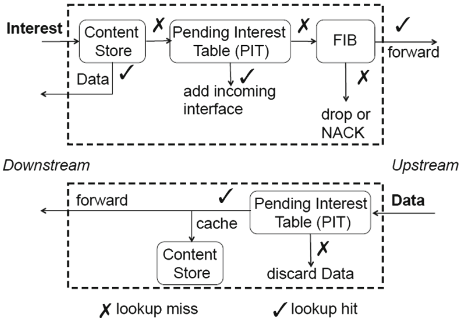


Fig. 2. NDN forwarding engine model [17].

The CS represents a cache for data packets. As illustrated in Fig. 2, the consumer interest first looks for data in the CS and, if there is a matching data, CS replies with a data packet. If the data packet is not in the CS, the router checks the data name in the PIT. If there is a matching name in the PIT, it records the incoming face (interface) of the interest. If not, the PIT forwards to the FIB the interest packet, which is then routed to the producer. For each interest packet that needs to be forwarded, the longest prefix (name) matching is searched in the FIB, which predicts when and where to forward the interest. The list of outgoing faces stored at the matched entry in the FIB is an important reference for routing.

When the data packet comes from the upstream router, the PIT is checked for a matching entry. If a match is found, the data packet is forwarded to the CS and then the entry is removed from the PIT for further incoming interest packets. If the authentication of the data packet fails, this packet is discarded by the PIT.

An NACK object informs consumers of data unavailability at the application level. Similar to NACK object, the Interest NACK is used to inform a router of its upstream router's inability at the network layer in NDN to forward an Interest packet, as described in [14].

In order to manage the packets in the CS, PIT, and FIB, the NDN developers created an engine called NFD (NDN Forwarding Daemon) [2]. This software tool is open source and keeps on evolving.

3 Side Channel Timing Attack

If a content is already cached in the CS, this replies to the user's interest with a data packet in a period of time. The RTT (Round Trip Time) is the time difference between sending the interest packet and receiving the data packet. The RTT of a cached content should be smaller than the RTT of a not-cached content. The adversary may take advantage of these RTT differences to know which contents have been or not placed in the CS. This technique is known as side-channel timing attack.

NDN struggles with four important privacy issues: *Naming*, *Content*, *Cache*, and *Signature*, as pointed out in [16]. This paper focus on the side-channel timing attack against the cached content in CS. Therefore, it mainly addresses privacy issues, but also name, content, and signature. These are an important issue to take into consideration, as it may affect the user privacy in NDN.

3.1 RTT Calculation

As shown in Fig. 3, a *timeout* occurs if the data packet is not received after a certain time interval. The timeout is related to the RTT estimation by a weight factor $\beta = 2$ (TimeOut $= \beta \times \text{RTT}_i$). Indeed, whenever an interest is forwarded to the upstream node, the router starts a timer, which will be used to measure the RTT. If the corresponding data packet arrives before the timer expiration, the router calculates the new RTT in accordance with Eq. 1, with $0 < \gamma < 1$. If γ is equal to $1 - \frac{1}{n}$ (n is the number of received data packets), then the real average RTT is obtained. If γ is close to 1, then the weighted average RTT is immune to delay changes for a short time interval. If γ is close to 0, then the weighted average RTT is very sensible to new delay changes.

If the data packet does not arrive, the router tries alternative routes until reaching the data packet. In case of non-existing data, a timeout is triggered and a NACK is sent. The router gives up of searching the data and the packet becomes an *unsatisfied interest*. But if a data packet was received, then the packet becomes a satisfied interest.

$$\boxed{<\text{RTT}>_n = \gamma * <\text{RTT}>_{n\text{-}1} + (1 - \gamma) * \text{RTT}_n} \qquad (1)$$

On the other side, if the downstream search has also exhausted for incoming interest packets, an interest NACK will be sent to identify, through an error code, the cause of the unsatisfied interest (*e.g., duplicate, congestion, no route*) [3]. Note that the interest NACKs cannot be used on a side-channel timing attack, because this attack requires the contents to be cached and retrieved.

3.2 Privacy in NDN

This sub-section explains how the side-channel timing attack is done and its effects on the CS. In this attack, an adversary node intends to break the cache

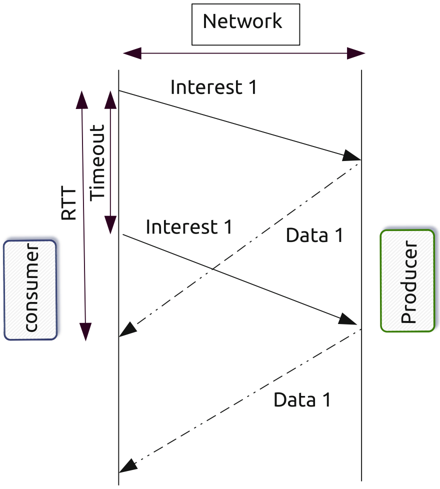


Fig. 3. Illustration of timeout and RTT in NDN

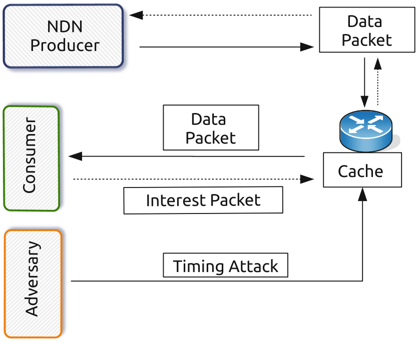


Fig. 4. Side-channel timing attack

privacy by requesting the same content from the CS that has been recently requested by a legitimate user.

Besides the cached contents, the adversary may also attack the certificate scheme. In NDN, each interest and data packets are bound with a public signature for integrity. Signatures are used for the authentication of the producer and each content is held in the CS for a time period. The public key of a certificate may also be used for a side-channel timing attack, that may jeopardize the user privacy, especially on real-time conversation applications and, trust-based communications. Next, it is presented the model based on RTTs, which may be used by an adversary to perform a side-channel timing attack in NDN.

Adversary Model. Considering the model illustrated in Fig. 4, let us suppose that the side-channel timing attack RTT measurement for a content is RTT_2 (retrieve content from NDN producer), RTT_1 is the RTT from the closest NDN router, RTT_e is the expected RTT of the intended content lookup, and ε is a negligible time difference. After collecting the timing samples, the adversary decides based on the following conditions [4]:

- If $|\text{RTT}_\text{e} - \text{RTT}_1| < \varepsilon$, the adversary node concludes that content has been cached by the closest router.
- If $|\text{RTT}_\text{e} - \text{RTT}_2| < \varepsilon$, the intended content is not held by any router, except the content producer.
- If $\text{RTT}_\text{e} > \text{RTT}_2$ and $\text{RTT}_\text{e} < \text{RTT}_1$, the adversary concludes that the lookup content has been fetched by away routers. Note that, the adversary can still predict the content location by relying on RTT_1 and RTT_2 values.
- $|\text{RTT}_2 - \text{RTT}_1| >> \varepsilon$.

4 Countermeasures

This section presents suitable countermeasure methods to mitigate side-timing actions on the cache. These methods manipulate the RTT of the content replied by the CS. The countermeasure methods are basically classified in three groups: no caching, artificial delay, and random caching. These methods are described next.

4.1 No Caching

In NDN, the CS can be configured for not caching. Since, there is no content held in the cache, the side-channel timing attack cannot be done. However, the cache is important for the NDN, as it is required for content distribution. So, directly giving up of the caching is not a good option in NDN, as described in [1].

4.2 Artificial Delay

Data delivery in the NDN is affected by a certain delay impose by the routers. An additional artificial delay can be used as a solution to prevent side channel timing attack, as explained next. Let us consider Δ the default delay value chosen randomly by a router. In side-channel timing attack, the adversary tries to figure out the Δ value. To complicate the adversary goal, a delay τ is added to Δ, in order to increase the router response delay. By measuring an higher RTT value, the adversary supposes that the content is not retrieved from the CS. Nevertheless, this is still a challenging problem, because the adversary may find the delay τ in a period of time and have success in the attack. The value τ can be changed by proposed algorithms, as described in [11]. Instead of using a constant τ, Schinzel [11] proposed a τ value based on a cryptographic hash function. Note that the delay methods imply a trade-off between privacy and latency, as a higher τ value affects negatively the latency on NDN.

4.3 Random Cache

A NDN router can cache the contents randomly, in order to mitigate side-channel timing attack. For instance, the CS may cache one data packet and then may not cache the next incoming data packet, based on a random probability number (k), as proposed in [1]. In such a random caching design, the side-channel timing measurement would fail for the contents not cached.

5 Implementation

In order to analyze the side-channel timing attack, a simulation scenario[1] was implemented on the simulator ndnSIM v2.4 [9]. We used a part of the ISP

[1] Code at: https://github.com/ertugd/ndnSIM-side-channel-timing-attack.git.

(Internet Service Provider) map dataset (SIGCOMM2002) by using rocketfuel mapper [12], which is used to convert and read large data sets for ndnSIM. As shown in Fig. 5, the physical topology is formed by sixteen consumers (called leaf), eight backbones (bb), eight central gateway routers (cgw) and one producer (gw-root). Two adversary nodes (leaf 6 and leaf 13) are located randomly into the topology. The producer payload size was 1024 bytes. The bandwidth of the links were 10 Mbps, 100 Mbps and 1000 Mbps, as indicated in Fig. 5. The minimum and maximum delays of the links are presented in Table 1 and were obtained from the data set of a real topology. NFD was configured for best-route strategy, in order to have the best network paths to the consumer nodes. The contents are created by the producer (gw-root) and the adversary nodes lookup for the intended contents on the gateway CS. Once a legitimate leaf node publishes an interest, the first lookup is done in the gateway router. However, if the data is not cached in the gateway router, then the lookup will be made in the backbone routers. The adversary nodes measure the RTT for the lookup contents that the legitimate nodes have requested. In our design, the attack is targeted to the gateway routers that manage interest packets by PIT and cache contents from the backbone routers.

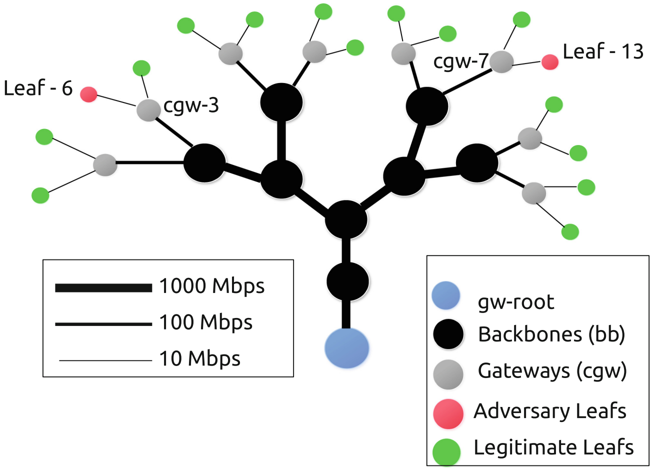


Fig. 5. Physical tree topology of the simulation scenario

Table 1. Delays (min/max) between nodes linked directly

Delays (ms)	bb	cgw	leafs	gw-root
bb	2.51/7.56	3.11/9.10	-	4.77
cgw	3.11/9.10	-	0.15/9.67	-

The adversary nodes sent interest packets at a rate of 100 packets/s with a malicious interest prefixes, and the legitimate nodes at a rate 100 packets/s sending unique (not requesting same content name again) interest prefixes. The legitimate leaf nodes were configured to generate interest traffic with a randomized uniform (7 leaves) pattern and an exponential pattern (7 leaves). The legitimate nodes sent interest packets (size = ~40 kB) with a randomized uniform pattern (0, 1/frequency), and exponential distribution (mean of $1/frequency$), as shown in Table 2. Every backbone and central gateway router caches the contents (data packet size = ~110kB), and the popular LRU (Least Recent Used), with a capacity of 1000 data packets per node, was chosen for the CS policy.

Table 2. Interest configurations

	Frequency (s^{-1})	Traffic pattern	Prefix
Legitimate leaves	100	Uniform and exponential	/google.com/sub_prefix
Leaf-6	100	Uniform	/google.com/%FE%01
Leaf-13	100	Uniform	/google.com/%FE%07%96

6 Results

The primary results showed that the adversary may retrieve the cached contents that were recently requested by the consumer. The NFD is located on each router to analyze metrics for RTT, and cache hits or misses per node.

6.1 Cache Analysis

If a content has been requested and cached, the next request for the same content causes a cache hit. The cache hit ratio (Eq. 2) is used to indicate a side-channel timing attack, it is calculated by cache hit and misses by a predefined time interval Δ. The adversarial leaf 6 and leaf 13 sent a lookup interest packet to cgw-3, 7. The lookup is done for cached contents, which are the contents recently requested by neighbor nodes (leaf 5, leaf 14).

The legitimate leaves sent interest packets with unique prefixes. Therefore, the cache hit ratio is not possible to be calculated for the leaves, because these are not re-consuming the content from the cgw CS. As Fig. 6 illustrates, the adversary leaves hit the cache during an attack period, so their cache hit ratios may reach 100%.

$$\boxed{\text{Cache hit ratio} = \frac{\sum_{\mathrm{i=0}}^{\mathrm{n}} \text{cache\_hit}_i}{\sum_{\mathrm{i=0}}^{\mathrm{n}} \text{cache\_hit}_i + \sum_{\mathrm{j=0}}^{\mathrm{m}} \text{cache\_miss}_j} * 100\%} \quad (2)$$

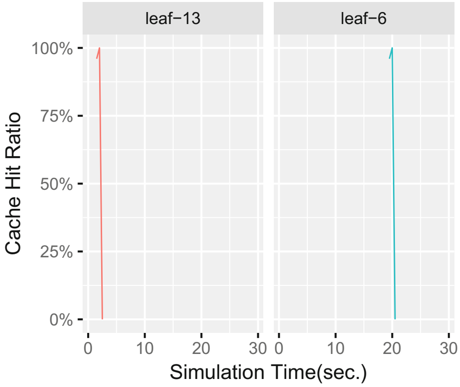


Fig. 6. Effect of a timing attack on the cache hit ratio

6.2 Countermeasure Algorithm

The countermeasure methods affects surely the CS performance. In order to improve the CS performance, we proposed a new countermeasure algorithm (Algorithm 1). The algorithm proposes a method to identify an adversary node and to apply the countermeasure mechanism. It requires that NFD is running in each router. The proposed algorithm examines the RTT and the cache hit ratio metrics to identify the adversary node. The first detection level is done by the RTT threshold. This threshold is calculated and updated for every consumer using Eq. 3, where n is the number of RTT samples obtained periodically during a predefined time interval (*e.g.* $\Delta = 0.5\,\mathrm{s}$). The detection decision is based on the RTT threshold that is expected from consumers under no attack condition. In case of a consumer RTT value being much lower than the RTT threshold, the router is set to random caching for that consumer.

The algorithm also provides a second level detection method by using the cache hit ratio, in order to identify adversary node. The cache hit ratio is calculated periodically during the same time intervals used to calculate the RTT threshold. The NFD considers the node as an adversary, if its cache hit ratio is above the cache hit ratio threshold (Eq. 4). If the cache hit ratio of a node is above the cache hit ratio threshold, the NFD sets the node for random caching. If the RTT of a node is above the RTT threshold and the cache hit ratio is considerably lower than the cache hit ratio threshold, then the NFD configures the router with the LRU policy.

$$\boxed{\text{RTT threshold} = \frac{\sum_{\mathrm{i=0}}^{\mathrm{n}} \mathrm{RTT}_i}{\mathrm{n}}} \tag{3}$$

$$\boxed{\text{Cache hit ratio threshold} = \frac{\sum_{\mathrm{i=0}}^{\mathrm{m}} \mathrm{CacheHitRatio}_i}{\mathrm{m}}} \tag{4}$$

```
while(true) {
   newRTTavailable = checkNewRTTavailable()
   newCacheHitRatioAvailable = checkNewCacheHitRatio()

   if(newRTTavailable) {
      RTT = getRTTfromNFD()
      RTTthreshold = calculateRTTthreshold(RTT)                    //Eq.3
      checkAttack(RTT, RTTthreshold, oldRTTthreshold, cacheHitRatio,
     cacheHitThreshold, oldCacheHitThreshold)
    }

   if(newCacheHitRatioAvailable) {
      cacheHitRatio = getCacheHitRatiofromNFD()                    //Eq.2
      cacheHitThreshold = calculateCacheHitThreshold(cacheHitRatio)//Eq.4
      checkAttack(RTT, RTTthreshold, oldRTTthreshold, cacheHitRatio,
     cacheHitThreshold, oldCacheHitThreshold)
      oldCacheHitThreshold = cacheHitThreshold
    }

   if(newRTTavailable) oldRTTthreshold = RTTthreshold
}



function checkAttack(RTT, RTTthreshold, oldRTTthreshold, cacheHitRatio,
     cacheHitThreshold, oldCacheHitThreshold) {
       state = NO_ATTACK

   if (RTT < RTTthreshold OR cacheHitRatio > cacheHitThreshold) {
       state = ATTACK_DETECTED
       router_random_cache() //apply CS random caching
     }

   else {
       state = NO_ATTACK
       router_LRU_cache()    //apply CS LRU caching
     }

   if (RTTthreshold < oldRTTthreshold OR cacheHitRatio >
    oldCacheHitThreshold) {
       state = ATTACK_DETECTED
       router_random_cache() //apply CS random caching
     }
}
```

Algorithm 1. Adversary detection

6.3 Timing Measurements

In order to analyze the values of both the RTT samples and the RTT threshold, we ran the scenario considering both the attack and no attack. As shown in Table 3, the RTT threshold values were used to identify the attack of the adversarial node. The attack is detected when the RTT of the sample is below the

RTT threshold. Then, we ran the scenario with under attack and collected the first RTT samples. The results between threshold and first samples are showed that RTT values may change, because of real throughput delays and congestions. However, both the expected RTT under no attack and the RTT variation (regarding the RTT threshold) reduced dramatically for leaf 6 and leaf 13. Therefore, these leaves are considered adversarial nodes. The experimental results showed that the RTT of the adversarial leaves are shorter than the RTT of the legitimate leaves. When this occurs, the NFD may engage the random caching against the attack.

Table 3. RTT analysis

Leaf	Estimated RTT threshold(s) no attack	First RTT sample(s)	RTT variation (%)
1	0.03122045	0.03099957	−0.71%
2	0.03124	0.03100993	−0.74%
3	0.03123136	0.0310034	−0.73%
4	0.03122998	0.03100182	−0.73%
5	0.03122413	0.03099494	−0.73%
6	0.03122487	**0.0194212**	**−37.80%**
7	0.03122427	0.03099683	−0.73%
8	0.03122295	0.030993578	−0.73%
9	0.03123878	0.03101116	−0.73%
10	0.03123794	0.03100934	−0.73%
11	0.03123692	0.03100801	−0.73%
12	0.03123955	0.03101118	−0.73%
13	0.03123717	**0.0144831**	**−53.64%**
14	0.03123604	0.03100802	−0.73%
15	0.03123482	0.03100664	−0.73%
16	0.03123906	0.03101061	−0.73%

7 Related Work

The related works considered the side-channel attacks on NDN.

Dogruluk *et al.* [6] evaluated the privacy attack in NDN, and proposed an adversary detection method controlled by the values of cache and interest hit ratios.

Felten and Schneider [7] evaluated side-channel timing attack measurements on web privacy. They show that the malicious web cookies can be used to acknowledge user's recent visited web sites. They also concluded that web anonymization tools and turning off the browser's JavaScripts features do not prevent side-channel timing attack measurements.

Zhang *et al.* [15] investigated the use of side-channel timing attack on VoIP privacy and how adversary may reveal the call history attacking with legitimate Public Key Infrastructure (PKI).

DiBenetetto and Gast [5] developed (AND$\bar{a}$NA) a tool to mitigate timing attacks in NDN. With this tool, the requested names are encrypted and afterwards are verified by the nodes and delivered to the user as data. AND$\bar{a}$NA is based on a TOR (The Onion Routing) paradigm [13], an anonymity browsing tool for Internet. AND$\bar{a}$NA makes the CS unpractical, because the data packet is only consumable for who requests it.

Mohaisen *et al.* [10] focused on cache privacy on ICNs (Information-centric networks) and proposed an user driven countermeasure method called Vanilla. For privacy-sensitive contents, an edge router caches the content from the producer and keeps the retrieval times of the first interest and delay the next coming requests. However, the per-client solution will not be feasible, because of the large number of consumers. Acs *et al.* [1] addressed side-channel timing attacks in NDN and proposed random caching and delay. This user driven method is based on flagging contents by */private* and */non-private*. Not with standing, we state that all contents have a sense of privacy, even daily visited sites such as google.com. To the best of our knowledge, per-client driven solution may not be the most appropriate approach for CS propose. For instance, the */private* content of a user could be stated as a */non-private* for other user(s).

8 Conclusions

In order to point out side-channel timing attack in NDN, we developed a scenario and show primary findings. The results show that the adversary may distinguish contents between cached and non-cached, by comparing content RTT differences. An adversary detection method algorithm is presented. The NFD based algorithm, distinguishes adversary and legitimate node by checking RTT and cache hit ratio threshold values.

The side-channel timing attack is easy to be implemented by an adversary, but the detection is a challenging issue to solve. Ideally, the cache hit and RTT metrics can be also used as an indications for side-channel timing attack. In order to keep CS performance, these indication methods can be used to identify adversary node controlled by NFD application. Through this identification method, the countermeasures (delay, random caching, and no cache) can be applied against the adversary nodes, in order to improve the CS performance.

Acknowledgment. This work has been supported by COMPETE: POCI-01- 0145-FEDER-007043 and FCT-Fundacao ao para a Ciencia e Tecnologia within the Project Scope: UID/CEC/00319/2013.

References

1. Acs, G., Conti, M., Gasti, P., Ghali, C., Tsudik, G.: Cache privacy in named-data networking. In: IEEE 33rd International Conference on Distributed Computing Systems, pp. 41–51. IEEE (2013)
2. Afanasyev, A., Shi, J., Zhang, B., Zhang, L., Moiseenko, I., Yu, Y., Shang, W., Huang, Y., Abraham, J.P., Dibenedetto, S., Fan, C., Pesavento, D., Grassi, G., Pau, G., Zhang, H., Song, T., Abraham, H.B., Crowley, P., Amin, S.O., Lehman, V., Wang, L.: NFD developer's guide. NDN. Technical report. NDN-0021 4, pp. 1–56 (2015)
3. April, M., Report, A., Jacobson, V., Burke, J., Zhang, L., Claffy, K., Papadopoulos, C., Wang, L., Halderman, J.A., Crowley, P.: Named Data Networking Next Phase (NDN-NP) Project May 2014–April 2015 Annual Report (2015)
4. Chaabane, A., Cristofaro, E.D.: Privacy in content-oriented networking: threats and countermeasures. ACM SIGCOMM Comput. Commun. Rev. **43**(3), 26–33 (2013)
5. DiBenedetto, S., Gasti, P.: ANDaNA: anonymous named data networking application. In: Proceedings of the Network and Distributed System Security Symposium, pp. 1–20 (2012)
6. Dogruluk, E., Costa, A., Macedo, J.: Evaluating privacy attacks in named data network. In: Proceedings of the IEEE Symposium on Computers and Communication, vol. 2016, August 2016
7. Felten, E.W., Schneider, M.A.: Timing attacks on web privacy. In: Proceedings of the 7th ACM Conference on Computer and Communications Security, CCS 2000, pp. 25–32 (2000)
8. Jacobson, V., Smetters, D.K., Thornton, J.D., Plass, M., Briggs, N., Braynard, R.: Networking named content. Commun. ACM **55**(1), 117 (2012)
9. Mastorakis, S., Afanasyev, A., Moiseenko, I., Zhang, L.: ndnSIM 2.0: a new version of the NDN simulator for NS-3, pp. 1–8 (2015)
10. Mohaisen, A., Mekky, H., Zhang, X., Xie, H., Kim, Y.: Timing attacks on access privacy in information centric networks and countermeasures. IEEE Trans. Dependable Secur. Comput. **12**(6), 675–687 (2015)
11. Schinzel, S.: An efficient mitigation method for timing side channels on the web. In: 2nd International Workshop on Constructive Side-Channel Analysis and Secure Design, pp. 1–6 (2011)
12. Spring, N., Wetherall, D.: Measuring ISP Topologies with Rocketfuel. In: Proceedings of the 2002 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, SIGCOMM 2002, pp. 133–145 (2002)
13. Wiangsripanawan, R., Susilo, W., Safavi-Naini, R.: Design principles for low latency anonymous network systems secure against timing attacks. In: Conferences in Research and Practice in Information Technology Series, vol. 68, pp. 183–191 (2007)
14. Yi, C., Afanasyev, A., Wang, L., Zhang, B., Zhang, L.: Adaptive forwarding in named data networking. ACM SIGCOMM Comput. Commun. Rev. **42**(3), 62 (2012)
15. Zhang, G., Fischer-Huebner, S., Martucci, L.a., Ehlert, S.: Revealing the calling history of SIP VoIP systems by timing attacks. In: 2009 International Conference on Availability, Reliability and Security, pp. 135–142 (2009)

16. Zhang, L., Estrin, D., Burke, J., Jacobson, V., Thornton, J.D., Smetters, D.K., Zhang, B., Tsudik, G., Massey, D., Papadopoulos, C., Wang, L., Crowley, P., Yeh, E.: Named data networking (NDN) project. NDN, Technical report NDN-0001, pp. 1–26, October 2010
17. Zhang, L., Jacobson, V., Diego, S., Crowley, P., Louis, S., Wang, L.: Named data networking. ACM SIGCOMM Comput. Commun. Rev. **44**(3), 66–73 (2014)

Security, Privacy and Ethics

A Framework to Explore Ethical Issues When Using Big Data Analytics on the Future Networked Internet of Things

Jeffrey S. Saltz(✉)

Syracuse University, 233 Hinds Hall, Syracuse, NY, USA
jsaltz@syr.edu

Abstract. The networked future will generate a huge amount of data. With this in mind, using big data analytics will be an important capability that will be required to fully leverage the knowledge within the data. However, collecting, storing and analyzing the data can create many ethical situations that data scientists have yet to ponder. Hence, this paper explores some of the possible ethical conundrums that might have to be addressed within a big data network of the future project and proposes a framework that can be used by data scientists working within such a context. These ethical challenges are explored within an example of future networked vehicles. In short, the framework focuses on two high level ethical considerations that need to be considered: data related challenges and model related challenges.

Keywords: Internet of things · Big data · Ethics · Network of the future

1 Introduction

Big data is the study of large amounts of data and the process of turning that data into actionable insight (Saltz and Shamshurin 2016). Big data is often described across four dimensions, noted as the 4 "Vs", which are volume (scale of data), veracity (uncertainty of data), velocity (streaming data) and variety (different forms of data). The network of the future, which will be a network of billions or even trillions of devices, including small components and larger machines, either in a standalone use case (such as a smartphone) or embedded within another device (such as within a vehicle). One way to think about the network is that the main goal of the interaction and cooperation between these networked things and objects is to be able to view the different components as a combined entity (Stergiou and Psannis 2017).

In exploring this network, it is clear that leveraging the information within this network will require big data analytics. In other words, this future network, which will truly be an internet of things, will generate huge amounts of data that needs to be explored. In using the 4 Vs, one can see that the volume of data will be significant, due to the number of devices that will be in existence and the velocity of the data will also be substantial due to the frequency of data being updated (often in real-time).

Hence, the data generated by the network of the future will generate data at a rate not yet seen within current data analytical situations, and in fact, will likely push the

R. Doss et al. (Eds.): FNSS 2018, CCIS 878, pp. 49–60, 2018.
https://doi.org/10.1007/978-3-319-94421-0_4

envelope on current big data analytical capabilities. However, leveraging big data techniques on the network of the future will also unlock huge value. In any event, since big data will be a key aspect of leveraging the network of the future, it makes sense to explore some of the potential challenges one might encounter when creating such a big data application. While there are many possible issues to explore, the focus in this paper is on the potential ethical issues when using big data within this rich internet of things data environment.

Specifically, this research aims to take a step forward in the creation of a framework to help identify ethical issues and challenges for big data projects within an integrated network of the future context by exploring the following research question:

> *What might be a useful framework to explore ethical challenges when establishing or executing a big data project within the network of the future context?*

The rest of this paper first provides some background on big data within a IoT/Network of the future context as well as how as some ethical challenges in big data, including highlighting applicable codes of ethics. Section 3 then provides an analysis of the ethical considerations for IoT big data analytics, focusing on data and model related challenges. The next Sect. 4, then explores a possible framework for teams to use to explore potential ethical conundrums when doing IoT Big Data analysis. Finally, Sect. 5 provides a conclusion to the research.

2 Background

In creating a framework to explore this type of big data application, this paper first explores what has been previously written with respect to big data and the network of the future. It then explores the ethical considerations that must be addressed within a project as well as the possibly relevant codes of ethics.

2.1 Big Data Within the Network of the Future

The Internet of Things and the Internet of Everything has been noted as needing big data for several years, at least as far back as 2013 (O'Leary 2013). In fact, it has been noted that the real problem is not that one can acquire large amounts of data, but whether it has any value or not (Stergiou and Psannis 2017).

Some have already started to explore big data within a network of the future context. For example, some have analyzed an integrated big data analytical framework for IoT and Smart City applications (Strohbach et al. 2015), and note that the main challenge is how to address the volume and velocity of the data generated. Others have explored this future environment within the context of wearable medical devices that continually generate a huge amount of data and, combined with big data, enable the creation of numerous applications for individualized eHealth (Firouzi et al. 2018). For example, Manogaran et al. (2017), discusses an Internet of Things (IoT) architecture that consists of a scalable platform and set of algorithms to process the huge amount of sensor data generated by the wearable medical devices and identify useful patterns, based on a scalable MapReduce-based application within an Apache Mahout and

Hadoop Distributed File System. Yet others have focused on leveraging the future network of connected components and big data for industrial informatics, which aims to enable knowledge-based factory automation, enhanced products, improved manufacturing processes and manufacturing systems (Bi 2017; Li et al. 2017). Finally, a pilot of this advanced network was created as a use case to support tourism, in an integrated applicaiton of IoT and big data analytics for smart tourism and sustainable cultural heritage in the city of Trento, Italy (Firouzi et al. 2018).

However, while big data is starting to be used in this future networked context, there has been minimal research with respect to ethics in this context of big data and the growing networked world in which we will live. The one relevant identified paper notes that today's big data IoT environment is introducing changes for which laws and rules of acceptable conduct have not yet been developed (Guan and Zhou 2017). That paper focuses on the E.U. and U.S. laws for the emerging social problem of analytical information brokerage.

2.2 Ethics and Big Data

At a high level, ethics is a moral framework by which one can determine right from wrong in human decision making. The objectives of ethical decision making is typically described in terms of justice, fairness and the avoidance of harm. For example, Bynum (2008) describes ethics as concerned with "protecting and advancing central human values, such as life, health, security, happiness, freedom, knowledge, resources, power and opportunity." Ethics also leads one to consider harm to society as a whole, for example consideration of whether an action harms a minority population. In fact, Dwork et al. (2012) suggest that treating individuals fairly can be insufficient, and that groups of people with similar characteristics must be treated fairly. Hence, ethics includes consideration of discrimination and bias, which is whether groups of humans are being treated differently in ways and that some groups are collectively harmed or disadvantaged compared to others. Put another way, ethics provides a moral framework to avoid harm to physical and psychological health, dignity, rights, reputation and financial wellbeing of a person or group of people.

As one can see from the brief description of ethics, there are certainly situations where ethics is needed within a big data science context. Some simple examples include the use of a human characteristic such as race or gender within machine learning algorithms or the act of disclosing personally identifiable information without the consent of that person.

It would be easy to suggest that where ethical standards are needed for data science, laws and regulations have been established and should be followed. However, Zwitter (2014) points out that ethics and regulation tend to lag technology improvements, and this certainly could be the case with data science. One might also think that an existing code of conduct, such as the ACM code of conduct for computing professionals or the Data Science Association's code of conduct, would be sufficient. While these might be sufficient, without more deeply exploring the types of ethics challenges a data scientist might encounter, one can not determine if an existing code fully covers the field of data science. Equally important, while there has been research published on a specific ethic challenge in data science, for example linking data sets to identify people from multiple anonymous data sets (Stevenson 2014), there has been minimal work exploring and documenting the breadth of ethical situations that a data scientist might encounter (Saltz and Shamshurin 2016). In fact, it was not even mentioned during the exploration of the socio-technical challenges in doing big data projects (Saltz et al. 2017). Perhaps the most comprehensive work to date was focused on the ethics of data science in journalism (Fairfield & Shtein, 2014), but journalism is clearly a small subset of where one might use data science.

With the newness of the field and the lack of an accepted list of ethical challenges, it is not surprising that many have argued for the development of applied ethics in big data science projects (Floridi and Taddeo 2016; Schwartz 2011; Metcalf et al. 2016).

2.3 Codes of Ethics

Numerous codes of ethics exist for different areas relating to data science. These codes can be leveraged to understand some of the potential ethical challenges. Some of these codes focus on general professional obligations such as honesty, competence, confidentiality, conflicts of interest, and professional conduct and most also mandate privacy protections. Others, such as the Data Science Association's Code of Professional Conduct go a little further into specific topics such as data quality and statistical techniques. Below, the relevant codes and frameworks briefly noted, and the appendix provides links to each of these codes.

Data Science Association's Code of Professional Conduct provides a comprehensive section on quality of data and evidence, with practical guidance for conducting data science projects. For example: "The data scientist shall not … misuse data science results to communicate a false reality or promote an illusion of understanding". However, the code lacks a wider consideration of how data science can cause harm, how models can contain biases, how data scientists make objective decisions about the structure of the model, and how to balance the risk of harm with an appropriate level of ethical oversight.

Association of Computer Machinery Code was developed for computer science practitioners and is the basis of computer science ethics education. Beyond general rules of conduct, it is valuable to the data scientist in that (1) many data scientists write computer code, and hence, covers the software development aspect of data science and (2) the code has stipulations about gathering requirements and validating that those requirements have been met, which is useful in both a software engineering and data science context.

Digital Analytics Association Code tries to ensure consumers are treated with respect. It has a focus on data, such as privacy and transparency of data usage.

UK Government Data Science Ethical Framework provides a list of six admonitions, supplemented for each point with tips and checklists. While not complete, the admonitions are useful, such as 'keep data secure', 'create robust data models' and 'be open to public perceptions'.

Jagadish's 2 Rule Code proposes a simple two-rule code. The first is that there should be no surprises to those described in the data and the second rule is to own your models.

Financial Modelers' Manifesto was developed for financial analysts soon after the 2008 financial crisis. The points included can be thought of as topics of reflection, such as "I will remember that I didn't make the world, and it doesn't satisfy my equations", "Though I will use models boldly to estimate value, I will not be overly impressed by mathematics" and "I will never sacrifice reality for elegance without explaining why I have done so".

Accenture's 12 Principles provides twelve universal ethical principles for data scientists covering a broad set of relevant topics were discussed. These principles cover highly valuable topics such as "the highest priority is to respect the persons behind the data" and "seek to match privacy and security safeguards with privacy and security expectations". It also proposes a model for the application of ethics throughout the "data supply chain" of Acquire, Store, Aggregate, Analyze Use, Share/Sell and Dispose.

Schwarz's framework is perhaps the most practical example of a broad ethical framework for data science, which begins with a set of ethical considerations that are general, for example "Companies should implement appropriate safeguards to protect the security of information". It then continues with other points that are specific to the stages in the analytical process: data collection, integration and analysis, decision making, and review/revision of models, and include, for example, considerations whether data is of sufficient quality for its likely future use.

3 Ethical Considerations for IoT Big Data Analytics

As summarized in Table 1, the key ethical challenges are described using two themes (data related challenges and model related challenges). The rest of this section explores these challenges within the context of the future connected network automobile of the future.

Table 1. Key ethical considerations within big data projects.

Theme	Topic within theme
Challenges when using data	Privacy and anonymity
	Data misuse
	Data accuracy and validity
Challenges when using analytical models	Personal and group harm
	Subjective model design
	Model misuse/misinterpretation

3.1 Data Related Challenges

This theme focuses on the key ethical challenges that can arise relating to the collection and use of data. The network of the future will generate significant streaming data, which can be analyzed in real-time or stored and analyzed later, for example, to do predictive analytics. Note that data scientists often integrate multiple distinct data sources to generate novel insights. So, one might use a yelp dataset, combined with automobile location data, to help predict where a person might want to stop and eat. However, the creation, collection and use of data create many potentially challenging ethical situations.

Privacy and Anonymity focuses on an individual's right to choose which of their activities and facts are shared with others. In a digital age this includes both what the individual *chooses* to publish and their ability to *control* with whom the data is shared. In the connected car, as an example of an ethical situation, one needs to explore which apps have the right to collect and store that location data. Which apps also have the right to then share that data with others? This sharing might lower the cost of the system or app, and the person driving the car likely agreed to terms of service, which might have included allowing the collection and sharing of data. But Tene and Polotensky (2012) suggest that consumers fail to read and understand these policies, which raises many questions with respect to actual consent.

The impact of aggregating and linking data, and the ability for harm to arise from that information, has been noted as differentiators from other fields (Stevenson 2014; Fairfield and Shtein 2014). For example, Metcalf et al. (2016) point out that the phenomenon of big data has introduced "a change in the relationality, flexibility, repurposing and de-contextualization of data" requiring development of new ethical considerations. One of Accenture's rules directly relates to this concept: "Data subjects hold a range of expectations about the privacy and security of their data and those expectations are often context-dependent. Designers and data professionals should give due consideration to those expectations and align safeguards and expectations as much as possible". In addition, Schwartz points out that privacy can be controlled or breached both at the point where data is created as well as at the point where it is shared or used.

Data Misuse is something that the data scientist must actively guard against. Being able to access or collect data does not mean that it is ethical to use that data (Boyd et al. 2014). For example, one might give permission for an app to share data between applications across cars to avoid collisions. But access to the data for collision detection is different than access to the data for advertising. The data science association's code notes this clearly with statements such as "if a data scientist reasonably believes a client is misusing data science … the data scientist shall take reasonable remedial measures … including, if necessary, disclosure to the proper authorities" and that the "the data scientist shall take reasonable measures to persuade the client to use data science appropriately". This is complicated by that some uses of data might not be anticipated when data collection was initiated. For example, location and other sensor data within a car might be used to help generate predictive analytics with respect to schedule, and unscheduled, maintenance. Which organizations might have access to the data that could help with such predictive analytics might not have been clearly defined.

Data Accuracy and Validity is something that the data scientist must consider. This includes not only the accuracy of the data, but also whether the data being used is appropriate for the problem being addressed. In other words, the data scientist needs to ensure the 'fitness of purpose' with respect to how the data is used. Otherwise, data can be taken out of context or might not be used in the spirit of how the data provider intended. For example, data preparation often requires imputing missing values or excluding records with missing values, which could generate inaccurate results or systematically disadvantage a group of people whose circumstances cause them to routinely not have certain data attributes recorded (Boyd et al. 2014). For example, in the networked car example, when doing collision detection, one needs to understand the accuracy of the GPS data, in conjunction with other data such as image processing and sensors to identify potential obstacles. Understanding data accuracy is a key aspect of the data scientist's role and is noted in the data science association's code that states that data scientists should explicitly rate the data quality used.

3.2 Model Related Challenges

This theme focuses on the ethical challenges that can arise from the building and using of analytical models. An analytical model is a mathematical technique used for simulating, explaining, and making predictions about future situations based on past data. In other words, an analytical model is a set of mathematical functions that encapsulate the prediction of a certain situation based on past information. However, the use of an algorithm (analytical model) might introduce or amplify a range of ethical situations.

Personal and Group Harm is important because there can be significant impact due to how models are used. One concern is that data science models are frequently built using data that records a bias, and thus the models can be employed such that they systematically disadvantage societal sub-groups (Crawford 2013). For example, with respect to the networked car, one might have sensors to identify potholes in roads. However, it is possible that the sensors in newer cars might be not as common in poorer neighborhoods. If this was the case, a city might incorrectly assume that those roads, in poorer neighborhoods, do not have as many potholes. Hence, there needs to be a focus on avoiding discrimination and bias, which might unknowingly occur via the use of a big data science model. The fact that models can perpetuate and amplify bias leads to an equally important need to consider whether the bias in a model might lead to any group of people from being disadvantaged.

Subjective Model Design is another concern in that while data science can bring objectivity to decision making, there is subjectivity within data science modeling, in that decisions must be made about which algorithm to use, which data sources to use, whether one data point should be used as a proxy for a missing fact, and how to interpret results (Sandvig et al. 2014). For example, a data scientist might assume that predictive analytics for when to replace car tires is dependent on the number of miles driven and the style of driving (sudden stops), and ignore the type of tire used. This might cause the model to be in accurate and create an ethical situation since cheaper tires might not wear as well and cause the model to give less accurate predictions for a specific portion of the population. Furthermore, biases contained in the data being used are often preserved or amplified in the results of the model (Boyd et al. 2014). While this concept is the focus of Jagadish's simple rule of "own your models", this concept is largely absent from existing codes.

Explain Model Limitations is an important responsibility for data scientists. Most predictive models are statistical in nature. They provide no guarantees; rather, they tell us about areas where increased probability of an outcome might guide us to act differently. The required accuracy of a model depends on the use of a model. For example, collision detection has a different tolerance for inaccuracy as compared to a marketing campaign based on a vehicle location. The data scientist's ethical responsibilities do not end with the completion of a model. The data scientist has a duty to explain their models and the implications. In particular, the model must be explained using language that non data scientists, such as managers, can understand. In addition, as noted by Schwartz, a deployed model must be periodically re-evaluated for soundness, and must also have appropriate oversight and governance. In a similar fashion, the data science association's code of conduct notes that data scientist should explain the data quality rating to their client. Furthermore, Schwartz notes the need to periodically re-evaluate models for soundness, and must have appropriate oversight and governance.

4 Discussion

4.1 Proposing an Ethics Framework

One way to explore how teams could use these themes is to integrate the identified ethical challenges within a data science process. Current descriptions on how to execute big data projects generally adopt a task-focused approach, conveying the techniques required to analyze data. While these process models differ in details, at a high level, they are broadly similar. For example, Jagadish et al. (2014) describe a process that includes acquisition, information extraction and cleaning, data integration, modeling, analysis, interpretation and deployment. This step-by-step view is similar to CRISP-DM (*Cross Industry Standard Process for Data Mining),* which was established in the 1990s, and is a data mining process model for data mining experts (Shearer 2000).

Since CRISP-DM is the most widely used process (Haffar 2015), one can use that process model as a way to integrate the identified ethical challenges with the phases of the big data science project life cycle. As shown Fig. 1, CRISP-DM mentions six high-level phases: business understanding, data understanding, data preparation, modeling, evaluation, and deployment. CRISP-DM also provides some high level iteration between the steps (Chapman et al. 2000). Typically, when using this framework, the team progresses through the different phases as they deem appropriate. As needed, the team can "loop back" to a previous phase (ex. more data preparation), and in general, can define milestones they think are useful.

Table 2 shows the mapping of the identified themes to the project phases. Specifically, not surprisingly, the data related challenges map to the data understanding and data preparation phases, and the model related challenges map to the modeling, evaluation and deployment phases. However, there was no clear ethical theme related to the business understanding phase. It makes sense that this theme was not a key area of focus in the literature, since this phase is more focused on topics such as ensuring accountability, which while important, might not be a key focus of a paper exploring new ethical issues relating to data science. For this business understanding phase, two ethical new considerations are proposed. First, at the start of the project, the team should consider, at a conceptual level, the potential personal and group harm. In addition, the team should also explore team accountability of potential ethical situations.

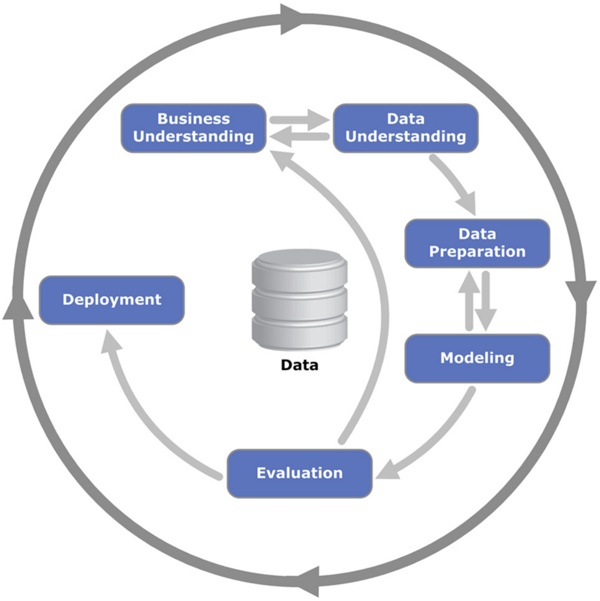


Fig. 1. Flow of a CRISP-DM Project (Wikipedia, 2017)

Table 2. Key ethical considerations by phase of project.

Project phase	Key ethical themes	Ethical considerations
Business understanding	Project initiation/management challenges	Personal and group harm
		Team accountability
Data understanding	Data challenges	Data misuse
Data preparation		Data privacy & anonymity
		Data accuracy
Modeling	Model challenges	Personal and group harm
Evaluation		Subjective model design
Deployment		Misuse/misinterpretation

4.2 Analyzing the Framework

Guan and Zhou (2017) suggest that these ethical issues can be categorized in five moral dimensions: information rights and obligations, property rights and obligations, system quality, quality of life, and accountability and control. With this in mind, one approach in reviewing the framework is to compare the framework to the key dimensions noted by Guan and Zhou.

As shown in Table 3, their accountability and control is fairly analogous to the team accountability considerations noted in this paper. In addition, information rights and obligations maps to data misuse and property rights and obligations maps to data privacy & anonymity. Quality of life can map to personal and group harm as well as subjective model design. Finally, system quality can be mapped to data accuracy as well as Model Misuse/Misinterpretation. Hence, one could view this framework as an extension of the work proposed by Guan and Zhou, in that it refines some of the ethical considerations as well as maps those considerations to the phase of the project.

Table 3. Mapping Guan and Zhou concepts to this framework

Guan and Zhou	Ethical considerations noted
Accountability and control	Team accountability
Information rights and obligations	Data misuse
Property rights and obligations	Data privacy & anonymity
Quality of life	Personal and group harm
Quality of life	Subjective model design
System quality	Data accuracy
System quality	Model misuse/misinterpretation

5 Conclusion

In this paper, some of the potential ethical conundrums that data scientists might encounter when working on big data network of the future effort are explored. Exploring these ethical situations can help the field fully exploit the potential benefit of big data without doing it in a way that harms a subset of the population. Future work can test this framework, either by doing an actual big data project using this framework or by going through the analysis of doing a big data project and identifying how this framework can be used. Future work could also explore how to integrate these concepts into an applied introduction to data science course (Saltz and Heckman 2016).

References

Bi, Z.: Embracing internet of things (IoT) and big data for industrial informatics. Enterp. Inf. Syst. **11**(7), 949–951 (2017)

Boyd, D., Levy, K., Marwick, A.E.: The Networked Nature of Algorithmic Discrimination. Data and Discrimination. Collected Essays, New America (2014)

Bynum, T.: Computer and Information Ethics. The Stanford Encylopedia of Philosophy, Online edn. Metaphysics Research Lab, Stanford University (2008)

Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C., Rudiger, W.: CRISP-DM 1.0. Retrieved from The Modeling Agency (2000). www.the-modeling-agency.com/crisp-dm.pdf

Crawford, K.: The hidden biases in big data. Harvard Business Review, Online edn. (2013)

Dwork, C., Hardt, M., Pitassi, T., Reingold, O., Zemel, R.: Fairness through awareness, In: Proceedings of the 3rd Innovations in Theoretical Computer Science Conference, pp. 214–226. ACM (2012)

Fairfield, J., Shtein, H.: Big data, big problems: emerging issues in the ethics and data science of journalism. J. Mass Media Ethics **29**(1), 38–51 (2014)

Floridi, L., Taddeo, M.: What is data ethics? Philos. Trans. R. Soc. **374**, 20160360 (2016)

Firouzi, F., Rahmani, A.M., Mankodiya, K., Badaroglu, M., Merrett, G.V., Wong, P., Farahani, B.: Internet-of-Things and big data for smarter healthcare: from device to architecture, applications and analytics (2018)

Guan, P., Zhou, W.: Business analytics generated data brokerage: law, ethical and social issues. In: Doss, R., Piramuthu, S., Zhou, W. (eds.) FNSS 2017. CCIS, vol. 759, pp. 167–175. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-65548-2_13

Haffar, J.: Have you seen ASUM-DM? Retrieved from IBM (2015) https://developer.ibm.com/predictiveanalytics/2015/10/16/have-you-seen-asum-dm/

Jagadish, H., Gehrke, J., Labrinidis, A., Papakonstantinou, Y., Patel, J.M., Ramakrishnan, R., Shahabi, C.: Big data and its technical challenges. Commun. ACM **57**(7), 86–94 (2014)

Li, Y., Roy, U., Saltz, J.: Modular design of data-driven analytics models in smart-product development. In: ASME 2017 International Mechanical Engineering Congress and Exposition, pp. V011T15A022–V011T15A022. American Society of Mechanical Engineers (2017)

Manogaran, G., Lopez, D., Thota, C., Abbas, K.M., Pyne, S., Sundarasekar, R.: Big data analytics in healthcare internet of things. In: Qudrat-Ullah, H., Tsasis, P. (eds.) Innovative Healthcare Systems for the 21st Century. UCS, pp. 263–284. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-55774-8_10

Metcalf, J., Keller, E., Boyd, D.: Perspectives on big data, ethics and society. Council for Big Data, Ethics and Society (2016). http://bdes.datasociety.net/council-output/perspectives-on-big-data-ethics-and-society/

O'Leary, D.E.: 'Big data', the 'internet of things' and the 'internet of signs'. Intell. Sys. Acc. Fin. Mgmt. **20**, 53–65 (2013)

Saltz, J., Shamshurin, I.: Big data team process methodologies: A literature review and the identification of key factors for a project's success. In: 2016 IEEE International Conference on Big Data (Big Data), pp. 2872–2879. IEEE (2016)

Saltz, J., Shamshurin, I., Connors, C.: Predicting data science sociotechnical execution challenges by categorizing data science projects. J. Assoc. Inf. Sci. Technol. **68**, 2720–2728 (2017). https://doi.org/10.1002/asi.23873

Saltz, J., Heckman, R.: Big data science education: a case study of a project-focused introductory course. Themes Sci. Technol. Educ. **8**(2), 85–94 (2016)

Sandvig, C., Hamilton, K., Karahalios, K., Langbort, C.: An Algorithmic Audit, Data and Discrimination: Collected Essays New America (2014)

Schwartz, P.M.: Privacy, ethics and analytics. IEEE Secur. Priv. **9**(3), 66–69 (2011)

Shearer, C.: The CRISP-DM model: the new blueprint for data mining. J. Data Warehouse. **5**(4), 13–22 (2000)

Stergiou, C., Psannis, K.E.: Recent advances delivered by mobile cloud computing and internet of things for big data applications: a survey. Int. J. Netw. Manag. **27**, e1930 (2017). https://doi.org/10.1002/nem.1930

Stevenson, D.: Locating Discrimination in Data-Based Systems. Data and Discrimination: Collected Essays 16–20. New America (2014)

Strohbach, M., Ziekow, H., Gazis, V., Akiva, N.: Towards a Big Data Analytics Framework for IoT and Smart City Applications. AGT International, Darmstadt (2015)

Wikipedia (2017). http://en.wikipedia.org/wiki/Cross-industry_standard_process_for_data_mining

Tene, O., Polotensky, J.: Privacy in the age of big data. Stanford Law Review (2012)

Wikipedia (2017). http://en.wikipedia.org/wiki/Cross-industry_standard_process_for_data_mining

Zwitter, A.: Big data ethics. Big Data Soc. **1**(2), 2053951714559253 (2014)

References Appendix: List of Codes and Frameworks

Data Science Code of Conduct. Data Science Association. http://www.datascienceassn.org/code-of-conduct.html

ACM Code of Ethics and Professional Conduct. Association for Computing Machinery (1994). https://www.acm.org/about-acm/acm-code-of-ethics-and-professional-conduct

Code of Ethics, Digital Analytics Association. https://www.digitalanalyticsassociation.org/codeofethics

Data Science Ethical Framework. The United Kingdom Government (2016). https://www.gov.uk/government/publications/data-science-ethical-framework

Jagadish, H.: Data Science Ethics. University of Michigan/EdX. https://www.edx.org/course/data-science-ethics-michiganx-ds101x-1#!

The Financial Modeler's Manifesto. The Society of Actuaries (2009). https://www.soa.org/Library/newsletters/risk-management-newsletter/2009/september/jrm-2009-iss17-derman.pdf

VSPReP: Verifiable, Secure and Privacy-Preserving Remote Polling with Untrusted Computing Devices

Amna Qureshi(✉), David Megías, and Helena Rifà-Pous

Internet Interdisciplinary Institute (IN3), Universitat Oberta de Catalunya (UOC), Barcelona, Spain
{aqureshi,dmegias,hrifa}@uoc.edu

Abstract. Internet-based polling systems allow voters to cast their votes at any time during the polling period, from any Internet-connected computing device anywhere in the world. Security is an important feature of such systems that should address inherent concerns, such as secrecy of vote, anonymity and unlinkability of voter, voter coercion, secrecy of intermediate results, verifiability, auditability, and poll integrity. Another major concern is that an infected voting device with a malicious program (e.g., virus, malware) could take control over the vote casting process and make unauthorized and potentially undetected modifications to the voter's voting choices, and, hence, should not be trusted. In this paper we present VSPReP, a verifiable, secure and privacy-preserving remote polling (e-poll) system, which provides vote's privacy and poll integrity, prevents double voting, enables multiple voting (within the allowed polling period), and achieves verifiability (cast-as-intended and tallied-as-recorded) and uncoercibility in the presence of an untrusted voting device. This paper presents a general design of VSPReP and describes its workflow during three polling phases: pre-polling, polling and post-polling. It also analyzes the security properties of VSPReP and evaluates its performance in terms of computational and cryptographic costs. The experimental results show that the average time a voter takes to cast his/her vote is less than 45 secs, thus demonstrating the practicality of VSPReP.

Keywords: Remote polling · Malware detection · Privacy
Verifiability

1 Introduction

In traditional elections, a voter presence is necessary to take part in an election/poll and cast a vote. With the rise and popularity of the Internet and mobile phones, elections/polls could be conducted remotely. Today, a trend towards electronic and Internet voting can be observed, e.g., online polls and surveys are popular in social networks, forums and newspapers. Similarly, till date, 14

R. Doss et al. (Eds.): FNSS 2018, CCIS 878, pp. 61–79, 2018.
https://doi.org/10.1007/978-3-319-94421-0_5

countries have conducted election trials to enable voters to cast votes on the Internet using their own computing devices [14].

Internet-based voting system is based on the voter's computing device (smartphone, tablet, desktop PC, etc.), the Internet, and the voting system. The voter's computing device casts the votes that are sent across the Internet to the voting system, where they are stored and tallied. These three different environments and the information shared between them are vulnerable to various attacks [18], such as voter coercion (a voter is put under pressure or is threatened by a coercer to vote in a particular manner) or vote buying (a voter is offered monetary benefits by a vote buyer to vote in a particular way, or not at all), vote modification due to an infected voting device (a malicious program such as a malware or virus may cause unauthorized and potentially undetected alterations to voter's selected voting choices), theft/forgery of voter identity (an attacker with an access to authentication credentials could cast votes using the identities of a legitimate voter), double voting (an eligible voter may cast multiple votes using his/her authenticated credentials), a coalition of malicious participants (involved parties may collude to alter or eliminate any voter's vote, or cast fake ballots on the behalf of authenticated voter), and disclosure of partial vote tally before the end of the voting period.

Designing a secure Internet-based voting system has become a considerable topic of discussion in the scientific community. A number of schemes have been implemented and deployed in real-world, e.g., Prêt à Voter [17], and Helios [1], which ensure vote privacy as well as verifiability in the presence of untrusted authorities. However, these systems assume that the voting device is trusted for privacy and verifiability. This assumption is unrealistic because a voting device might be controlled by an attacker or host a malicious program. To resolve this problem, many e-voting protocols are proposed to provide three types of verifiability: (1) cast-as-intended verifiability [3,11] that provides a voter with means to make sure that the vote cast by his/her voting device contains the intended voting option, and that no changes have been performed, (2) tallied-as-cast verifiability [12] that allow voters, auditors and third party observers to check that votes tallied corresponds to the cast votes, and (3) end-to-end verifiability [5] that provides both cast-as-intended and tallied-as-cast verifiability. Here, the tallied-as-cast verifiability is divided into two phases: recorded-as-cast (voters can check that their cast votes have been properly recorded in the ballot box) and tallied-as-recorded (voters, auditors and third party observers can verify that the votes published on the BB are correctly included in the tally, without knowing how any voter voted). In the literature, there exists a few protocols that use return codes to provide cast-as-intended verifiability [2,3,9,11], and a Bulletin Board to provide tallied-as-cast verifiability [12]. These return code-based systems send a code sheet containing pre-generated return codes, and finalization codes to the registered voter over a secondary channel (postal mail) before the voting phase. During voting, when the voter selects his/her voting choices and the voting device submits an encrypted vote to the remote voting server, the voting authorities calculate or retrieve return codes corresponding to voter's choices. These codes are sent back to the voter who compares them with the pre-generated return codes printed on his/her code sheet against the selected choices. If matches, the voter finalizes the vote casting process by using finalization codes. The issue of these schemes is that they assume the voting device is not compromised and supports single vote casting.

The scientific community has published a lot of research work in designing secure voting systems for national-level or big elections, and less attention has been paid to develop secure e-polling systems (low-risk or small-level public-opinion systems, where reasonable level of security, privacy, and functionality should be provided to the voter).

In this paper, we propose an e-polling system, VSPReP, inspired by return codes-based protocols [3,6,9] to provide cast-as-intended verifiability in the presence of untrusted voting devices. Also, VSPReP provides recorded-as-cast verifiability (while preserving the privacy of the voter), poll integrity, non-coercibility, resistance against collusion of voting authorities, and supports multiple voting within an allowed polling period, while preventing double voting. The security analysis of the e-polling protocol, and the experimental results of the polling phase (implemented on Java programming language) are presented to show that VSPReP provides a balance between security and functionality.

The rest of the paper is organized as follows. In Sect. 2, we provide the building blocks of VSPReP. Section 3 describes VSPReP in detail. The security analysis and experimental results are discussed in Sect. 4. Finally, Sect. 5, concludes the paper.

2 Building Blocks

A. Distributed ElGamal Cryptosystem: In a distributed cryptosystem, a set of agents cooperate to perform decryption on encrypted messages so as to provide confidentiality by preventing any single agent from decrypting messages. In VSPReP, distributed ElGamal cryptosystem proposed in [10] is used to provide voters' privacy. Distributed ElGamal cryptosystem is a set of three protocols: key generation (*KeyGen*), encryption (*Enc*), and decryption(*Dec*). In *KeyGen* algorithm, a subgroup $\mathcal{G}_p$ is taken on as input which has a generator g of order q of elements in $\mathbb{Z}_p^*$ (a message space of the cryptosystem), where p and q are two large numbers with $p = 2kq + 1$ for some integer constant $k > 0$. *KeyGen* outputs ElGamal public key $y = g^x$ (global and known to all parties), and a secret key x that is shared among t polling organizers $(PO_1, \ldots, PO_t)$ using a polynomial f of degree l over $\mathbb{Z}_q$ such that each polling organizer holds a share $x_i = f(i)$. In *Enc* algorithm, a message $m \in \mathcal{G}_p$, y, and a randomly chosen $r \in \mathbb{Z}_q$ are taken as inputs to compute a cipher-text c: $c = (c_1, c_2) = (g^r, y^r.m)$. For decryption of c, *Dec* algorithm requires all polling organizers to compute decryption shares $d_i = {c_1}^{x_i}$ to output a plain-text message m. To provide verifiability, non-interactive zero-knowledge proofs are computed during *KeyGen* and *Dec* protocols.

B. Pseudo-random Function Based on Decisional Diffie-Hellman (DDH): A pseudo-random function (PRF) is a deterministic-keyed function $F : \mathcal{K}$ x $\mathcal{X} \rightarrow \mathcal{Y}$ (where $\mathcal{K}$ is the set of keys, $\mathcal{X}$ is the domain, and $\mathcal{Y}$ is the range) guaranteeing that a computationally bounded adversary having access to PRF's outputs at chosen points, cannot distinguish between the PRF and a truly random function mapping between the same domain and range as the PRF. In VSPReP, we use a variant of PRF, a key homomorphic PRF (F_{DDH} based on DDH), proposed by Naor et al. [15]. A PRF is key homomorphic if given $F(k_1, m)$ and $F(k_2, m)$, there is a procedure that outputs $F(k_1 \oplus k_2, m)$, where $\oplus$ denotes group operation on k_1 and k_2. F_{DDH} is constructed by considering a cyclic group $\mathcal{G}_p$ of order q, and a hash function $\mathcal{H}_1$: $\mathcal{X} \rightarrow \mathcal{G}_p$ modeled as a random oracle. F_{DDH} is defined as: $F_{\text{DDH}}(k, m) \leftarrow \mathcal{H}_1(m)^k$ with the following homomorphic property, $F_{\text{DDH}}(k_1 + k_2, m) = F_{\text{DDH}}(k_1, m) \cdot F_{\text{DDH}}(k_2, m)$. F_{DDH} is a secure PRF in the random oracle model assuming the DDH assumption holds in $\mathcal{G}_p$.

C. Verifiable Mixnet: Verifiable mixnet is used to provide an anonymous and verifiable tally in electronic voting systems. Verifiable mixnet enables a collection of trustworthy servers to take as input an ordered set of cipher-texts $E = E_1, E_2, \ldots, E_N$ to be re-encrypted using a new randomization value without changing the decryption process. The output is an ordered set of encryptions $E' = E'_{\pi(1)}, E'_{\pi(2)}, \ldots, E'_{\pi(N)}$ (where $E'_{\pi(N)}$ is a re-encryption of E_N, and π is a uniformly random and secret permutation), and non-interactive zero-knowledge proofs π_{mix_t} (where $t = 1, \ldots, N$) of correct mixing. Thus, this re-randomized encryption prevents an adversary to determine the link between the output and the input cipher-texts. The link between elements from input and output is only retrieved in case of conspiring mix-nodes. Verifiability is provided by π_{mix_t}, which is checkable by any party and demonstrates that E' is correctly constructed. The tallying phase (Sect. 3.4) of VSPReP employs the verifiable mixnet proposed in [20].

D. Digital Signature Scheme: A digital signature scheme (e.g., RSA, DSA) is used to provide data integrity, data origin authentication and non-repudiation. In our proposed system, we have used the RSA signature [4] that is made up of three algorithms, (*Gen*, *Sign*, *Verify*), for generating keys, signing, and verifying signatures, respectively. *Gen* is a key generation algorithm that creates an RSA pubic key pk ($pk = (n, e)$), and a corresponding RSA private key sk ($sk = d$), where n is a product of two large distinct prime numbers p and q, e is a public exponent (a randomly generated integer with $1 < e < \phi$, where $\phi = (p-1)(q-1)$), and d is a private unique integer with $1 < d < \phi$. *Sign* is a probabilistic signature algorithm that takes a message m as an input, produces a hash H of m, and then computes a signature S on hash value (H_s) using sk. *Verify* is a deterministic verification algorithm that takes pk and a signature S as inputs to extract hash H_s from S. Also, it computes hash on the received message to generate another hash value (H_v), and compares it with H_s for verification purposes. If both hashes are identical, S is considered valid, otherwise invalid.

E. Crypto MAC: Message Authentication Code (MAC) is a cryptographic primitive that relies on a pseudorandom function to provide authentication, and verification of received messages. A specific type of MAC, the keyed-hash message authentication code (HMAC), is used to provide data integrity and authenticity of the message. HMAC is obtained by using a cryptographic hash function (e.g., SHA256) over the data (to be authenticated) in combination with a secret (symmetric) key. The cryptographic strength of a HMAC depends on the properties of the underlying hash function. The ballot processing phase (Sect. 3.4) and polling codes generation phase (Sect. 3.4) of VSPReP relies on the HMAC algorithm described in [13].

F. Non-Interactive Zero Knowledge Proofs: A non-interactive zero knowledge proof (NIZKP) is a variant of zero knowledge proof that does not require an interaction between the prover and the verifier. The prover computes and sends a statement to the verifier, who either accepts or rejects it. NIZKPs can be obtained in the random oracle using Fiat-Shamir heuristic [8]. To provide verifiability in VSPReP, we have used the following proofs in different phases of the polling: (1) proof of correct encryption based on Schnorr protocol [19] (polling phase), (2) proof-of-equality of discrete logarithms based on Chaum-Pederson protocol [7] (polling phase), (3) proof of correct decryption of ElGamal ciphertexts (π_{dec_t}) (mix and tallying phase), and (4) proof of correct mixing (π_{mix_t}) of ElGamal encryptions in the mixnet (mix and tallying phase).

3 VSPReP Model

This section describes the design and functionality of VSPReP. In Sect. 3.1, we describe the role of each entity. Section 3.2 defines the functionality requirements and the security assumptions. An attack model is described for VSPReP in Sect. 3.3. Section 3.4 describes three phases of VSPReP in detail.

3.1 VSPReP Entities

VSPReP consists of eight basic entities. The functionality of each entity is defined as follows: **(1)** The **voter**(V_k) is a participant who has a valid credential obtained from the credential issuer to cast a vote ($k = 1, \ldots, N$, where N is equal to maximum voters allowed in polling). **(2)** The **voting device** (VD$_S$) is a computing device responsible of casting a ballot given the options selected by V_k. A voter V_k can use as many as S computing devices to cast his/her vote. Besides VD$_S$, V_k uses another computing device as a validation device to receive return and confirmation codes. **(3)** The **polling organization** is a trusted entity that is in-charge of setting up the poll (poll questions and their corresponding voting options, etc.), tallying the votes and publishing the results of the poll. The polling organization consists of t polling organizers: PO$_1 \ldots$ PO$_t$. It is assumed that out of t POs, there is one main PO who manages the remaining POs. **(4)** The **credential issuer (CI)** is a trusted third party that is responsible for authentication and registration of the voter. It provides authenticated voters with the necessary polling credentials (keys and pseudo-identities). **(5)** The **bulletin board (BB)** is a publicly verifiable entity where the results of various steps of the polling process including the final polling result are published by the authorized entities. All the entities of VSPReP have read-only access to BB, whereas some parties have write-only and append-only access to BB. No party is allowed to delete the existing data. **(6)** The **polling server (PS)** checks the correctness of the ballots cast by the authenticated voters, updates, records and stores these ballots into the ballot box. **(7)** The **code generator (CG)** is an entity that manages multiple polling code generators (PCG). In VSPReP, we have assumed six $PCG_{\mathbb{X}}$ ($\mathbb{X} = 1, \ldots, 6$) that are responsible of generating return codes, acknowledgment and confirmation codes to be used in the polling phase. Also, each PCG generates a mapping table to map long-length return codes to small-length return codes. **(8)** The **printing facility (PF)** is in-charge of printing voting options along with their corresponding return codes, polling code sheet identity (PCS$_{ID}$), acknowledgment and confirmation codes. Also, PF in cooperation with CI, delivers polling card sheets to the authenticated voters only.

3.2 Design Requirements and Security Assumptions

In this section, the design requirements and assumptions of VSPReP are described.

A. Design Requirements: In the following, the design requirements related to the construction of VSPReP are defined: **(1)** Only an authenticated voter can use VSPReP on his/her lightweight computing device to cast his/her votes for a maximum of three times. Only the last vote cast by the voter (within an allowed voting period) is considered valid. Double voting by the same voter is not allowed. **(2)** A voter should use the same pseudo-identity (issued by CI at the time of registration) in three rounds of the poll. In case of a new pseudo-identity request, all the previous votes of the voter shall be revoked. **(3)** Three votes from the same voter within a permitted polling period shall be linked together by a tag, which is a poll-specific pseudonym signed by the PS. **(4)** No vote can be linked to the identity of the voter who has cast it. **(5)** All ballots must remain secret while polling is in progress. A voter can read the contents of the BB once the polling phase is finished. **(6)** Since the voting device of the voter is untrusted, the integrity of the vote must be guaranteed, i.e. a verification mechanism is required that should prevent vote's manipulation by the malware-affected device. **(7)** A voter should not be able to provide a proof of his/her vote to any other entity. **(8)** A voter cannot be coerced by a coercer to cast a vote for a specific voting option or abstain from voting. **(9)** No entity can gain any knowledge about the tally before the start of the vote counting phase. **(10)** After the polling phase and before the start of the tallying phase, a voter should be able to verify that the vote cast by his/her voting device corresponds to what he/she intended to cast in the voting phase. **(11)** After the tallying phase, the results should be published on the BB and can be verified by the voter, auditors or passive observers that the final tally is correctly computed from the votes that were cast. Also, a voter should be able to verify that his/her vote was correctly included in the tallying phase. **(12)** The polling system should be efficient and scalable.

B. Design and Security Assumptions: The underlying design and security assumptions of our scheme are described as follows: **(1)** The poll consists of multiple choice questions in which each voter should mark his/her preferences (selecting one option per question) and order them sequentially. **(2)** A voter is allowed to cast his/her vote three times within the allowed voting period using his/her voting device. A tag is used to identify different votes sent by a single voter within the voting period. **(3)** VSPReP assumes a TLS channel between a voting device and the polling server during polling phase. **(4)** The polling protocol of VSPReP depends on the voter using two computing devices (one for casting vote, i.e. a voting device, and another for receiving return and confirmation codes, i.e. a validation device). **(5)** The polling card sheets are provided to each voter through a secure communication channel (post, email, etc.) by a printing facility. Once the polling card sheets are delivered to the voters, PF destroys all the information related to these sheets. **(6)** The existence of PKI is assumed such that any entity who uses the public key of another participant knows that this key belongs to a legitimate party. The RSA and ElGamal key generation is performed offline to generate key pairs. **(7)** Cryptographic primitives and constructions used in VSPReP are secure and verifiable. **(8)** The return, acknowledgment and confirmation codes are composed of 6, 6 and 8 alphanumeric digits, respectively. We have assumed the use of all uppercase and lower-case letters, and digits (0–9). **(9)** Cast-as-intended verifiability depends on the assumption that the voting device and the polling server cannot be malicious simultaneously. **(10)** Generation of polling card sheets is an offline process. In our experimental case, we assume each polling code generator has 50 pre-generated keys (1024-bits) to assist more than $10,000$ users. **(11)** Each voter has access to the general parameters and the public keys, which are made available by the polling server, the polling organizers, the printing facility, the code generator and the polling code generators. **(12)** Six polling code generators are assumed in generation of the polling card sheets. The reason of considering multiple PCGs in VSPReP is to make the system scalable. With an increase in the number of PCGs in the system, the computational cost of generating temporary keys in the return codes generation phase is reduced.

3.3 Threat Model

This sub-section highlights an attack model for VSPReP related to coercion resistance, double voting, vote manipulation by a malicious voting device, and the voter's privacy. The security of the system against these attacks is discussed in Sect. 4.1.

A. Voter Coercion: In voter coercion, the voter may be threatened by a coercer to vote his/her choice of voting options. Once a vote casting phase finishes, the election authority may want to provide a receipt to the voter to allow individual verifiability. The vote coercion attack is possible as long as the voters are able to prove to the coercer how they voted.

B. Double Voting: Remote polling is vulnerable to electoral fraud due to the possibility of double voting, i.e. an authenticated but a malicious voter may request different polling credentials from the credential issuer to cast multiple ballots in the same poll.

C. Vote Modification by a Malicious Voting Device: In a remote electronic polling system, the voters input their vote choices on a privately owned computing devices. If the voting device is infected with a malware, it can modify the voter's choices covertly before these are submitted to the election authorities and, therefore, falsely recorded and counted by the election authority undetectably (without the voter's knowledge).

D. Coalition of Malicious Entities: The following three attacks describe the coalition of malicious VSPReP's participants: (a) A malicious VD$_S$ may form a coalition with PS to generate partial return codes (not corresponding to voter's encrypted voting options) undetectably; (b) PS and CG may collude to infer the voting choices selected by the voter; and (c) After sending the valid confirmation code to the voter, PS may collude with the CG to replace the voter's ballot in the ballot box with the colluded vote.

3.4 Overview of VSPReP

VSPReP, as shown in Fig. 1, consists of 3 phases: pre-polling, polling, and post-polling.

In the pre-polling phase, cryptographic keys and polling parameters are cooperatively generated by the POs of VSPReP. Also, a web address containing the list of voting options for each polling question, a unique poll identity, and a polling period, are generated by the POs. This url is only sent to the authenticated voters on request of CI. A voter gets registered to the system through the voter registration phase, in which the voter receives a unique and valid credential from CI after a successful authentication. Only the authenticated voters receive the polling card sheets (PCSs) (via mail) from the PF on CI's request. A PCS contains return codes, acknowledgment and confirmation codes, and is generated by the CG, six PCGs, and the PF.

During the polling phase, the authenticated voter uses his/her credential to input his/her voting options into VD_S that encrypts the selected voting options, computes and encrypts partial return codes, generates NIZKPs and forms a ballot. The ballot contains an encrypted ID, cipher-text of votes, NIZKPs, time stamp, session ID (a poll-specific pseudonym), and encrypted partial codes. VD_S sends the ballot to the PS via an anonymous and secure channel (TLS). Before

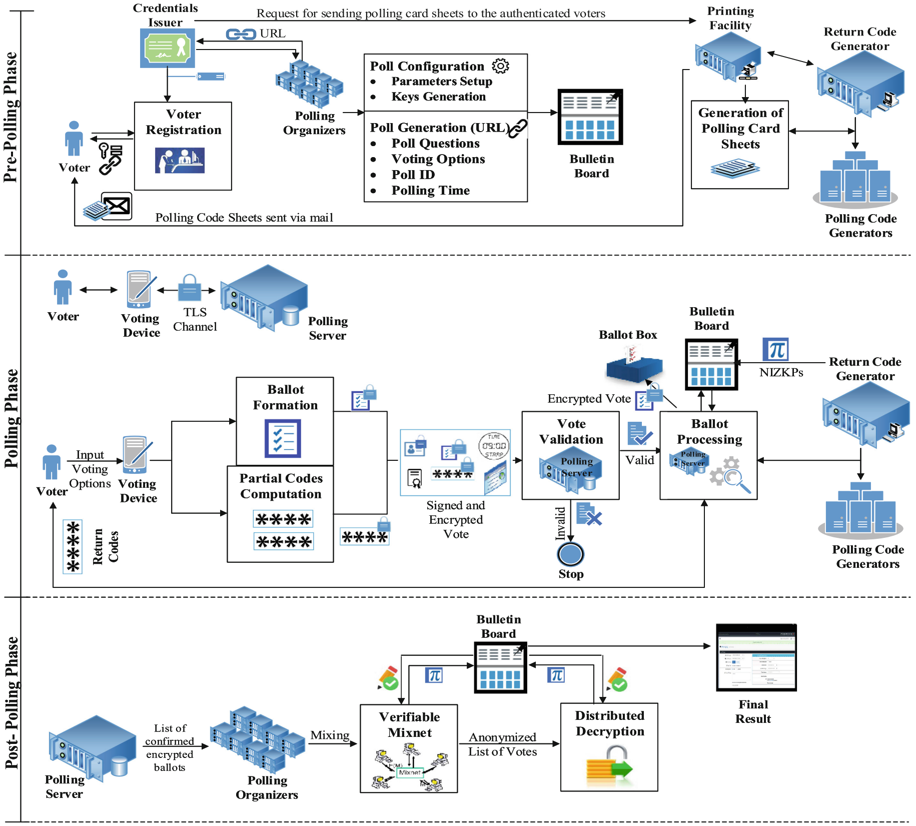


Fig. 1. Overview of VSPReP.

the post-polling phase, vote validation and ballot processing phases are performed to remove duplicate votes, and provide individual verifiability to the voter.

In the post-polling phase, POs input the list of confirmed encrypted votes into a verifiable mixnet that outputs an anonymized list of cipher-texts and NIZKPs (proofs of correct mixing). These cipher-texts are then distributedly decrypted by the POs to reveal the original votes, which are then published on the BB.

In this paper, we have described two phases (generation of a PCS, and a polling phase) in detail due to the fact that these two processes of e-poll protocol address our objectives of providing protection against malware (during polling), prevention of double voting, individual verifiability, and coercion resistance. The post-polling is similar to other voting schemes in the literature that employ mixnets to preserve anonymity of votes.

A. Pre-polling Phase: In this preliminary phase, V_k gets registered, and the polling system is configured: an e-poll is created, cryptographic parameters and keys are generated and published on the BB by POs, and PCS are generated and distributed to the authenticated voters.

I. Voter Registration: To be able to cast a vote, V_k must first register to VSPReP to obtain his/her polling credential from CI. V_k can prove his/her identity (e.g., eID card, a digital certificate issued by a trusted authority, verified email address) to CI, and obtains his/her polling credentials, i.e. a key pair (K_{pV_k}, K_{sV_k}) and a pseudo-identity (we abstract here from the details of authentication and assume that a secure authentication mechanism is used). The pseudo-identity is obtained through a successful run of an interactive protocol [16] between CI and V_k. This protocol results in a shared secret random value r_{V_k} between CI and V_k, which is used along with other identity details of V_k to generate a unique pseudo-identity.

II. Poll Configuration: During polling configuration phase, the polling cryptographic parameters (p, q, g) to be used in ElGamal cryptosystem and homomorphic PRF are defined and published. A cyclic $\mathcal{G}_p \subseteq \mathbb{Z}_p^*$ of quadratic residues modulo a safe prime $p = 2q + 1$ is chosen as a common group for all the cryptographic operations used in VSPReP. The key pairs of PF (K_{pPF}, K_{sPF}), PS (K_{pPS}, K_{sPS}), CG (K_{pCG}, K_{sCG}), and PCGs are generated. Also, PCGs create their joint public encryption key K_{pPCG} and a shared secret decryption key K_{sPCG} using distributed cryptosystem. Similarly, POs create a joint public encryption key and a shared secret decryption key for ElGamal encryption and decryption. Each PO creates its share of the key and posts the public part along with the proofs at BB. BB checks the proofs and combines the shares to form a public election key (K_{pPO}). A message encrypted under K_{pPO} can only be decrypted by K_{sPO} if all POs collaborate. POs and PS are provided with "write" and "append" access to the BB. CG is provided with "write-only" access to the BB. The voters are provided with "read-only" access to the BB. A poll description is generated by POs that contains a unique poll identifier, poll questions, voting options $v_j = \{A, B, C, D\}$ (small bit-length prime numbers $\in \mathcal{G}_p$) for each question, polling time period (t_p), and K_{pPO} to be used by the voters to encrypt their votes before casting them. This data is signed by the main PO, and is appended to the poll description. The link containing the poll description is sent to authenticated voters by CI after a successful registration.

III. Generation of Polling Card Sheets: For the generation of PCSs, CG, $PCG_{\mathbb{X}}$, and PF perform cryptographic operations using their respective key pairs. For proof-of-concept, it is assumed that there are 3 polling questions (Q_i with $i = 1, 2, 3$) with each Q_i having 4 voting options (v_j with $j = 1, 2, 3, 4$) represented by small bit-length prime numbers. For example, the following voting options for 3 questions are generated by the main PO, and are communicated to $PCG_{\mathbb{X}}$ before generation of PCSs: $v_{1j} = \{11, 13, 17, 19\}$, $v_{2j} = \{7, 29, 31, 41\}$ and $v_{3j} = \{433, 53, 5, 47\}$. Each $PCG_{\mathbb{X}}$ randomly picks up a key from a pool of 50 ($l = 1, \ldots, 50$) pre-generated keys. Also, each $PCG_{\mathbb{X}}$ generates a unique key $K_{sess_{\mathbb{X}}}$ of 256-bits. Figure 2 illustrates the following steps performed between CG, $PCG_{\mathbb{X}}$ and PF to generate PCSs.

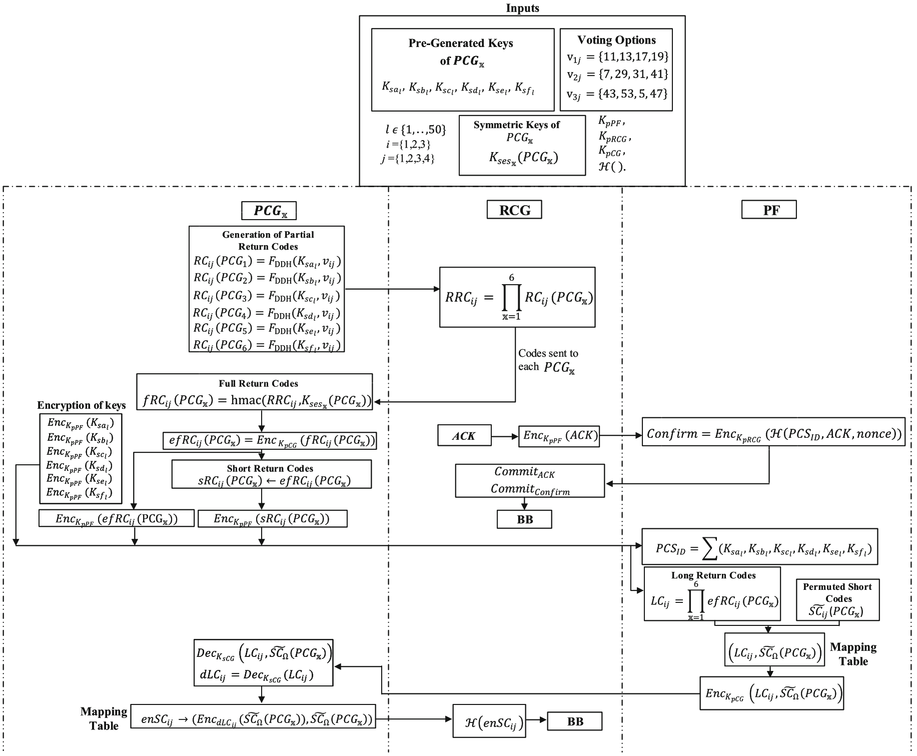

Inputs
Pre-Generated Keys of $PCG_{\mathbb{X}}$
$K_{sa_l}, K_{sb_l}, K_{sc_l}, K_{sd_l}, K_{se_l}, K_{sf_l}$
Voting Options
$v_{1j} = \{11,13,17,19\}$
$v_{2j} = \{7,29,31,41\}$
$v_{3j} = \{43,53,5,47\}$
$l \in \{1,..,50\}$
$i = \{1,2,3\}$
$j = \{1,2,3,4\}$
Symmetric Keys of $PCG_{\mathbb{X}}$
$K_{ses_{\mathbb{X}}}(PCG_{\mathbb{X}})$
K_{pPF}, K_{pRCG}, K_{pCG}, $\mathcal{H}()$.
$PCG_{\mathbb{X}}$
RCG
PF
Generation of Partial Return Codes
$RC_{ij}(PCG_1) = F_{DDH}(K_{sa_l}, v_{ij})$
$RC_{ij}(PCG_2) = F_{DDH}(K_{sb_l}, v_{ij})$
$RC_{ij}(PCG_3) = F_{DDH}(K_{sc_l}, v_{ij})$
$RC_{ij}(PCG_4) = F_{DDH}(K_{sd_l}, v_{ij})$
$RC_{ij}(PCG_5) = F_{DDH}(K_{se_l}, v_{ij})$
$RC_{ij}(PCG_6) = F_{DDH}(K_{sf_l}, v_{ij})$
$RRC_{ij} = \prod_{\mathbb{X}=1}^{6} RC_{ij}(PCG_{\mathbb{X}})$
Codes sent to each $PCG_{\mathbb{X}}$
Full Return Codes
$fRC_{ij}(PCG_{\mathbb{X}}) = \text{hmac}(RRC_{ij}, K_{ses_{\mathbb{X}}}(PCG_{\mathbb{X}}))$
Encryption of keys
$Enc_{K_{pPF}}(K_{sa_l})$
$Enc_{K_{pPF}}(K_{sb_l})$
$Enc_{K_{pPF}}(K_{sc_l})$
$Enc_{K_{pPF}}(K_{sd_l})$
$Enc_{K_{pPF}}(K_{se_l})$
$Enc_{K_{pPF}}(K_{sf_l})$
$efRC_{ij}(PCG_{\mathbb{X}}) = Enc_{K_{pCG}}(fRC_{ij}(PCG_{\mathbb{X}}))$
Short Return Codes
$sRC_{ij}(PCG_{\mathbb{X}}) \leftarrow efRC_{ij}(PCG_{\mathbb{X}})$
$Enc_{K_{pPF}}(efRC_{ij}(PCG_{\mathbb{X}}))$
$Enc_{K_{pPF}}(sRC_{ij}(PCG_{\mathbb{X}}))$
ACK
$Enc_{K_{pPF}}(ACK)$
$Confirm = Enc_{K_{pRCG}}(\mathcal{H}(PCS_{ID}, ACK, nonce))$
$Commit_{ACK}$
$Commit_{Confirm}$
BB
$PCS_{ID} = \sum(K_{sa_l}, K_{sb_l}, K_{sc_l}, K_{sd_l}, K_{se_l}, K_{sf_l})$
Long Return Codes
$LC_{ij} = \prod_{\mathbb{X}=1}^{6} efRC_{ij}(PCG_{\mathbb{X}})$
Permuted Short Codes
$\widetilde{SC}_{ij}(PCG_{\mathbb{X}})$
$(LC_{ij}, \widetilde{SC}_{\Omega}(PCG_{\mathbb{X}}))$
Mapping Table
$Enc_{K_{pCG}}(LC_{ij}, \widetilde{SC}_{\Omega}(PCG_{\mathbb{X}}))$
$Dec_{K_{sCG}}(LC_{ij}, \widetilde{SC}_{\Omega}(PCG_{\mathbb{X}}))$
$dLC_{ij} = Dec_{K_{sCG}}(LC_{ij})$
Mapping Table
$enSC_{ij} \rightarrow (Enc_{dLC_{ij}}(\widetilde{SC}_{\Omega}(PCG_{\mathbb{X}})), \widetilde{SC}_{\Omega}(PCG_{\mathbb{X}}))$
$\mathcal{H}(enSC_{ij})$
BB

Fig. 2. Generation of polling card sheets.

(1) Each $PCG_{\mathbb{X}}$ calculates partial return codes ($RC_{ij}(PCG_{\mathbb{X}})$) using PRF based on DDH assumption for each voting option v_j of Q_i. Each $PCG_{\mathbb{X}}$ uses its selected secret key to obtain $RC_{ij}(PCG_{\mathbb{X}})$ in the following way:

$$RC_{ij}(PCG_1) = F_{\text{DDH}}(K_{sa_l}), \qquad RC_{ij}(PCG_2) = F_{\text{DDH}}(K_{sb_l}),$$
$$RC_{ij}(PCG_3) = F_{\text{DDH}}(K_{sc_l}), \qquad RC_{ij}(PCG_4) = F_{\text{DDH}}(K_{sd_l}),$$
$$RC_{ij}(PCG_5) = F_{\text{DDH}}(K_{se_l}), \qquad RC_{ij}(PCG_6) = F_{\text{DDH}}(K_{sf_l}),$$

where $F_{\text{DDH}}(K, v_{ij}) = \mathcal{H}_1(v_{ij})^K$. **(2)** For each code, $PCG_{\mathbb{X}}$ sends $RC_{ij}(PCG_{\mathbb{X}})$ to CG that computes product of the received partial codes, and sends the result (RRC_{ij}) to each $PCG_{\mathbb{X}}$, e.g., RRC_{11} that corresponds to voting option "1" of Q_1 is computed as: $RRC_{11} = \prod_{\mathbb{X}=1}^{6} RC_{11}(PCG_{\mathbb{X}})$. **(3)** Each $PCG_{\mathbb{X}}$ computes full return code using the received code RRC_{ij} (from CG), and the symmetric key in the following way: $fRC_{ij}(PCG_{\mathbb{X}}) = hmac(RRC_{ij}, K_{ses_{\mathbb{X}}})$. Then, $fRC_{ij}(PCG_{\mathbb{X}})$ is encrypted with K_{pPCG} to obtain encrypted return codes ($efRC_{ij}(PCG_{\mathbb{X}})$) to be sent to PF. Also, each $PCG_{\mathbb{X}}$ generates small-length (64-bits codes) random codes $sRC_{ij}(PCG_{\mathbb{X}})$ that correspond to long (1024-bits $efRC_{ij}$) return codes ($sRC_{ij}(PCG_{\mathbb{X}}) \leftarrow efRC_{ij}(PCG_{\mathbb{X}})$), and encrypts both $sRC_{ij}(PCG_{\mathbb{X}})$

and $efRC_{ij}(PCG_{\mathbb{X}})$ with the public key (K_{pPF}) of PF to be sent to PF. Also, each $PCG_{\mathbb{X}}$ encrypts its secret key used in computation of partial return codes $(RC_{ij}(PCG_{\mathbb{X}}))$ with K_{pPF}, and sends the encrypted key to PF. **(4)** CG generates a random Acknowledgment (ACK) code (64-bits code encoded with Extended ASCII encoding to 6 digits), which a voter uses in the polling phase to provide confirmation of the received return codes. CG encrypts ACK with K_{pPF}, and sends the encrypted code $(Enc_{K_{pPF}}(ACK))$ to PF. **(5)** When PF receives the sets of the return codes (both long and short), the encrypted keys, and $Enc_{K_{pPF}}(ACK)$, it computes the following:

(a) PF decrypts the received encrypted keys using its K_{sPF}, and generates a Polling Card Sheet ID: $PCS_{ID} = \sum(K_{sa_l}, K_{sb_l}, K_{sc_l}, K_{sd_l}, K_{se_l}, K_{sf_l})$.
(b) PF decrypts the received encrypted long return codes with K_{sPF} to compute a long code (LC_{ij}) for each voting option: $LC_{ij} = \prod_{\mathbb{X}=1}^{6} efRC_{ij}(PCG_{\mathbb{X}})$.
(c) PF decrypts $Enc_{K_{pPF}}(ACK)$ to obtain ACK. Also, PF decrypts encrypted short return codes to obtain plain-text short return codes $SC_{ij}(PCG_{\mathbb{X}})$.
(d) PF permutes $SC_{ij}(PCG_{\mathbb{X}})$ with a random permutation key ρ, and then randomly selects permuted $(\widetilde{SC_{ij}}(PCG_{\mathbb{X}}))$ codes, and pairs them with the encrypted long return codes such that it obtains $i \times j$ pairs of return codes to create a mapping table, e.g., in our proof-of-concept, $3 \times 4 = 12$ pairs of codes are generated: $(LC_{ij}, \widetilde{SC_{ij}}(PCG_{\mathbb{X}}))$. Each entry of the mapping table is then encrypted with K_{pPCG}, and sent to each $PCG_{\mathbb{X}}$.
(e) PF generates a confirmation number $(Confirm)$ by using PCS_{ID}, ACK, and a *nonce*: $Confirm = H(PCS_{ID}, ACK, nonce)$. $Confirm$ is used as a proof that the vote has been confirmed by the voter. PF encrypts $Confirm$ with K_{pCG}, and sends the encrypted code to CG.
(f) PF prints the randomly selected 12 short return codes $(\widetilde{SC_{ij}}(PCG_{\mathbb{X}})$ along with the corresponding voting options $\{A, B, C, D\}$, PCS_{ID}, $Confirm$, and ACK as a Polling Card Sheet.

(6) Upon receiving $Enc_{K_{pPCG}}(LC_{ij}, \widetilde{SC_{ij}}(PCG_{\mathbb{X}}))$ from PF, each $PCG_{\mathbb{X}}$ distributedly decrypts the encrypted entries (one time to decrypt the pair, and a second time to decrypt LC_{ij} to obtain long return codes (dLC_{ij})). The short return codes $(\widetilde{SC_{ij}}(PCG_{\mathbb{X}}))$ are encrypted with the corresponding long return codes dLC_{ij} to obtain $Enc_{dLC_{ij}}(\widetilde{SC_{ij}}(PCG_{\mathbb{X}}))$, which is paired with plain-text $\widetilde{SC_{ij}}(PCG_{\mathbb{X}})$ to create a mapping table that contains pairs $(i \times j)$ of return codes:

$$enSC_{ij} = (Enc_{dLC_{ij}}(\widetilde{SC_{ij}}(PCG_{\mathbb{X}})), \widetilde{SC_{ij}}(PCG_{\mathbb{X}})).$$

This mapping table is shared between CG and $PCG_{\mathbb{X}}$. **(7)** CG computes the hash of each pair in the mapping table, and publishes it on BB as commitments to the return codes. Also, CG computes commitments to ACK and $Confirm$ codes, and publishes $Commit_{ACK_i}$ and $Commit_{Confirm_i}$ on the BB (since 3 PCSs will be sent to each voter, thus, BB would contain 3 tables of commitments to the return codes, and $i = 3$ values of $Commit_{ACK_i}$ and $Commit_{Confirm_i}$ for each voter).

B. Polling Phase: Once the pre-polling phase is finished, each V_k may cast his/her ballot using his/her VD_S. VD_S of each V_k creates a ballot with the selected voting options of each polling question (Q_i), and submits it to PS. V_k can cast his/her ballot at most three times ($1 \leq t \leq 3$). Each V_k casts his/her ballot as follows: **(1)** VD_S sets up a TLS connection with PS. PS authenticates the VD_S and receives a session ID (a poll specific pseudonym PI_{Poll}) and a time-stamp (t_s) with the current time. **(2)** V_k selects one option for each Q_i, i.e. it inputs v_j of each Q_i into his/her VD_S. **(3)** VD_S computes a partial ballot as a product of V_k's selected options (v_{ij}): $B_{V_k} = \prod_{i=1}^{3} v_{ij}$. VD_S encrypts B_{V_k} with the joint public key of POs (K_{pPO}) to obtain ElGamal cipher-text: $(c_1, h_1) = Enc_{K_{pPO}}(B_{V_k})$. Also, VD_S generates NIZKP (π_{enc}) to prove knowledge of the randomness used for computing the encryption of B_{V_k}. **(4)** Additionally, V_k inputs a 3-digit (alphanumeric) random number for each voting option ($\gamma_1, \gamma_2, \gamma_3$) into his/her VD_S. **(5)** VD_S concatenates three digits $\gamma_{V_k} = \gamma_1 || \gamma_2 || \gamma_3$, and encrypts γ_{V_k} with K_{pPO}: $(d_1, e_1) = Enc_{K_{pPO}}(\gamma_{V_k})$. VD_S concatenates both the cipher-texts $(c_1, h_1)||(d_1, e_1)$, and generates NIZKP ($\pi_{enc_{con}}$) to prove that $(c_1, h_1)||(d_1, e_1)$ is equivalent to the concatenation of two ElGamal encrypted cipher-texts under K_{pPO}. The concatenated cipher-texts, PI_{Poll} and t_s are digitally signed by VD_S: $\text{Sign}_{K_{sV_K}}((c_1, h_1)||(d_1, e_1), PI_{Poll}, t_s, \pi_{enc_{con}})$. **(6)** V_k inputs his/her PCS_{ID} into VD_S, who would compute partial codes corresponding to voter's selected v_{ij} options using PRF based on DDH assumption with homomorphic properties. VD_S encrypts each computed partial return code with the public key of CG (K_{pCG}). Also, VD_S generates i NIZKPs (π_{PCS_i}) for each computed partial return code. **(7)** The final ballot ($ballot_{V_k}$) submitted by VD_S to PS consists of the following items:

$$ballot_{V_k} = Enc_{K_{pPS}}(ID_{V_k}), (c_1, h_1)||(d_1, e_1), \pi_{enc}, PI_{Poll}, t_s, \pi_{enc_{con}}, t_{V_k},$$
$$\text{Sign}_{K_{sV_K}}((c_1, h_1)||(d_1, e_1), PI_{Poll}, t_s, \pi_{enc_{con}}), Enc_{K_{pCG}}(F_{\text{DDH}}(PCS_{ID}, v_{1j})),$$
$$Enc_{K_{pCG}}(F_{\text{DDH}}(PCS_{ID}, v_{2j})), Enc_{K_{pCG}}(F_{\text{DDH}}(PCS_{ID}, v_{3j})), \pi_{PCS_1}, \pi_{PCS_2}, \pi_{PCS_3}$$

(where t_{V_k} is the time of voting according to VD_S system clock).

I. Vote Validation: When PS receives $ballot_{V_k}$ from VD$_S$, it starts a verification process. PS decrypts $Enc_{K_{pPS}}(ID_{V_k})$ to obtain ID_{V_k}, and checks if there is already an entry of ballot $ballot_{V_k}$ for V_k. If found, then PS checks the value of the flag (FL) that indicates the number of entries of V_k. If $FL = 0$, i.e. $ballot_{V_k}$ is not found against V_k's record, PS continues the validation process by verifying the digital signature and proofs ($\pi_{enc}, \pi_{enc_{con}}$) contained in $ballot_{V_k}$. PS verifies that t_s and PI_{Poll} used in $ballot_{V_k}$ are equal to the ones sent to V_k. If verified, PS creates a new entry for V_k, stores his/her $ballot_{V_k}$, and sets $FL = 1$. In case $FL = 3$, three entries exist for V_k (i.e. the voter has cast his/her vote three times), PS halts the polling process. If PS finds that there is already an entry of V_k and $FL < 3$, it updates t_s and PI_{Poll} in $ballot_{V_k}$, and checks that the new time is more recent than that of an old entry.

II. Ballot Processing: **(1)** After creation or update of V_k's voting record, PS encrypts voter's id (ID_{V_k}) with K_{pCG}, and sends $Enc_{K_{pCG}}(ID_{V_k})$, encrypted partial return codes $(Enc_{K_{pCG}}(F_{\text{DDH}}(PCS_{ID}, v_{ij}))$, and NIZKPs $(\pi_{PCS_1}$, π_{PCS_2}, $\pi_{PCS_3})$ to CG. **(2)** Upon receiving encrypted identity and partial return codes, CG decrypts these using K_{sCG} to obtain ID_{V_k} along with the clear-text of partial return codes. **(3)** CG sends partial return codes, and NIZKPs to each $PCG_{\mathbb{X}}$. **(4)** Each $PCG_{\mathbb{X}}$ verifies NIZKPs, and upon successful verification, computes full return codes using the keyed-PRF and a symmetric key (the same key used to compute the return codes during pre-polling generation of PCS):

$$NRC_{ij}(PCG_{\mathbb{X}}) = hmac(F_{\text{DDH}}(PCS_{ID}, v_{ij}), K_{ses_{\mathbb{X}}}).$$

Each $PCG_{\mathbb{X}}$ encrypts $NRC_{ij}(PCG_{\mathbb{X}})$ with K_{pPCG} to obtain $eNRC_{ij}(PCG_{\mathbb{X}})$:

$$eNRC_{ij}(PCG_{\mathbb{X}}) = Enc_{K_{pPCG}}(NRC_{ij}(PCG_{\mathbb{X}})),$$

$PCG_{\mathbb{X}}$ sends $eNRC_{ij}(PCG_{\mathbb{X}})$ to CG. **(5)** CG computes long return codes NLC_{ij}: $NLC_{ij} = \prod_{\mathbb{X}=1}^{6} eNRC_{ij}(PCG_{\mathbb{X}})$, and sends these long codes to each $PCG_{\mathbb{X}}$. **(6)** Upon receiving NLC_{ij}, each $PCG_{\mathbb{X}}$ looks into its stored mapping table (sent by PF during pre-polling PCS generation) to extract the corresponding short return codes. Each $PCG_{\mathbb{X}}$ encrypts these codes with NLC_{ij}, and sends to CG. **(7)** Upon receiving the encrypted codes, CG uses its stored mapping table (shared with $PCG_{\mathbb{X}}$) to extract the corresponding short return codes $(\widetilde{SC_{ij}}(PCG_{\mathbb{X}}))$. Once matching entries are found, CG encrypts the corresponding $\widetilde{SC_{ij}}(PCG_{\mathbb{X}})$ with K_{pPS}, and sends these to PS. **(8)** PS decrypts $Enc_{K_{pPS}}(\widetilde{SC_{ij}}(PCG_{\mathbb{X}}))$ and sends the plain-text short codes to the voter either via mobile connection or an email (in a form of self-destructing message). **(9)** When V_k receives the message from PS, he/she opens it in his/her validation device, and checks whether the received short codes corresponds to the printed short return codes in the PCS. If all the received codes match with the printed ones, V_k inputs ACK to his/her voting device to finalize the ballot casting phase. **(10)** VD_S encrypts ACK with K_{pCG} and sends $Enc_{K_{pCG}}(ACK)$ to PS.

(a) PS sends $Enc_{K_{pCG}}(ACK)$ to CG, who decrypts it with K_{sPS}, and then performs a check on it to confirm that the received ACK is a valid opening for the $Commit_{ACK_i}$. If yes, CG checks the index of $Commit_{ACK_i}$ since there are three published commitments for each voter. CG checks the number corresponding to index "i" of $Commit_{Confirm_i}$. CG extracts the corresponding $Confirm$ code and encrypts it with K_{pPS} and sends it to PS.

(b) PS decrypts the encrypted code and sends $Confirm$ code to V_k. PS adds the $ballot_{V_k}$ to its ballot box. Only the validated votes with "confirmed" codes would be considered in tallying phase. If $Confirm$ code as displayed by V_k's validation device matches with the $Confirm$ code on his/her PCS, the vote confirmation must have been successful. After the confirmation phase, PS generates hash of ballot and publishes it on the BB (concealed from the voters until the final results are announced).

C. Post-polling Phase: After the polling period (t_p) expires, PS no longer accepts the votes. PS sends the list of cipher-texts $\mathscr{C}_k = (c_k, h_k)||(d_k, e_k)$ (stored in its ballot box) with "Confirmed" status to the main PO. The main PO initiates the mixing process to anonymize the votes such that it is impossible to trace which cipher-text belongs to which voter. To mix $\mathscr{C}_0$, a verifiable mixnet is instantiated based on ElGamal encryption, and the shuffle size (equal to the number of POs, i.e. t). Permutation ($\bar{\mathfrak{r}}$) and re-encryption randomizations ($\mathfrak{s}$) are selected at random. To provide the proof of correctness of the mixing, each of these values ($\bar{\mathfrak{r}}$ and $\mathfrak{s}$) must be generated explicitly. Each PO acts like a mixer, who permutes and re-encrypts the ballots ($\mathscr{C}_k^t = (c_k, h_k)||(d_k, e_k)$) and forwards it to the next mixer (PO). It is necessary to protect the privacy of the vote, as the joint decryption phase will reveal vote contents to allow tallying.

I. Mixing and Tallying Phase: $\mathscr{C}_k^1$ is input to the first PO_1 that chooses a random permutation $\bar{\mathfrak{r}}^{(t_1)}$ and permutes the input list to achieve a new list $\overline{\mathscr{C}_k^1} = \Big\{(c^1_{\bar{\mathfrak{r}}^{(t_1)}k}, h^1_{\bar{\mathfrak{r}}^{(t_1)}k})||(d^1_{\bar{\mathfrak{r}}^{(t_1)}k}, e^1_{\bar{\mathfrak{r}}^{(t_1)}k})\Big\}$. $\bar{\mathfrak{r}}$ only changes the order of the cipher-texts contained in $\mathscr{C}_k$, while the message hidden in the cipher-text remains unchanged. PO_1 re-encrypts ($\overline{\mathscr{C}_k^1}$) using $\mathfrak{s}^1$ to obtain $\mathscr{C}_k^{'1}$. PO_1 submits the mixing result along with the proof (π_{mix_1}) to the BB. BB verifies π_{mix_1} and, on successful verification, posts the mixed vote list for the next PO to mix. $\mathscr{C}_k^{'1}$ is input to the next PO_2, and so on. The output of the last PO_t is the output of the mixing phase. Once all POs have completed the mix, and BB has verified all the proofs ($\pi_{mix_1}, \ldots, \pi_{mix_t}$), the mixing phase is over. The result is an anonymized list of mixed cipher-texts that can be downloaded from BB by each PO to perform decryption using his/her share of secret key, and produce a list of plain-text ballots ($\mathscr{B}_{V_k}||(\gamma_{V_k})$. Each PO must generate and publish NIZKP (π_{dec_t}) on the BB. BB verifies all π_{dec_t} proofs. Once all proofs are validated by BB, the main PO outputs the factors v_{ij} from $\mathscr{B}_{V_k}$ by performing prime factorization. PO checks for each factor to obtain the corresponding voting option, and publishes the output and associated γ_{V_k} on the BB against each polling question.

4 VSPReP Analysis

This section provides an analysis of VSPReP in terms of security and performance.

4.1 Security Analysis

This section discusses the security of VSPReP according to the design requirements and the threat model presented in Sects. 3.2 and 3.3.

A. Coercion Resistance: VSPReP minimizes the possibility of coercion since it allows multiple voting within t_p (with only the last vote being considered valid), and provides multiple PCSs to the authenticated voters. Thus, the voters can always update their votes by using a different PCS, and embedding a new time stamp and a constant PI_{Poll} in the updated ballot before the poll is closed. Therefore, the coercer has no way of knowing if the vote cast in his/her presence and the return codes shown to him/her represents the ballot that was actually counted for that voter. Alternatively, if the coercer has control over the voter's voting device, the voter can forge the contents of the PCS received in the validation device and generate a fake ACK to send to the PS via a controlled voting device. On receiving incorrect ACK, CG would not send a confirmation to the voter, and thus, the ballot would not be considered confirmed and not counted in the tally.

B. Double Vote Prevention: A poll-specific pseudonym PI_{Poll} (signed by PS) is used to identify different votes ($\leq$3) cast by a single authenticated voter during the polling phase to prevent double voting. During vote validation phase, once the PS verifies ID_{V_k}, Sig, and NIZKPs, it checks the received ballot to verify that PI_{Poll} embedded in the ballot matches with the one sent to the voter earlier (step 1 of the polling phase). Since the voter is allowed to vote three times within t_p, the valid ballot must always contain the same PI_{Poll} as described in assumptions (Sect. 3.2). If all other credentials (ID_{V_k}, Sig, and NIZKPs) are verified but PI_{Poll} is not matched, PS halts the polling process. In another possible scenario, a malicious voter may attempt double vote casting by using different identity (pseudo-identity issued at the time of registration). This attack is not possible due to the fact that during three rounds of polling, the pseudo ID of the voter must remain constant. In case of a new pseudo ID request, all the previous votes of the voter shall be revoked by PS on CI's request.

C. Verifiability: Individual verifiability is achieved through the proposed cast-as-intended mechanism based on return codes, which enables the detection of a possible malware attack on the voting device, e.g., if a malicious voting device tries to modify the vote contents, and submit the vote on voter's behalf, the return codes sent to the voter by the PS would not match with the voter's intended voting options. PCGs would also detect the manipulated vote by means of NIZKPs, i.e. the partial return codes and their proofs would not be verified. Moreover, the malicious voting device could not get any information about the received return codes, since the voter uses the validation device to read the message (containing the return codes) received from the PS. In case of mismatch, the voter will then cast his/her vote using a different voting device.

In VSPReP, POs publish the output of the mixing and tallying phase (voting options along with three-digit random codes, associated NIZKPs, and hashes of the confirmed ballots) on the BB so that a voter, any other participant, or auditor can check whether the votes are counted correctly or not. The voters can verify the votes by generating hashes of their submitted ballots, and then compare them to the ones displaying on the BB. Moreover, the published three-digit random code (only known by the voter) on the BB confirms to the voter that his/her vote has been recorded correctly.

D. Privacy of Votes: The possible attacks against the privacy of the votes, as described in Sect. 3.3, can be circumvented in the following ways: (1) a possible coalition between VD_S and PS could not affect vote's privacy, due to the fact that even if a malicious PS verifies incorrect NIZKPs corresponding to manipulated encrypted votes, at the next stage, PCGs would detect the manipulated vote by means of NIZKPs and a voter would not receive any return codes; (2) given the fact that the relation between return codes and the voting options is only known to the voter, neither the PS nor CG/PCGs can use the generated return codes to infer the voter's selected voting options; and (3) after sending the confirmation code to the voter, PS may attempt to collude with CG/PCG to replace the confirmed vote with the colluded vote, i.e. by only replacing the encrypted voting options with their chosen options, and partial return codes computed by brute forcing. However, this attack is infeasible due to the fact PCS_{ID} used by the voter is only known to him/her. Also, at the end of post-polling phase, the voter could compute the hash of the published vote on the BB, and in case of mismatch, complain to the POs of vote manipulation.

4.2 Computational and Cryptographic Costs

We have implemented the polling phase (Sect. 3.4) of VSPReP in Java programming language on a workstation equipped with an Intel i-5 processor at 2.5 GHz and 8 GB of RAM to compute the costs of involved cryptographic operations. Table 1 presents the cryptographic primitives used in the polling phase and the computational costs associated with each operation. The results in Table 1 correspond to 100 runs of each operation on the system (assuming the voter has only cast his/her vote once during t_p). Considering the costs of other operations (computing safe primes, ElGamal key distribution, RSA keys generation, poll setup), on average, a voter requires less than 45 s to cast his/her vote, thus, demonstrating the practicality of the proposed polling protocol.

Table 1. Computational costs of cryptographic primitives.

Phase	Entity	Operations	Time (ms)
VD_S joins with PS	VD_S	TLS	1100
Polling Phase	VD_S	ElGamal Enc of votes and a random no	492
	$VD_S + PS + CG$	RSA Enc/Dec of voter ID	5/38
	$VD_S + CG$	Partial return codes Gen	59
	VD_S	RSA Enc/Dec of partial return codes	11/98
	VD_S	RSA Sig on ballot contents	65
	VD_S	NIZKPs Gen	58
	PS	RSA Sig/NIZKPs Ver	4/545
	PCG	Full return codes Gen	10
	PCG	ElGamal Enc of long and short codes	1895
	$CG + PS$	RSA Enc/Dec of short codes	2/19
	$CG + VD_S$	RSA Enc/Dec of ACK	1/11
	$CG + VD_S$	RSA Enc/Dec of $Confirm$	1/9

In the polling phase, the product of selected voting options, a 3-digit random number, and the long and short return codes are encrypted with ElGamal encryption algorithm, which requires 2 exponentiations each (total 16 modular exponentiations). The voter's pseudo ID, the partial and short return codes, ACK, and $Confirm$ codes are encrypted with the RSA encryption algorithm that requires 1 exponentiation to generate a cipher-text (total 9 exponentiations), and 1 exponentiation to decrypt the cipher-text (total 9 exponentiations). The computation of partial return codes requires 1 exponentiation and M-modular multiplications of each voter selection (1 option per 3 questions that sums up to 3 exponentiations) to polling code sheet ID (a voter-specific key). Two NIZKPs are computed by VD_S: (1) Schnorr identification protocol; and (2) Chaum-Pederson protocol. The generation of first proof requires one modular exponentiation (total 2 for generating ElGamal ciphers) and its verification requires 2 exponentiations (total 4). The second proof requires 2 modular exponentiations (total 6 exponentiations for 3 partial return codes) and its verification requires 4 exponentiations (total 12 exponentiations). The ballot contents are digitally signed using the RSA algorithm that requires one modular exponentiation for signature generation and one modular exponentiation for signature verification. It can be observed that VD_S does not need to perform most expensive cryptographic operations (NIZKP Ver, ElGamal Enc of long and short return codes), which demonstrates the feasibility of implementation of the polling phase on the smartphones.

5 Conclusion

This paper presents a remote polling system, VSPReP, which provides vote anonymity, poll integrity and uncoercibility, and prevents malware infected device to cast a vote on behalf of an authenticated voter during polling phase. VSPReP provides verifiability based on short return codes, a separate voting device, and a BB. To provide cast-as-intended verifiability, VSPReP employs cryptographic primitives to design a complex voting interaction between the voting device and the polling server, which is experimentally shown to be computationally feasible for implementation on portable communication devices. Also, VSPReP supports multiple voting by providing multiple voting sheets, while preventing double voting. As a future work, we intend to address authentication in VSPReP, and develop a working prototype.

Acknowledgment. This work was partly funded by the INCIBEC-2015-02491 "Ayudas para la excelencia de los equipos de investigacin avanzada en ciberseguridad" and TIN2014-57364-C2-2-R "SMARTGLACIS".

References

1. Adida, B.: Helios: web-based open-audit voting. In: SS 2008, pp. 335–348 (2008)
2. Allepuz, J.P., Castelló, S.G.: Cast-as intended verification in Norway. In: EVOTE 2012, pp. 49–63 (2012)

3. Allepuz, J.P., Castelló, S.G.: Internet voting system with cast as intended verification. In: Kiayias, A., Lipmaa, H. (eds.) Vote-ID 2011. LNCS, vol. 7187, pp. 36–52. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-32747-6_3
4. Böck, J.: RSA-PSS provable secure RSA signatures and their implementation (2011). https://rsapss.hboeck.de/rsapss.pdf
5. Benaloh, J., Rivest, R., Ryan, P.Y.A., Stark, P., Teague, V., Vora, P.: End-to-end verifiability (2013). https://www.microsoft.com/en-us/research/publication/end-end-verifiablity/
6. Brelle, A., Truderung, T.: Cast-as-intended mechanism with return codes based on PETs. In: Krimmer, R., Volkamer, M., Braun Binder, N., Kersting, N., Pereira, O., Schürmann, C. (eds.) E-Vote-ID 2017. LNCS, vol. 10615, pp. 264–279. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-68687-5_16
7. Chaum, D., Pedersen, T.P.: Wallet databases with observers. In: Brickell, E.F. (ed.) CRYPTO 1992. LNCS, vol. 740, pp. 89–105. Springer, Heidelberg (1993). https://doi.org/10.1007/3-540-48071-4_7
8. Fiat, A., Shamir, A.: How to prove yourself: practical solutions to identification and signature problems. In: Odlyzko, A.M. (ed.) CRYPTO 1986. LNCS, vol. 263, pp. 186–194. Springer, Heidelberg (1987). https://doi.org/10.1007/3-540-47721-7_12
9. Galindo, D., Guasch, S., Puiggalí, J.: 2015 Neuchâtel's cast-as-intended verification mechanism. In: Haenni, R., Koenig, R.E., Wikström, D. (eds.) VOTELID 2015. LNCS, vol. 9269, pp. 3–18. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-22270-7_1
10. Gennaro, R., Jarecki, S., Krawczyk, H., Rabin, T.: Secure distributed key generation for discrete-log based cryptosystems. J. Cryptol. **20**(1), 51–83 (2007)
11. Gjøsteen, K.: The Norwegian internet voting protocol. In: Kiayias, A., Lipmaa, H. (eds.) Vote-ID 2011. LNCS, vol. 7187, pp. 1–18. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-32747-6_1
12. Joaquim, R., Ribeiro, C., Ferreira, P.: VeryVote: a voter verifiable code voting system. In: Ryan, P.Y.A., Schoenmakers, B. (eds.) Vote-ID 2009. LNCS, vol. 5767, pp. 106–121. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-04135-8_7
13. Krawczyk, H., Bellare, M., Canetti, R.: HMAC: keyed-hashing for message authentication (1997). https://tools.ietf.org/html/rfc2104
14. Mulligan, G.: Has the time now come for internet voting? (2017). http://www.bbc.com/news/business-39955468
15. Naor, M., Pinkas, B., Reingold, O.: Distributed pseudo-random functions and KDCs. In: Stern, J. (ed.) EUROCRYPT 1999. LNCS, vol. 1592, pp. 327–346. Springer, Heidelberg (1999). https://doi.org/10.1007/3-540-48910-X_23
16. Qureshi, A., Megías, D., Rifà, H.: Framework for preserving security and privacy in P2P content distribution systems. ESWA **42**(3), 1391–1408 (2015)
17. Ryan, P.Y.A., Bismark, D., Heather, J., Schneider, S., Xia, Z.: Prêt à voter: a voter-verifiable voting system. IEEE Trans. Inf. Forensic Secur. **4**(4), 662–673 (2009)
18. Schneider, A., Meter, C., Hagemeister, P.: Survey on remote electronic voting. CoRR (2017). http://arxiv.org/abs/1702.02798
19. Schnor, C.P.: Efficient signature generation by smart cards. J. Cryptol. **4**(3), 161–174 (1991)
20. Terelius, B., Wikström, D.: Proofs of restricted shuffles. In: Bernstein, D.J., Lange, T. (eds.) AFRICACRYPT 2010. LNCS, vol. 6055, pp. 100–113. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-12678-9_7

Secret Key Classification Based on Electromagnetic Analysis and Feature Extraction Using Machine-Learning Approach

Naila Mukhtar(✉) and Yinan Kong

Macquarie University, Sydney, Australia
{naila.mukhtar,yinan.kong}@mq.edu.au

Abstract. Despite having a secure algorithm running on a cryptographic chip, in an embedded system device on the network, secret private data is still vulnerable due to Side-Channel leakage information. In this paper, we have focused on retrieving secret-key information obtained from one of the Side Channels, namely Electromagnetic radiation signals. We have captured leaked Electromagnetic signals from a Kintex-7 FPGA, while AES is running over it, and analyzed them using machine and deep-learning based algorithms to classify each bit of the key. Moreover, we aim to analyze the effect of having different signal properties as features in these classification algorithms. The results will help in defining which features give maximum information about the captured signal, hence leading to key recovery.

Keywords: Side-Channel analysis · Embedded system security
Signal-processing · Machine-learning classification
Neural-network classification

1 Introduction

Using Side-Channel analysis to recover key, goes back to early 90s when a group of researchers proposed a method of using the Side-Channel leakage to recover secret information [3,5]. Following the trend, Mulder et al. presented analysis of Electromagnetic radiations emitting out of FPGA to get information about the secret used for encryption [8]. Electromagnetic radiations from circuits due to magnetic fields produced by electric currents. The captured EM radiations are then analyzed to look for secret-information retrieval. Power signals and electromagnetic signal leakage can cause a great risk to secret information, however the later is a variant of the former. Over the last decade, the research focus was on the power-analysis attack as it is convenient to launch, though not practical in all scenarios [4]. On the other hand, Electromagnetic attacks are non-invasive and more practical, with the right probes for signal capturing and the correct analysis for key recovery. De Mulder has shown a way of capturing the EM Radiations

R. Doss et al. (Eds.): FNSS 2018, CCIS 878, pp. 80–92, 2018.
https://doi.org/10.1007/978-3-319-94421-0_6

from FPGAs, processing and analyzing them using mathematical and statistical models to recover a secret key [8]. The defined process works offline and can be time consuming for key retrieval. Similar work is shown by the authors in [7]. Machine learning can help in fast information extraction, based on the classification models being used. Different classification models have been tried and tested for Power analysis signals but not much analysis exists for Electromagnetic signal analysis [25]. Liran and his team have worked on key recovery from AES using machine learning by capturing the power signals emitted out of the device [10,15]. In addition to embedded systems, Genkin et al. found a way of attacking mobile phones using EM analysis [31]. Moreover, neural network based classifiers have been tested for Side-Channel leakage from hardware systems [2]. Our first contribution is to set up a system which is used to capture the EM radiations from a Kintex-7 FPGA while AES is running on the chip (as AES data for Kintex-7 does not exist), secondly we have used the signal properties as features to be fed to classification algorithms, and finally we have analyzed the signals using machine learning and neural network classification techniques (with different signal properties) to classify and recover each secret key bit. Our aim is to find which features (based on EM signal properties) or combinations of features can help in better key bit classification.

The rest of the paper is organized as follows, Sect. 2 explains our methodology for key recovery using classification and outlines the properties of a signal used for feature formation along with feature selection and extraction methods; this section also explains the classification techniques used, Sect. 3 explains the experimental setup and Sect. 4 gives the results of analysis while Sect. 5 concludes the paper.

2 Methodology

The purpose of this research is to capture and analyze the EM radiations out of the FPGA, while AES is encrypting data with a secret key. The analysis is carried out using Machine learning and deep-learning based classifiers rather than traditional statistical methods. To use the classification algorithms, the machine needs to be trained with a set of data. This set of data (EM radiations) is obtained from Kintex-7, mounted over a Sakura-X board. Sakura-X is a series of specialised boards designed to evaluate the algorithm implementation on FPGA, against Power Analysis, EM Analysis and Fault Injection Attacks [20]. The captured raw datasets are then processed to form feature datasets, which are created by using signal properties. Nine signal properties are measured and different combinations of these properties are used as features for training the learning machine, after filtering and evaluating the feature sets using nine standard evaluators/selectors. Reason for using the evaluators is to screen the features to overcome the problem of over-fitting, hence reducing the training time and diminishing the chances of miss-classification. The larger the dataset, the greater are the chances of inaccurate classification. For testing the trained system, another featureset is formed based on the same methodology with a

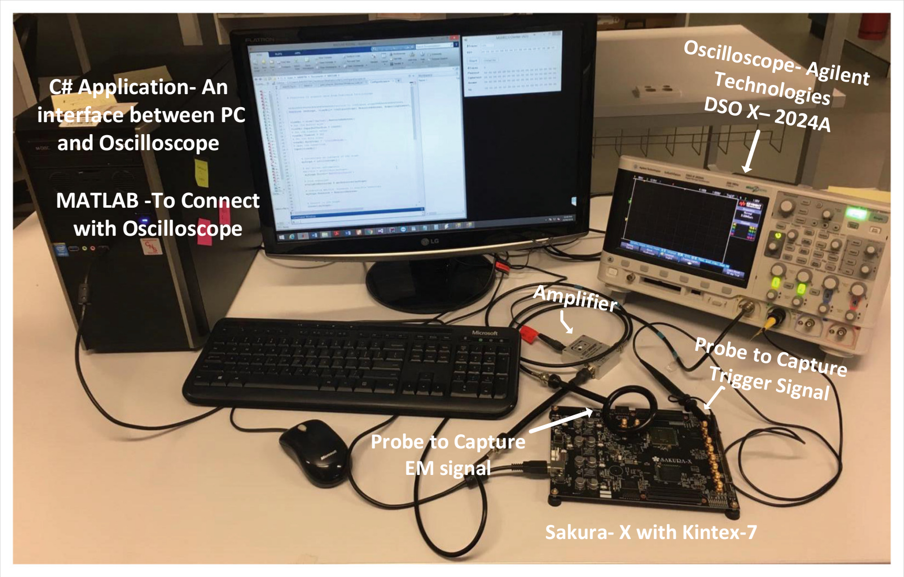

C# Application- An interface between PC and Oscilloscope
MATLAB -To Connect with Oscilloscope
Oscilloscope- Agilent Technologies DSO X– 2024A
Amplifier
Probe to Capture Trigger Signal
Probe to Capture EM signal
Sakura- X with Kintex-7

Fig. 1. Setup for electromagnetic analysis

different secret key. The hardware setup used for acquisition of raw datasets is shown in Fig. 1.

2.1 Advance Encryption Algorithm

The algorithm under test is the Advance Encryption Standard (AES) which is NIST standard for secure communications [19]. It consists of four blocks (Sboxes, ShiftRows, MixColumns and AddRoundKey) for encrypting data, using three different key sizes, i.e. 128, 192 and 256-bit keys having 10, 12 and 14 rounds respectively. All rounds are the same except for the last one which lacks addroundkey block, which makes it vulnerable to Side-Channel attack. Side-Channel attacks can be categorized into Divide-and-Conquer and Analytic attacks [23], [13]. In Analytic attacks, a complete sub-key is recovered using mathematical equations, while in Divide-and-Conquer attacks, only a part of the key is recovered. We will be following the latter approach [15]. To recover the key, one byte is selected at a time and in each byte a single bit is targeted as shown in Fig. 2. For a single byte 256 combinations exist. To mark and classify the samples as ‘1’ or ‘0’, relevant bit location samples are segregated in the form of a group, e.g. to target MSB of the last byte, collected samples having the MSB as 1 are classified as ‘1’ while others are marked as ‘0’. We have collected 100 samples (each sample encrypted with random plaintext) for each possible combination of the fixed key (i.e. total 256 combinations), leading to a total of $100 * 256$ samples as shown in Fig. 5. 51200 samples are then processed using

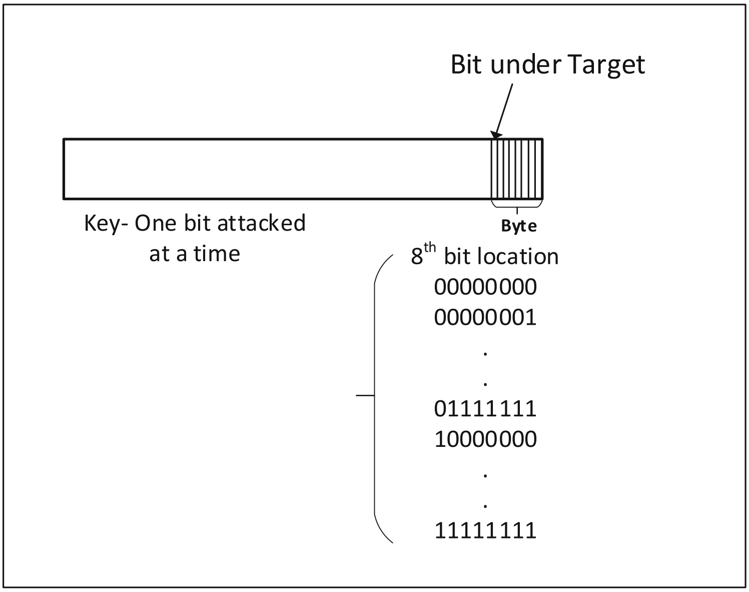


Fig. 2. 256 possible combinations for each bit location

MATLAB to calculate the signal properties (explained in Sect. 2.3). The resulting output datasets are then processed to form feature sets (using Java code) for input to the classification algorithm. Figure 4 shows the preparation process of the input datasets to be fed to classification algorithms (for the training phase), for the most significant bit of the byte under examination, after forming feature sets based on the signal properties. As mentioned before, one aim of this research is to find the features or combination of features which can produce better results for the secret key bit classification (used for this study as mentioned in Sect. 2.2), which has not been analyzed before in the literature for leaked Electromagnetic Radiation.

2.2 Classification Algorithms

Three main classification algorithms have been used for analysis, two machine-learning and one deep-learning based algorithm.

Random Forest. Random Forest is a type of supervised machine-learning classification algorithm in which a number of trees are built during the training process and the classification mode is determined during the testing phase. Random trees use the feature-bagging scheme to build the trees. Details of the algorithm can be found in [29].

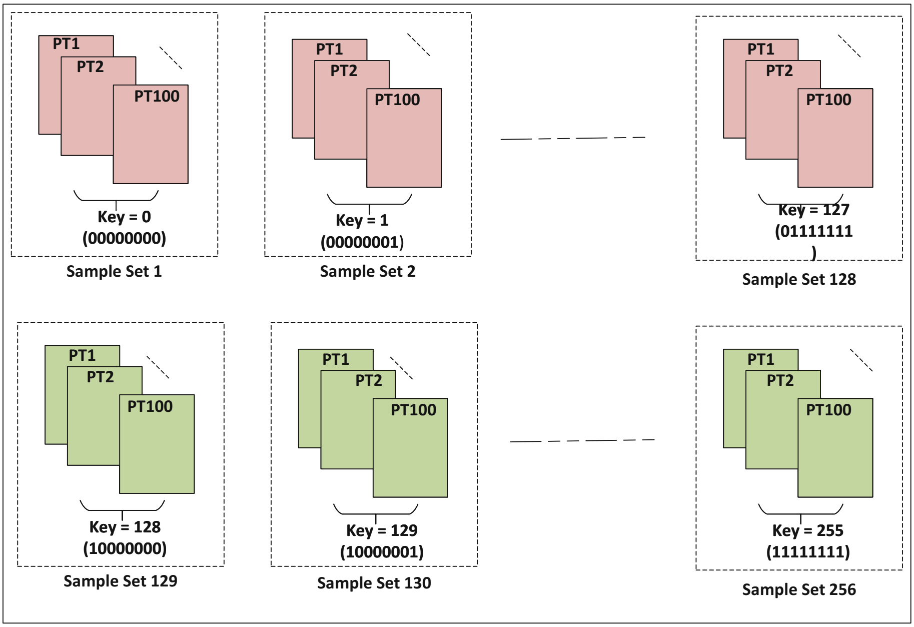


Fig. 3. Sample set formation for a single bit location in a byte

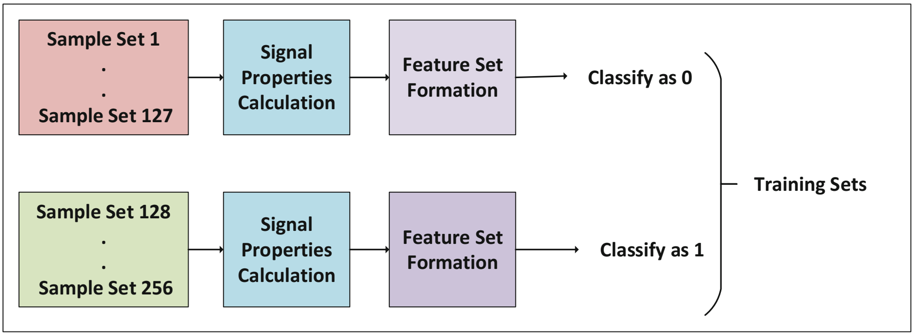


Fig. 4. Process of preparing training and testing data for classification

Naive Bayes. Naive Bayes is a class of supervised learning algorithm, based on Bayes' theorem. It works on a probability model built on the probabilities of outcomes and reveals the uncertainty of the model [30].

MultiLayer Perceptron (MLP). A multilayer Perceptron is a supervised artificial neural network consisting of three or more layers - one input layer, one output layer and two or more hidden layers. Each node in a hidden layer acts as a neuron which works on a nonlinear activation function. MLP is different from a linear perception because of its multiple layers and non-linear activation functions. MLP is best for solving complex problems stochastically (Fig. 3).

2.3 Properties Used as Features

Captured EM signals are subjected to analysis by the above mentioned machine-learning algorithms, based on signal properties (frequency and time domain) as given below.

- Mean of Absolute Value (MAV) - For MAV, the mean of all signals is calculated.
- Slope Sign Change (SSC) - In a signal, slope sign changes are recorded against a pre-determined threshold.
- Sum of Squares (SSI) - In a signal, the sum of the squares of the values is calculated.
- Zero Count (ZC) - The number of times the signal crosses zero is calculated.
- Kurtosis - The sharpness of the peak of a frequency-distribution curve is noted.
- Median PSD (FMD) - The median of a distribution, in the frequency domain, is calculated.
- Mean PSD (FMN) - The mean of a distribution, is recorded.
- Frequency Ratio (FR) - The ratio of the lowest to the highest frequency is calculated.
- Median Amplitude Spectrum (MFMD) - For signals, the median amplitude spectrum is calculated.

2.4 Feature/Attribute Selection and Extraction

Feature Engineering is an important task when it comes to the problem of overfitting in machine-learning classification. It helps in reducing/rearranging, by selecting/extracting those features, which can give the best results. Having too many features can lead to miss-classification. There are two main concepts in feature engineering, used for analysis in this paper and given below.

- Feature Selection: In feature selection, a subset of features is selected from the available pool of feature data.
- Feature Extraction: In feature extraction, a new set of features is formed from existing sets of features.

For our analysis using supervised classification techniques, we need to have defined features from the raw set of data signals captured. We have defined the features based on the signal properties as mentioned in the previous section. Now, from the available set of feature data, we need to form a usable set of features in a dataset on which classification techniques can be applied for training and testing of data, to determine which features will give the best classification results. Below are the feature selection/extraction algorithms used.

Learner-Based Feature Selection - LBS. In this technique, a generic yet powerful algorithm is selected to analyze the performance of the algorithm under test, with subsets of datasets. The subset which performed the best is selected for further analysis. Generally, a decision tree is used as the algorithm.

Chi-Square. Chi-Square is a statistical test which measures the dependency of features on the output variable. If a dependency exists then the features are selected, otherwise they are discarded.

Correlation-Based Feature Selection. The correlation between the attributes and the output variables is calculated using Pearson's correlation coefficient function given in Eq. (1). Attributes having a high correlation (close to −1 or 1) are selected.

$$\text{Correlation} = \frac{\mu_i(1) - \mu_i(0)}{\sigma_i(1) + \sigma_i(0)} \tag{1}$$

μ and σ represent the mean and standard deviation of the features, with respect to class 0 and class 1.

Gain Ratio. Equation (2) can be used for gain-ratio calculation of features based on class.

$$\begin{aligned} Gain(C,A) = H(C) - H(C|A)/H(A) \\ H(C) &= Entropy\,of\,Class \\ H(A) &= Entropy\,of\,Attribute \\ H(C|A) &= Entropy\,of\,Class\,given\,Attribute \end{aligned} \tag{2}$$

Information-Gain Based Feature Selection. For the output variable, information gain or entropy is calculated for each possible feature. Scores/ranks are assigned to the features based on the information contribution towards the output variable. To evaluate it with Weka, ranker is selected. Equation (3) is used to calculate the entropy.

$$Information\,Gain(C,A) = H(C) - H(C|A) \tag{3}$$

One-Rule Attribute Evaluation - OneR. As the name implies, one-rule attribute evaluation means to have one rule set for all predictors. Calculate errors for each predictor, based on frequency table, and select the one which shows least error.

Principal Components. In Principal components, a subset of the data (linearly uncorrelated) is formed based on the original feature set, whose data is correlated. The newly formed variables are known as principal components. It is a kind of data extraction not selection, because a new linearly independent subset is formed.

Relief. This method is based on a feature-weighting approach [16]. A target sample is selected and then the relevance of the features in the neighborhood of that sample are measured. The samples are marked as 'hit' and 'miss', if they belong to the same category as the target sample or to a different category, respectively. After marking, the distance from all hits and misses is calculated for the target sample, and is used as the weight of a target feature.

Symmetric Uncertainty. Ideally, the information gain calculated in Eq. (3) should be symmetric, i.e. the information gained about Y while observing X is same as the amount of information gained while observing Y. Unfortunately, that is not the case, as it is biased towards features with higher values. Moreover, the correlation measured among different features should be normalized and comparable. To handle the information gains' biased behavior towards attributes, the symmetric uncertainty is calculated, which brings out an unbiased response with normalized values in the range of [0, 1]. Equation (4) gives the formula for calculating uncertainty.

$$Symmetric\,Uncertainty = \frac{H(C) - H(C|A)}{(H(C) + H(A)} \tag{4}$$

3 Experiments

3.1 Step 1 - Hardware Experimental Setup

To conduct our experiment, we have captured the EM radiation (shown in Fig. 5) out of the FPGA (Kintex-7), mounted over SAKURA-X and operating at 200 MHz, while AES is running on it. During the encryption process after Sbox, ShiftRows, MixCoulmns and AddroundKey, samples are taken using a KeySight Agilent Oscilloscope. We have targeted one byte of the key at a time, so each bit is classified as '0' or '1'. For each bit classification, we have acquired 100 samples, each consisting of 10k points, for all possible 256 values. Samples are collected using MATLAB and C# platforms. The C# application acts as an interface between the FPGA board (Sakura-X) and the Oscilloscope, which is configured and operated using MATLAB libraries in C#. This gives an automated standalone application for the data collection process without frequent involvement of the user, the GUI is shown in Fig. 6. The application is a modified automated version of the one provided by SAKURA [20].

3.2 Step 2 - Datasets Formation

Once samples are obtained, then features (properties) are calculated using MATLAB customized code. After having a defined set of features, combinations of different features are formed using a Java snippet, written using Weka Libraries [21]. Combinations of features used for analysis are shown in Table 1.

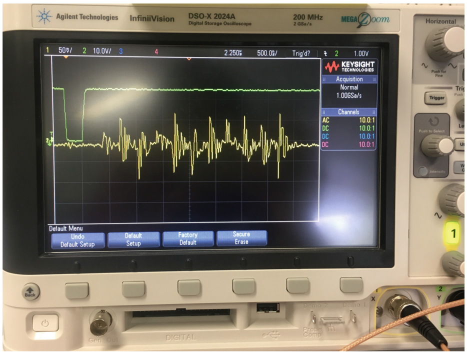
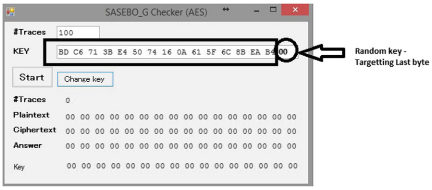


Fig. 5. Captured electromagnetic radiation emitting out while AES is running on SAKURA-X


SASEBO_G Checker (AES)

#Traces 100

KEY BD C6 71 3B E4 50 74 16 0A 61 5F 6C 8B EA B4 00

Start

Change key

#Traces 0

Plaintext 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

Ciphertext 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

Answer 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

Key 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

Random key - Targetting Last byte

Fig. 6. Application GUI- start of app

Table 1. Combinations of feature sets

Combination of feature	Features
Comb-1	MAV
Comb-2	SSC
Comb-3	SSI
Comb-4	ZC
Comb-5	KURTOSIS
Comb-6	FMD
Comb-7	FMN
Comb-8	FR
Comb-9	MFMD
Comb-10	ZC, KURTOSIS, FMD, FMN, FR, MFMD
Comb-11	MAV, SSC, MFMD
Comb-12	MAV, SSC, SSI, FR

3.3 Step 3 - Analysis

The feature sets formed are then subjected to filtering using feature extraction and selection to reduce the number of features. As our target is to test different features and combinations of features with three classification algorithms, so these features' dataset files are used as input to the algorithms, mentioned in Sect. 2.2, for the training phase.

4 Results

At first, the classification accuracy is calculated for all feature sets combinations, as given in Table 1, without using any feature extractor or selectors, for three classifiers (Random Forest, Naive Bayes and MLP). After that, the classification accuracies are calculated for all the feature combination sets using the feature selectors/extractors, and then the difference of accuracies is calculated, to see how much improvement occurred using the newly formed feature sets. It is worth noting that the comparisons are relative and are not based on the best classification algorithm. Our target is to deduce from the analysis which features can improve the results. Figure 7 show results for the analyzed data.

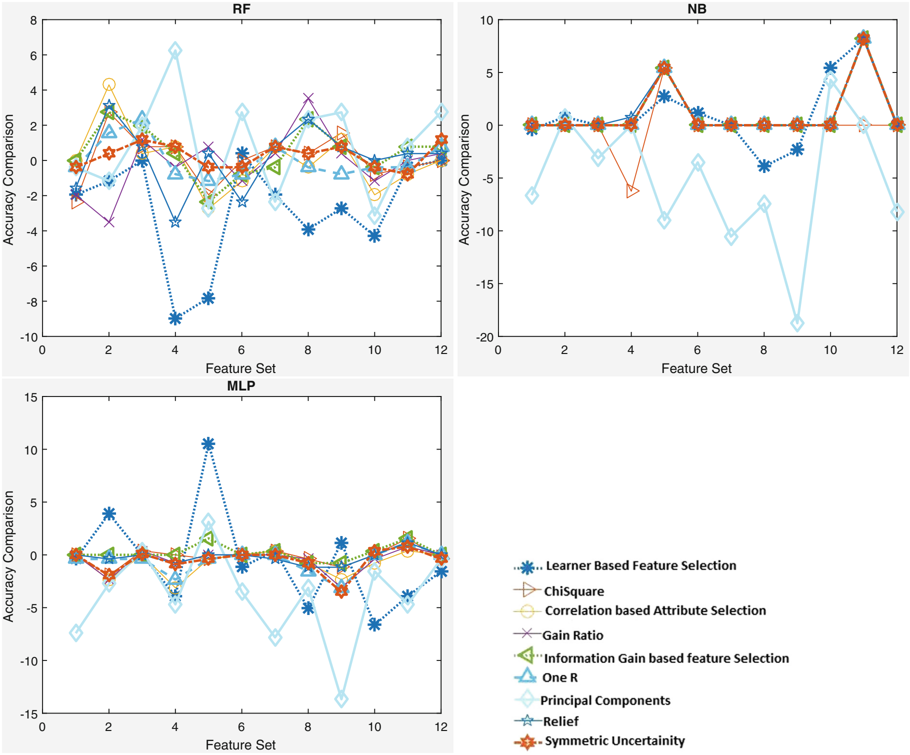


Fig. 7. Accuracies with 100 traces per key bit

4.1 Random Forest (RF)

Varying trends are seen for feature evaluation with Random Forest, as shown in Fig. 7. It is observed that Principal Component Analysis, when applied to Comb-4, gives the best results by increasing the accuracy. Principal Component Analysis performs poorly for Comb-2, Comb-5, and Comb-10. The accuracy gain for Symmetrical Uncertainty and OneR remains almost the same as that of the applied classifier, without any specific features selected. It can be seen that LBS showed decreased accuracy for all feature combinations, however for Comb-4 and Comb-5 the performance is very poor. The gain ratio specifically showed good results for Comb-8. Correlation evaluation, Chi-Square and Information gain exhibits varying trends and gave good accuracy for Comb-2. Overall, it can be seen that every feature combination gave improved results for Comb-12.

4.2 Naive Bayes

The results for Naive Bayes are shown in Fig. 7. It can be seen that there is not much variation in the output accuracies for all combinations. Principal Component Analysis, particularly, performed poorly in all cases, especially for Comb-9. Chi-square accuracy decreased by 7% for Comb-4. All feature extractors and selectors didn't show any improvement at all except for Comb-4 and Comb-11. It can be stated that, for Naive bayes, a combination of MAV, SSC, MFMD is a good choice of features.

4.3 MultiLayer Perceptron

For MLP, variations are seen just like the Random Forest case. Almost all features extractors and selectors behaved in a similar fashion, with insignificant accuracy gain. However, LBS showed surprisingly better performance for Comb-5 and decreased efficiency for the rest of them. Principal Component Analysis decreased the accuracy for Comb-9 by 14%. The accuracies of Comb-1, Comb-11 and Comb-12 are 90.6%, 91.4% and 91.4% respectively. With MLP, MAV alone, the combination of MAV, SSC, MFMD, and the combination of MAV, SSC, SSI, FR are recommended combinations of features.

5 Conclusion

In retrospect, after analyzing the results on the EM radiations obtained from the Kintex-7, we can conclude that, for different classification algorithms, the choice of features or combination of features would be different. For Random Forest, features MAV, combination of MAV, SSC, SSI, FR can be used along with Principal Components. However, for Naive Bayes, MAV and a combination of MAV, SSC, MFMD is best choice, if used with Symmetric Uncertainty and Information Gain. For MLP, MAV alone, the combination of MAV, SSC, MFMD, and the combination of Mav, SSC, SSI, FR are the recommended sets of features. The overall trend shows that a combination of time and frequency-domain features gives better performance for secret-key estimation.

References

1. Gilmore, R., Hanley, N., O'Neill, M.: Neural network based attack on a masked implementation of AES. In: Hardware Oriented Security and Trust (HOST), pp. 106–111. IEEE Computer Society (2015)
2. Maghrebi, H., Portigliatti, T., Prouff, E.: Breaking cryptographic implementations using deep learning techniques. In: Carlet, C., Hasan, M.A., Saraswat, V. (eds.) SPACE 2016. LNCS, vol. 10076, pp. 3–26. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-49445-6_1
3. Kocher, P.C.: Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems. In: Koblitz, N. (ed.) CRYPTO 1996. LNCS, vol. 1109, pp. 104–113. Springer, Heidelberg (1996). https://doi.org/10.1007/3-540-68697-5_9
4. Kocher, P., Jaffe, J., Jun, B.: Differential power analysis. In: Wiener, M. (ed.) CRYPTO 1999. LNCS, vol. 1666, pp. 388–397. Springer, Heidelberg (1999). https://doi.org/10.1007/3-540-48405-1_25
5. Rivest, R.L.: Cryptography and machine learning. In: Imai, H., Rivest, R.L., Matsumoto, T. (eds.) ASIACRYPT 1991. LNCS, vol. 739, pp. 427–439. Springer, Heidelberg (1993). https://doi.org/10.1007/3-540-57332-1_36
6. Levina, A., Sleptsova, D., Zaitsev, O.: Side-channel attacks and machine learning approach. In: FRUCT, pp. 181–186 (2016)
7. Longo, J., De Mulder, E., Page, D., Tunstall, M.: SoC it to EM: electromagnetic side-channel attacks on a complex system-on-chip. Cryptology ePrint Archive, Report 2015/561 (2015)
8. de Mulder, E., Ors, S.B., Preneel, B., Verbauwhede, I.: Differential electromagnetic attack on an FPGA implementation of elliptic curve cryptosystems, pp. 1–6 (2006)
9. Genkin, D., Shamir, A., Tromer, E.: RSA key extraction via low-bandwidth acoustic cryptanalysis. In: Garay, J.A., Gennaro, R. (eds.) CRYPTO 2014. LNCS, vol. 8616, pp. 444–461. Springer, Heidelberg (2014)
10. Lerman, L., Bontempi, G., Markowitch, O.: A machine learning approach against a masked AES. J. Cryptogr. Eng. **5**, 123–139 (2013)
11. Oswald, D., Paar, C.: Improving side-channel analysis with optimal linear transforms. In: Mangard, S. (ed.) CARDIS 2012. LNCS, vol. 7771, pp. 219–233. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-37288-9_15
12. Bhasin, S., Danger, J.-L., Guilley, S., Najm, Z.: Side-channel leakage and trace compression using normalized inter-class variance. Cryptology ePrint Archive, Report 2014/1020 (2014)
13. Renauld, M., Standaert, F.-X., Veyrat-Charvillon, N.: Algebraic side-channel attacks on the AES: why time also matters in DPA. In: Clavier, C., Gaj, K. (eds.) CHES 2009. LNCS, vol. 5747, pp. 97–111. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-04138-9_8
14. Gierlichs, B., Batina, L., Tuyls, P., Preneel, B.: Mutual information analysis. In: Oswald, E., Rohatgi, P. (eds.) CHES 2008. LNCS, vol. 5154, pp. 426–442. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-85053-3_27
15. Lerman, L., Bontempi, G., Markowitch, O.: Power analysis attack: an approach based on machine learning. Int. J. Appl. Cryptogr. (IJACT) **3**, 97–115 (2014)
16. Kira, K., Rendell, L.A.: A practical approach to feature selection. In: Proceedings of the Ninth International Workshop on Machine Learning, pp. 249–256. Morgan Kaufmann Publishers Inc. (1992)
17. Yun, C., Shin, D., Jo, H., Yang, J., Kim, S.: An experimental study on feature subset selection methods. In: Seventh International Conference on Computer and Information Technology, pp. 77–82. IEEE Computer Society (2007)

18. Hospodar, G., De Mulder, E., Gierlichs, B., Verbauwhede, I., Vandewalle, J.: Least squares support vector machines for side-channel analysis. In: 2nd Workshop on Constructive Side-Channel Analysis and Secure Design (COSADE) (2011)
19. NIST, FIPS-197: Advance Encryption Standard (2001)
20. http://satoh.cs.uec.ac.jp/SAKURA/index.html
21. http://www.cs.waikato.ac.nz/ml/weka/
22. Lerman, L., Bontempi, G., Markowitch, O.: Side channel attack: an approach based on machine learning. In: Constructive Side-Channel Analysis and Secure Design, pp. 29–41. Springer (2011)
23. Bogdanov, A., Kizhvatov, I.: Beyond the limits of DPA: combined side-channel collision attacks. IEEE Trans. Comput. **8**, 1153–1164 (2012)
24. Archambeau, C., Peeters, E., Standaert, F.-X., Quisquater, J.-J.: Template attacks in principal subspaces. In: Goubin, L., Matsui, M. (eds.) CHES 2006. LNCS, vol. 4249, pp. 1–14. Springer, Heidelberg (2006). https://doi.org/10.1007/11894063_1
25. Batina, L., Hogenboom, J., van Woudenberg, J.G.J.: Getting more from PCA: first results of using principal component analysis for extensive power analysis. In: Dunkelman, O. (ed.) CT-RSA 2012. LNCS, vol. 7178, pp. 383–397. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-27954-6_24
26. Bohy, L., Neve, M., Samyde, D., Quisquater, J.: Principal and independent component analysis for crypto-systems with hardware unmasked units. In: Proceedings of e-Smart (2003)
27. Standaert, F.-X., Archambeau, C.: Using subspace-based template attacks to compare and combine power and electromagnetic information leakages. In: Oswald, E., Rohatgi, P. (eds.) CHES 2008. LNCS, vol. 5154, pp. 411–425. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-85053-3_26
28. Kocher, P., Lee, R., McGraw, G., Raghunathan, A., Ravi, S.: Security as a new dimension in embedded system design. In: Proceedings of the 41st Design Automation Conference, pp. 753–760 (2004)
29. Breiman, L.: Random forests. Mach. Learn. **45**(1), 5–32 (2001)
30. Mitchell, T.M., Hill, M.: Generative and discriminative classifiers: Naive Bayes and logistic regression. In: Machine Learning (2016)
31. Genkin, D., Pachmanov, L., Pipman, I., Tromer, E., Yarom, Y.: ECDSA key extraction from mobile devices via nonintrusive physical side channels. Cryptology ePrint Archive, Report 2016/230 (2016)

Authentication Protocols for an Object with Dynamic RFID Tags

Selwyn Piramuthu(✉)

Information Systems and Operations Management,
University of Florida, Gainesville, FL 32611-7169, USA
selwyn@ufl.edu

Abstract. A majority of existing RFID authentication protocols consider tagged items that are independent of other tagged items. However, as RFID tags permeate to item-level granularity where several items comprise an object of interest there is a need to develop protocols that seamlessly accommodate inclusion and exclusion of tags on such an object. We propose protocols for this scenario.

1 Introduction

As RFID tags become ubiquitous, there is a need to develop authentication protocols that ensure secure communication between the tagged item and the reader. This is a challenging task given the over-the-air communications medium and the RFID tag resource constraints including its processing capacity, memory, and power source. Since the early 2000s, there has been an explosion of interest in this area both among researchers and practitioners. While there is a vast amount of literature on RFID authentication protocols (e.g., http://www.avoine.net/rfid/index.html), several of the proposed protocols have been plagued by (1) vulnerabilities to attack by a resourceful adversary, and/or (2) the use of primitives that are not lightweight and therefore cannot be implemented in commonly used tags.

Given the diversity of idiosyncrasies streams of research have developed over the years (e.g., [4]). Among the various streams, the ones that deal with the simultaneous authentication of multiple tags are those that evolved from the Yoking Proof introduced by Juels [2]. These protocols authenticate the simultaneous presence of multiple tags in the field of the reader. While this works well for authenticating independent items, there is a need for protocols that consider objects with multiple components with their individual RFID tags. This is predicated on recent trends where, for example, item-level or component-level tagging is in place and these items or components are highly likely to be added or removed from the primary object over time. Objects with multiple RFID-tagged components do not, generally speaking, have the need for authentication of tags as in yoking proof and its variants since these components are attached

R. Doss et al. (Eds.): FNSS 2018, CCIS 878, pp. 93–101, 2018.
https://doi.org/10.1007/978-3-319-94421-0_7

or bundled together to (form) the object. However, these situations dictate a need for continual communication between the object and its component parts as a group.

Consider a supply chain where individual items are RFID-tagged. For example, consider a stack of item-level RFID tagged Wrangler jeans of a certain size (say, 30 × 30) on an RFID-tagged pallet that leave the manufacturing facility to a Walmart warehouse. When several such pallets reach the warehouse, their contents are redistributed and then sent over to individual stores. For example, 20 jeans of size 30×30, 15 jeans of size 36×36, and 25 jeans of size 42×34 maybe included in a pallet that is shipped to a Walmart store in Gainesville, Florida. From the perspective of a pallet, various different items (different quantities of different sizes of Wrangler jeans in this example) are associated with it across different points in time. It should be noted that once its contents are assembled together, the pallet is tracked and traced as a whole and its contents are generally not scanned until it is 'disassembled' and its contents change. Considered at a higher level of granularity, a delivery truck (with RFID reader) can continually communicate with its pallets to determine their destination, which can be modified *en route* when necessary and appropriate. Incidentally, Wrangler jeans' sold in Walmart stores in the U.S. are item-level RFID-tagged beginning August 2010 [1].

Objects with multiple RFID tags are not uncommon. Another example scenario that illustrates this include a primary object (e.g., car chasis) with several attached parts (e.g., car door, wheels) each with its own RFID tag. In such scenarios, both the number of tags as well as the individual tags themselves may vary over time. I.e., when a tire is replaced, the new tire may come with its own embedded RFID tag; when the owner decides to add a GPS system, it may come with its own RFID tag; when the spare tire is removed from the car, there would be one less RFID tag on the car. As seen from these example scenarios, the set of component RFID-tagged items that belong to the main object (here, delivery truck and car respectively) varies over the lifetime of the object (i.e., delivery truck, car). Clearly, there is a need to manage the 'content' of such an object over time from an authentication perspective. Generally speaking, delivery truck X is not interested nor required to know details of the content of delivery truck Y (where X $\neq$ Y) in a similar vein as car A is not interested in information about car B's (A $\neq$ B) speaker system.

We propose authentication protocols that address inclusion and exclusion of several components over time. These protocols avoid some of the identified vulnerabilities of the protocol presented in [5] while being relatively lightweight.

This paper is organized as follows: The next section provides a sketch of the proposed protocol for multiple tags on an object. Section 3 provides an alternative approach to the same scenario. Section 4 provides a brief security analysis of the proposed protocol. Section 5 concludes the paper with a brief discussion.

2 Protocols for Multi-tagged Object

The following notations are used throughout the paper:

- N_t, N_p, N_r, N_u: random n-bit nonce
- s_c, s_{c+1}: group of tags' current and subsequent keys
- $\{\}_k$: keyed (with key k) encryption function
- t_j: shared secret between tag$_j$ and TTP
- r_i: shared secret between Reader R_i and TTP
- id_{t_j}: tag t_j identifier.

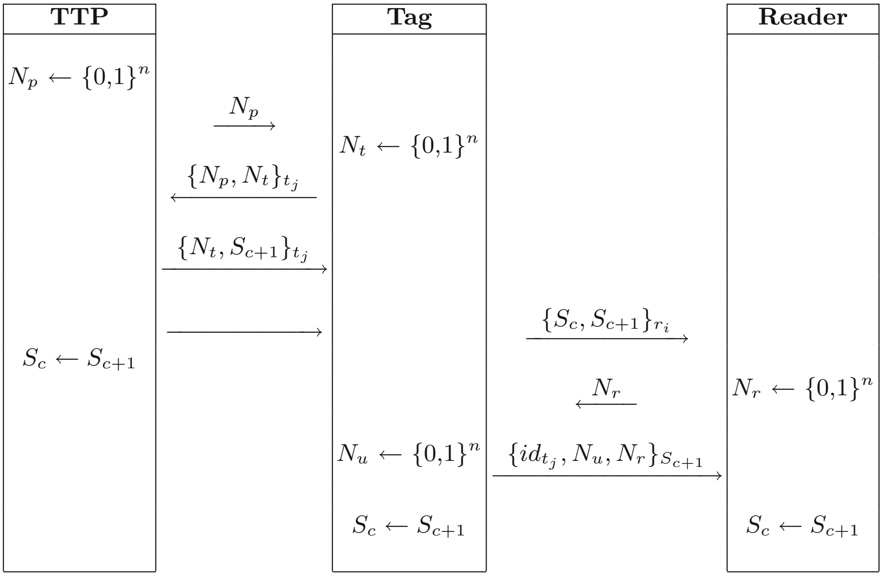


Fig. 1. The proposed protocol

2.1 The Proposed Protocol

There are several entities in this context - a primary object (e.g., car) and a set of component items (e.g., tire, door) that belong to the primary object and the RFID tags on the component items are associated with (the RFID tag on) only one primary object at any given point in time. We do not consider the possibility where a component item could simultaneously belong to several primary objects. The process of inclusion and exclusion of component tags is accomplished in the proposed protocol through a common shared secret key among all the included tags. We assume that a TTP mediates between the reader and tags in accomplishing this change in shared key. The actors involved in this protocol include the reader, the TTP, and every tag that is a part of the object of interest either before or after components (tags) are added or removed.

We assume that every component (tag) that is a part of the object of interest share a common secret key (s_c). This key is updated every time the object of interest experiences addition or removal of a component or group of components. The primary purpose here is to ensure that the updated key is known only to the reader, the TTP, and the tags that are currently attached to the object. The components (tags) that were dropped from this object should not have knowledge of this new shared key. This protocol is repeated for each tag that is associated with the object including those that are present on the object and those that were just removed from the object.

The reasoning for adopting a single common key are (1) ease of key maintenance and (2) fewer messages from reader to tags in the long run since all tags understand any given message that is encrypted with the common key. Drawbacks of this setup include the potential for compromising the entire system when a key is compromised and the initial setup cost of changing every tag's key when a tag enters or leaves the 'system.'

The proposed protocol follows three stages: TTP updates key and communicates this to the component tags, the reader is updated on the new component key, and the reader authenticates the tags.

The TTP initiates the process when a component is either added or removed from the object by generating and sending a nonce (N_p) to all currently existing component tags on the object. These tags then respond by generating a nonce and encrypting both nonce using their shared secret with the TTP (i.e., t_j).

The TTP then sends the updated (group-)key to the component tags encrypted with their shared keys. The component tags update their keys and acknowledge receipt of the same to the TTP. Now, the reader is informed of the new component key through messages that are encrypted using the shared key between reader and TTP.

Finally, the reader authenticates the tags by sending them a nonce and the tags respond by encrypting with the new key a message including their ID, a new nonce and the reader's nonce. This completes the process of updating the common key among the tags.

The new common component key is not known to the (component) tags that were just excluded from the object since they do not receive this new key from the TTP. If and when an excluded tag gets assigned to another object (e.g., an used tire from car A is put on car B after appropriate retreading), the previous reader (here, car A) will not have access to it since the previous reader cannot decrypt communication between TTP and new reader (here, car B).

3 Alternative Approach

The following (set of) protocols may be considered as a solution to the same problem, but the TTP does not have to be invoked for every update of the group key. If we do it this way, then the adversary could record messages and then later crack open a tag to obtain S_c. By repeatedly applying the hash he could end up at the S_c that was used for this encryption. A possible way to address this is to not use S_c for encryption, but a "salted" version of it (e.g. $h(\text{salt}_i, S_c)$. An attacker then additionally needs to have the "salt" which he can only have if he eavesdropped on *all* group key updates.

We propose a set of three protocols to perform different tasks. An *initialization* protocol run between a reader R, a tag T, and a trusted third party TTP. The initialization protocol writes the group key to the tag in a secure and private manner. The *group authentication* protocol authenticates tags to a reader based on the group key. The *group update protocol* updates the group key of a tag to its next value. The next value is the previous value in the hash chain and thus the validity of the new key can be verified by the tag.

We assume that each tag is equipped with an identity id_{t_j} and a key k_j. Readers are equipped with a key r_i. The keys r_i and k_j are shared with the trusted third party. The idea behind our protocols is that tags that belong to the same group share a group key S_c. If a tag has to be included or excluded

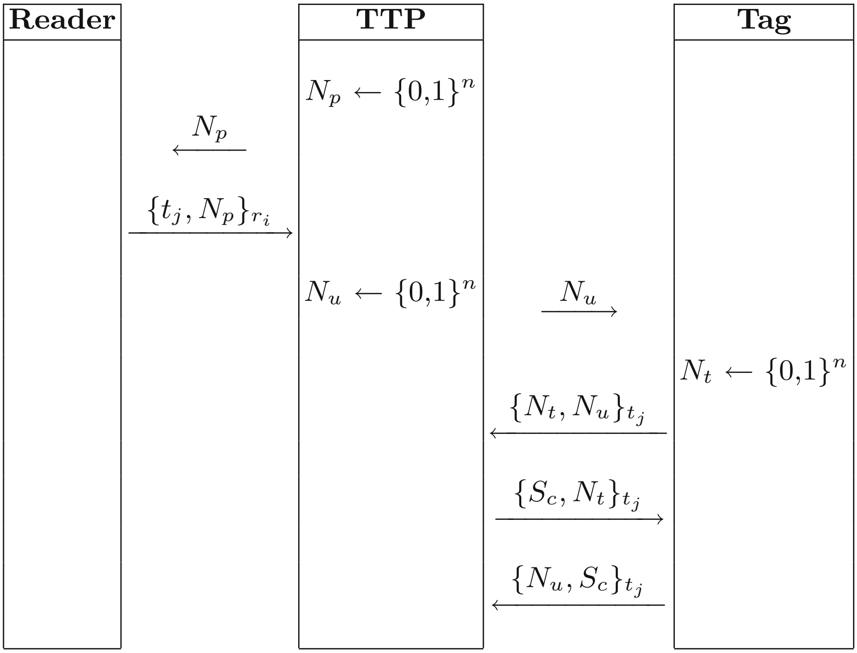


Fig. 2. Initialization protocol

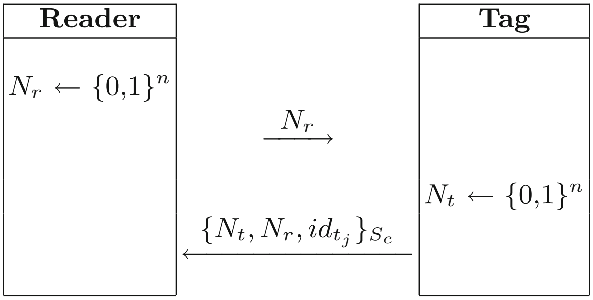

Reader
Tag
$N_r \leftarrow \{0,1\}^n$
N_r
$N_t \leftarrow \{0,1\}^n$
$\{N_t, N_r, id_{t_j}\}_{S_c}$

Fig. 3. Group authentication protocol

all keys get updated by a protocol involving tag and reader. The new group key S_{c+1} is chosen such that $S_{c+1} \leftarrow hash(S_c)$.

The *initialization* protocol is executed between a reader R, a tag T and a trusted third party TTP. It allows the reader to update the group key on a tag without knowing the tag-specific secret k_j. To update the secret, the TTP challenges the reader with a nonce N_p. The reader replies with the group key S_c and N_p encrypted under the secret r_i. The TTP updates the group key on the tag as follows. He challenges the tag with a nonce N_u. The tag generates a nonce N_t and encrypts both these under the key t_j. The TTP now sends the group key S_c and the tag nonce after which the tag updates the group key. Finally, the tag acknowledges the receipt of the message by encrypting the nonce N_u and the group key S_c for the TTP. The protocol is depicted in Fig. 2.

The *group authentication* protocol authenticates a tag T to a reader R based on the group key S_c. The protocol, depicted in Fig. 3, follows a challenge-response

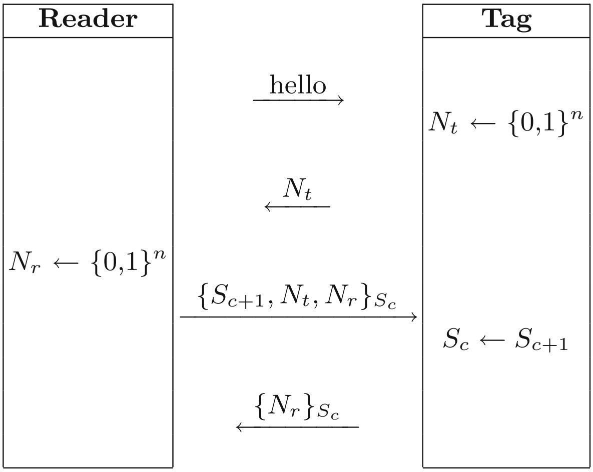

Reader
Tag
hello
$N_t \leftarrow \{0,1\}^n$
N_t
$N_r \leftarrow \{0,1\}^n$
$\{S_{c+1}, N_t, N_r\}_{S_c}$
$S_c \leftarrow S_{c+1}$
$\{N_r\}_{S_c}$

Fig. 4. Group update protocol

structure. The reader R initiates the protocol by sending a nonce N_r to the tag. The tag generates a nonce N_t and replies with the encryption of N_r, N_t, and id_{t_j} under the group key S_c.

The *group update* protocol (see Fig. 4) updates the group key S_c on a tag to the new value S_{c+1}. The reader initiates the protocol by sending a hello message. The tag responds by generating and sending a nonce N_t. The reader generates a nonce N_r and replies with both nonce and the new key S_{c+1} encrypted with the previous key S_c. The tag verifies that S_c is the preimage of S_{c+1} and updates its key. It then responds with the encryption of N_r under the new key.

4 Security Analysis

We do not assume the presence of secure channel between any pairs of entities. We assume the existence of an online TTP. The protocols proposed are to be executed during the physical transfer of a tagged item either into a group or away from a group of items.

The proposed protocols have several characteristics that ensure their security. Freshly-generated nonce (N_p, N_r, N_u, N_t) are used during every run of the protocol. Knowledge of any one of the shared secrets (t_j, r_i) does not lead to any advantage to the adversary since the authentication protocol cannot be successfully completed without knowledge of both the shared secrets (Figs. 1 and 2). However, it is difficult to retrieve any of the shared secrets from passively observing the messages passed among tag, TTP, and reader or even through active capture and modification of messages.

We now consider a few specific attacks on such authentication protocols.

Tag/Reader Anonymity: The tag and reader identification information (e.g., secret keys) are protected from the possibility of information leakage since this information can be used to track and/or trace the tag or (mobile) reader. This is significant since knowledge of such information can allow for the possibility of cloning the tag or reader. We include the possibility of the reader being mobile, as is the case in some RFID applications.

Forward Security: If all shared secrets are somehow known to an adversary, these secrets can be used to decrypt all earlier messages that also include the group key.

Tag/Reader Location Privacy: Since the messages are seemingly random between any two authentication rounds, it is difficult for an adversary to use any of the messages to track the tag and/or the (mobile) reader.

Secrecy/Data Integrity and Authenticity: The integrity of the messages passed between tag and reader is ensured by not sending anything that could compromise the security of the protocol in cleartext. The protocols are designed to be secure and to maintain the secrets regardless of active or passive attacks from adversaries.

DoS/Desynchronization: Since the shared secret keys are not updated after every authentication round, desynchronization is not an issue. The possibility for Denial of Service (DoS) attacks in the proposed protocol is only through blocking and/or modification of message(s). Blocking messages will not grant an adversary any advantage: the reader and TTP wait for acknowledgement message from the recipient of their message within a pre-determined amount of time, and aborts if this does not happen. Modification of any of the messages by an adversary similarly will not allow for protocol compromise. The group key is updated at the very end (Fig. 1), after which the TTP and reader store both the current and previous group keys just in case of DoS attack. These attacks therefore will not succeed.

Passive Replay: Passive replay of any of the three messages that are passed between tag and reader from a previous authentication round will not result in successful authentication due to the existence of N_p, N_t, N_r, N_u that introduce sufficient randomness in the passed messages during each authentication round.

Reader/Tag Impersonation Attack: For an adversary to impersonate a reader, tag, or TTP to one another, it should have the ability to generate messages that seem appropriate and valid to the recipient. An adversary cannot successfully impersonate any entity to any other entity (here, TTP, tag, reader) due to the built-in dependencies among the messages in the authentication protocols.

5 Discussion

Ownership transfer protocols (e.g., [3,6]) are essential for seamless integration of RFID-tagged items in environments such as supply chains. Although not too common at this point in time, it won't be too long before components with RFID tags are put together in a higher-level object with its own RFID tag and possibly a reader. As components enter and leave the domain of the object of interest over time, there is a need to capture this dynamic and be able to deal with the related constraints including those associated with privacy and security issues. The protocol presented in this paper is an attempt at addressing ownership transfer issues from the perspective of component tags and the changing set of ownership from the perspective of the primary object.

It is likely that whenever a group of RFID-tagged items are present, it might be necessary to verify that all these tags are indeed simultaneously present together. We did not consider this scenario since there exist protocols (e.g., Yoking Proof and its variants) that are exclusively designed to accomplish this purpose. Such a protocol can easily be appended to the protocols presented in this paper to form a complete suite of multi-tag authentication and verification protocols.

Acknowledgement. I thank Ton van Deursen for discussions/feedback during the writing of this paper. All error(s) remain my own.

References

1. Bustillo, M.: Wal-Mart radio tags to track clothing. Wall Street J. Bus. Technol. Sect. (2010)
2. Juels, A.: Yoking-Proofs for RFID tags. In: Proceedings of the First International Workshop on Pervasive Computing and Communication Security, pp. 138–143. IEEE Press (2004)
3. Kapoor, G., Piramuthu, S.: Single RFID tag ownership transfer protocols. IEEE Trans. Syst. Man. Cybern. Part C **42**(2), 164–173 (2012)
4. Piramuthu, S.: Lightweight cryptographic authentication in passive RFID-tagged systems. IEEE Trans. Syst. Man Cybern. Part C **38**(3), 360–376 (2008)
5. Piramuthu, S.: Inclusion/exclusion protocol for RFID tags. In: Meghanathan, N., Kaushik, B.K., Nagamalai, D. (eds.) CCSIT 2011. CCIS, vol. 133, pp. 431–437. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-17881-8_41
6. Sundaresan, S., Doss, R., Zhou, W.L., Piramuthu, S.: Secure ownership transfer for multi-tag multi-owner passive RFID environment with individual-owner-privacy. Comput. Commun. **55**, 112–124 (2015)

Big Data and Applications of Future Network Systems

Deconstructing the SME Spectrum from a Knowledge Management Perspective: Proposing an Adapted SECI Model

Shohil Kishore(✉) and David Sundaram

Department of Information Systems and Operations Management,
University of Auckland, Auckland, New Zealand
{s.kishore,d.sundaram}@auckland.ac.nz

Abstract. The term SME [Small to Medium Enterprise] is used extensively, both by practitioners and by academics. However, while both argue the importance of research relevant to SMEs few follow the same definition of the term. The lack of agreement has fostered inconsistency, as within the definition of SME, multiple heterogeneous subcategories exist and research specific to each of those subcategories has not yet received serious academic attention. SME-specific research is essential, as they are not simply scaled-down versions of their larger counterparts; neither are businesses belonging to the SME category identical in terms of their characteristics and reactions. To ensure survival and continual advancement in modern environments, innovation, resourcefulness and particularly, knowledge play a crucial role in long-term success. Thus, deconstructing the differences that exist within the broad range of business categories within the SME spectrum, and how those differences impact knowledge management, offers valuable insights. This article argues that the differences between small (50 employees or less) and medium (250 employees or less) businesses are correlated with the inconsistencies in the literature, and proposes an adapted SECI model to view SME knowledge management in a new light.

Keywords: Knowledge Management [KM]
Knowledge Management Systems [KMS]
Small to Medium Enterprises [SMEs] · Knowledge attrition · SECI

1 Introduction

The foundation of organisational competitiveness has shifted from physical and tangible resources to knowledge [1]. Knowledge is regarded as an asset that is essential to the success of contemporary societies and organisations [2–4]. Due to its dynamism and complexity, knowledge is both difficult to imitate and business-specific which allows for the creation of long-term competitive advantages, cost savings and continual growth when utilised effectively [3, 5, 6]. Knowledge Management [KM] assists organisations in doing so. It aims to reshape organisational culture, structure, systems and technologies in an effort to enhance collaboration, productivity and creativity [7, 8]. Furthermore, knowledge and KM are unique concepts as they are both explicitly

R. Doss et al. (Eds.): FNSS 2018, CCIS 878, pp. 105–116, 2018.
https://doi.org/10.1007/978-3-319-94421-0_8

and implicitly utilised within a wide range of industries, businesses and organisations, including Small to Medium Enterprises [SMEs] [3, 9, 10].

SMEs are considered as the backbone of economic development, competition and innovation in many regions throughout the world [11–13]. For instance, SMEs represent at least 90% of all businesses in America [14], the European Union [15], Australia [16] and New Zealand [13]. However, what constitutes as an SME is different depending on the governing body defining it. The Ministry of Business, Innovation and Employment [13] states that businesses within New Zealand should have less than 50 employees to be considered as an SME. In comparison to other parts of the world, America and the European Union both consider SMEs to have less than 500 and 250 employees, respectively [15, 17]. Even though this is understandable due to the variations in size and population, this difference in definitions has the potential to create issues in terms of generalisability.

1.1 Knowledge Management Systems in SMEs

Knowledge Management Systems [KMS] is defined as a combination of software and technologies designed to support the creation, transfer and application of knowledge [3]. While critical, KMS is not the same as KM. Researchers argue that KM must be seen as a business-wide shift in perspective, not just as the technology that facilitates it [18–21]. They claim that people and culture lie at the core of KM, and should be considered as the focal point during any KM related activities. This ensures that knowledge is utilised to its full potential, and that research regarding KM is applicable, comprehensive, valid and generalizable to a wide range of businesses and scenarios [22, 23].

For instance, specifically focusing on KMS instead of KM restricts the applicability of research [3]. SMEs do not manage their knowledge in the same way as their larger counterparts as their performance is inhibited by a lack of resources which restrains their utilisation of technology [24, 25]. However, knowledge- and KM-related research in the context of SMEs is a necessity as SMEs do extensively exploit knowledge throughout their day-to-day activities, just not exclusively from a technological standpoint. Wong and Aspinwall [1] state that SMEs alternatively transfer and utilise knowledge by verbally communicating with other employees or observing experts perform a task. This allows SMEs to disseminate knowledge to meet deadlines, increase creativity and further differentiate themselves from competitors without the facilitation of technology. Therefore, the success of an SME can be linked to how well they manage and utilise knowledge, both from a technological and non-technological standpoint, and research specific to these topics assists SMEs in comprehensively doing so [26–29].

1.2 SME Heterogeneity

While research from a technological and non-technological perspective is essential, simply focusing on SMEs is insufficient [25, 30]. Heterogeneity exists within the concept of SME, as research relevant to medium-sized businesses (250 employees or less) is not generalizable to small businesses (50 employees or less) [15, 30]. Small businesses face similar challenges to medium-sized businesses; however, their effects are amplified. Resources, technology, capital and available infrastructure are further constrained within the context of small businesses due to a lack of skilled employees, revenue and high-level management [31–33]. In particular, small businesses are much less likely to utilise any form of electronic KM in comparison to medium-sized businesses [21, 23, 34]. Furthermore, a significant proportion of research focuses on medium-sized businesses as they have adopted the European Commission's [15] definition of SME [35–38]. Therefore, research specific to small businesses is a necessity and academics must go beyond abstract SME research to ensure that the different sizes of SMEs have access to information relevant to them [30].

It is evident that a substantial proportion of the existing literature related to KM may not be relevant to small businesses. In addition, as the majority of businesses in most regions are small, not providing those businesses with the appropriate guidance, support and information they need to thrive creates an environment where they are unlikely to continually grow and succeed, and become dominant players within their respective industries [13, 15–17, 39]. Thus, due to a difference in definitions of the term SME [13, 15–17] differences in characteristics and reactions between SMEs [25, 30] and the variability of technological KM implemented by SMEs [1, 25, 35, 39, 40], research relevant to different categories of SMEs may not be generalizable to others.

1.3 Research Objective

This articles adopts the European Union's definition of small (50 employees or less) and medium-sized (250 employees or less) enterprises with the intention of evaluating the existing contention in SME KM literature, and proposes a new perspective on the SECI knowledge creation and transfer model.

2 SME Knowledge Attrition

The general consensus amongst academics is that a decrease in overall organisational knowledge, more commonly known as knowledge attrition, is a serious concern in the context of SMEs [25, 35, 36, 41]. In particular, Wong and Aspinwall [35] contend that SMEs must manage knowledge attrition appropriately, as these businesses are especially prone to its effects. They claim that due to a lack of advancement opportunities and low levels of remuneration, experienced employees are more likely to move onto competing businesses that offer superior salaries or better prospects [42, 43]. In addition, if these employees do leave, they are likely to take their know-how, experience and insights with them, leaving the business with a gap in their organisational knowledge structure [44]. Wickert and Herchel [37] go on to state that filling this gap may be difficult, as new employees take a significant amount of time to acquire knowledge and become accustomed to a business's environment. Therefore, not taking into account the effects of knowledge attrition may jeopardise performance, decrease efficiency and weaken a business's overall robustness [35, 41, 45, 46].

The literature regarding succession planning also contends that every business must have a plan in place to ensure that valuable knowledge is not lost [47–49]. This perspective is supported by the extensive literature regarding succession planning in SMEs [48, 50–52]. In specific, Durst and Wilhelm [25], state that succession planning, which is the "attempt to plan for the right number and quality of managers and key-skilled employees to cover retirements, death, serious illness or promotion, and any new positions which may be created in future organisation plans" [53], is a necessity in ensuring survivability, particularly in SMEs. In other words, Durst and Wilhelm [25] suggest that ignorance regarding knowledge attrition introduces avoidable risk into an already challenging environment.

Conflictingly, other researchers argue that the effects of knowledge attrition within the context of SMEs are somewhat insignificant [39, 54–56]. They state that, unlike larger organisations, most SME employees do not possess skills or experience that would result in significant gaps in the business's knowledge structure if they were to leave. For instance, Desouza and Awazu [39] interviewed a small business manager who stated that each of his employees knew how to perform most business-related tasks. Therefore, each employee in his organisation maintained a similar level of knowledge, and if one employee were to leave, another would be able to perform his or her duties effectively [55]. This suggests that knowledge disperses itself relatively evenly throughout SMEs to form common knowledge, which decreases the amount of specialist skills held by one person, stimulates innovation, encourages creativity and assists in ensuring robustness and survivability [39, 57].

Nevertheless, if a manager or key decision maker were to leave, it would be assumed that the knowledge structure of the business would be severely affected. Evangelista et al. [56] argue against this point, stating that the close social ties that are formed between employees in SMEs are likely to deter employees from completely abandoning their position. In addition, Desouza and Awazu [39] claim that high-level employees in SMEs are much less likely to leave as they may have a personal connection with the business or hold partial ownership. Furthermore, if a high-level employee did decide to leave, in most cases, they would assist in training the next most competent person within the organisation and be contactable due to previously formed personal relationships [58]. Thus, this literature suggests that employees leaving a SME does not necessarily result in knowledge attrition, and that SMEs can generally mitigate the effects knowledge attrition due to the inherent size and structure of their businesses [56].

It is evident that the literature regarding knowledge attrition within the context of SMEs is inconsistent. Some claim that a lack of appropriate planning can have devastating effects on a SMEs long-term competitiveness [25, 35, 41, 45, 46] while others suggest that SMEs indirectly manage knowledge attrition due to the size and structure of their businesses [39, 54–56, 58]. However, it is likely that this conflict has arisen due to the aforementioned heterogeneity that exists within the term SME [30]. For instance, those that advocate the significance of knowledge attrition have adopted the European Commission's [15] definition of SME, and therefore, solely examine SMEs from a medium-sized business perspective (250 employees or less) [35, 41]. This includes the majority of the SME related studies in the field of succession planning [48, 50, 51]. Conversely, academics that criticise the relevance of knowledge attrition within the context of SMEs usually adopt a definition of SME that specifically examines smaller businesses. Desouza and Awazu [39] considered SMEs to have less than 100 employees and primarily focused on small businesses (50 employees or less) to gather the majority of their findings. In addition, Wee and Chua [58] carried out their study in Singapore, therefore examining businesses with less than 50 employees.

As mentioned, Curran and Blackburn [30] and Durst and Runar Edvardsson [25] clarify that heterogeneity exists in the concept of SME, as research relevant to medium-sized businesses may not be generalizable to small businesses. The literature regarding knowledge attrition within the context of SMEs is a key example of this heterogeneity in practice. Research specific to SMEs is not necessarily relevant to every subcategory within the concept of SME. Therefore, the distinct effects of knowledge attrition on different sized SMEs must be clearly identified and supported by valid findings; otherwise, vital research may not be pursued.

3 Clarifying the Difference: Adapting the SECI Model to SMEs

3.1 The Socialisation, Externalisation, Combination and Internalisation [SECI] Model

Academics researching knowledge attrition from the perspective of medium-sized businesses tend to view the effects of knowledge attrition to be more severe, and therefore, have proposed a range of solutions. Durst and Runar Edvardsson [25], Durst and Wilhelm [59] and Wong and Aspinwall [35] suggest that SMEs should document, codify and store knowledge to build up the organisations knowledge base and minimise the effects of knowledge attrition. Furthermore, Wong and Aspinwall [60] suggest that a combination of job rotation, regular training, mentoring, and technologies (such as KMS) would further diminish the effects of knowledge attrition. However, the fundamental characteristics of all of these recommendations are captured in the SECI model [19, 61, 62].

The SECI model, or SECI cycle (Fig. 1), is a widely applicable knowledge creation and transfer model. It describes the knowledge creation and transfer process, and consists of four main elements – socialisation, externalisation, combination and internalisation [19]. According to Nonaka and Toyoma [61], socialisation is the beginning of the knowledge creation process and consists of communicating tacit knowledge between individuals through shared experiences and social interactions rather than written or verbal communication [63]. Tacit knowledge is both vital and difficult to formalise. For example, when trying to teach an apprentice how to perform a task, experts may perform the task themselves as the apprentice observes. However, writing down how to perform the same task may be difficult due to the contextual nature of certain tasks and skills. Tacit knowledge can also be acquired implicitly, where there is no intention to teach or learn [64].

The process of externalisation represents the articulation of tacit knowledge to form explicit knowledge. Explicit knowledge is simpler to articulate, capture, store, edit and share. Even though tacit knowledge is relatively difficult to formalise, experts may still

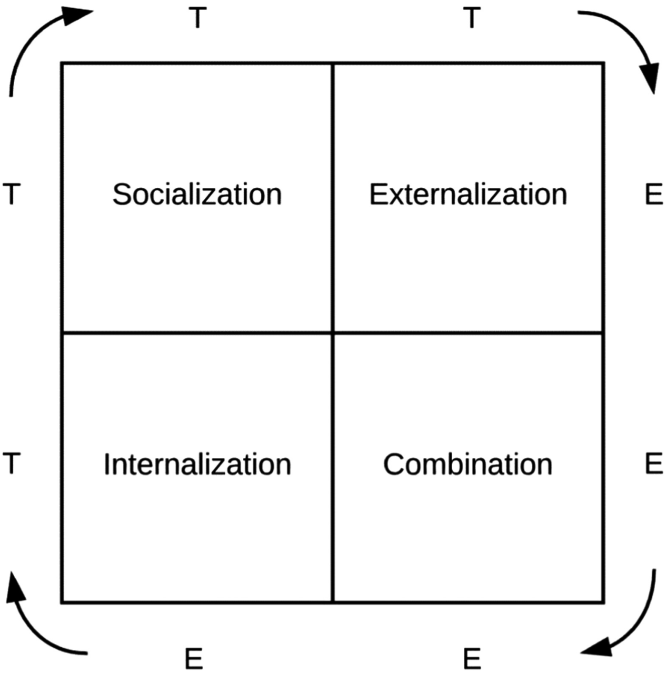


Fig. 1. The SECI model (adapted from Nonaka and Konno [63])

try to express their knowledge to others in the form of dialogue, and build upon their knowledge through discussion [63]. Combination refers to the collection, processing and editing of explicit knowledge to form a more comprehensive knowledge base. This includes collecting and integrating information from internal and external sources, disseminating knowledge, and further processing explicit knowledge to enhance usability [63]. Finally, internalisation represents the process where knowledge is utilised in practical situations to recreate tacit knowledge. For instance, new employees may read relevant manuals and documents to carry out a particular action. In doing so, employees "learn by doing", enriching and expanding their own tacit knowledge base [19, 61, 65, 66].

3.2 Proposing an Adapted SECI Model from the Perspective of SMEs

KM research from the perspective of medium-sized businesses contend that adopting the SECI model, or at least elements of the SECI model, is a necessity as it ensures that the effects of knowledge attrition are minimised [1, 25, 67]. In addition, they argue that the facilitation of software and technology (such as KMS) allows knowledge to be more effectively collected, articulated, integrated, transferred, utilised and disseminated throughout the organisation [1, 68, 69]. However, academics who pursued KM related research in the context of smaller businesses contend contradictory results. DiPasquale and McInerney [70] state that the SECI cycle from the perspective of small businesses is distorted due to a significant emphasis on informal tacit knowledge transfer. This notion is verified by Desouza and Awazu [39]. They state that due to the size of small businesses, employees and managers normally work in close proximity to one another. Therefore, not only do employees and managers informally communicate on a regular basis, but also, this type of regular communication fosters an environment where knowledge is shared throughout the organisation and internalised through action. New employees may ask their co-workers questions and adjust their reactions to certain scenarios based on others [64, 71]. Moreover, as small businesses do not readily implement or maintain KMS [40], gather and collate knowledge on a regular basis [39] or maintain comprehensive manuals or guidelines [72], the importance of the socialisation and internalisation elements are further amplified.

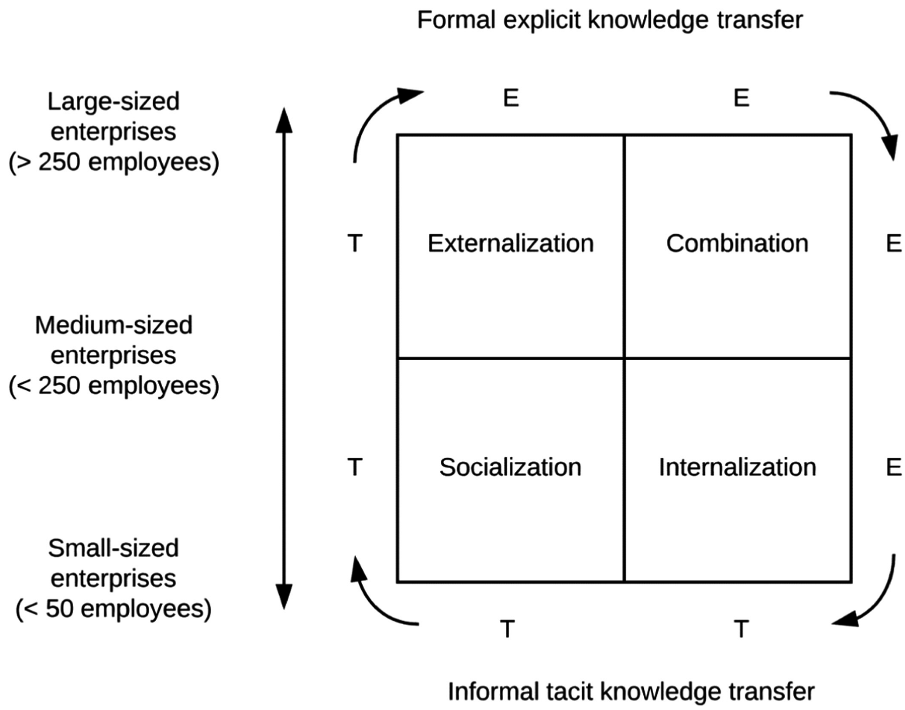

Formal explicit knowledge transfer
Large-sized enterprises (> 250 employees)
Medium-sized enterprises (< 250 employees)
Small-sized enterprises (< 50 employees)
E
E
T
Externalization
Combination
E
T
Socialization
Internalization
E
T
T
Informal tacit knowledge transfer

Fig. 2. The SECI model from the perspective of the SME spectrum (adapted from Nonaka and Konno [63])

Based on this, we propose that the SECI model exists on a spectrum where the focus on certain types of knowledge and knowledge transfer vary depending on the size of the organisation, especially within the context of SMEs [39] (Fig. 2). According to the literature, small businesses tend to focus more deeply on transferring knowledge (primarily tacit knowledge) through joint activities, experiences and interactions with those throughout the organisation [39, 70]. On the other hand, medium-sized businesses face challenges in disseminating tacit knowledge due to their inherent size but still maintain some level of tacit knowledge transfer due to interpersonal relationships and organisational structure [25, 35, 39, 44, 73]. Nevertheless, knowledge in medium-sized enterprises is primarily transferred explicitly (through videos, guides, workflows and KMS), in a more structured and communicative format. Thus, as an organisation becomes larger, informal knowledge transfer tends to decrease and formal knowledge transfer tends to increases (Fig. 3).

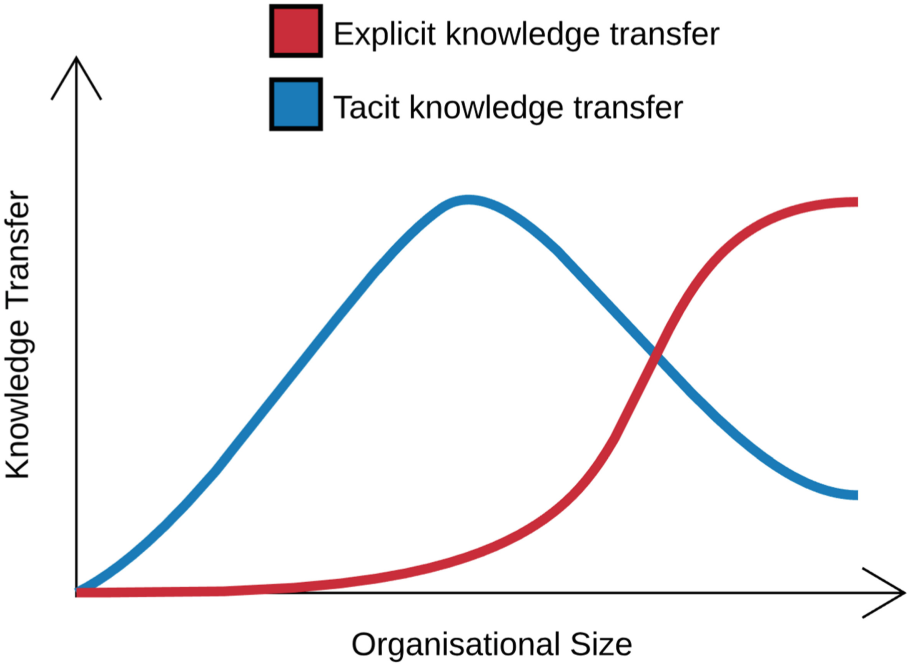


Fig. 3. The influence of organizational size on knowledge transfer

4 Conclusion

This research adopts the European Union's definition of small (50 employees or less) and medium-sized (250 employees or less) enterprises with the intention of evaluating the existing contention in SME KM literature. While size is not the only salient factor in understanding how SMEs manage knowledge, these findings suggest that KM research specific to SMEs may not be generalizable to every subcategory that exists within the concept of SME as researchers adopt varied definitions of the term. The article goes on to explore these inconsistencies from the perspectives of tacit and explicit knowledge transfer, and the four phases of the SECI cycle. Results suggest that the size of an organisation has a significant impact on the type of knowledge transferred.

This discrepancy in the literature is incorporated into an adapted version of the SECI model (Fig. 2) which proposes that smaller businesses tend to focus on socialisation and internalisation when transferring knowledge while medium-sized businesses tend to focus on externalisation and combination. As size increases, informally disseminating knowledge throughout an organisation becomes more difficult, so more structured knowledge transfer methods must be incorporated. In doing so, smaller businesses effectively transfer knowledge regarding culture, context and the intricacies associated with an organisation while medium-sized organisations allocate more resources to explicit knowledge management.

Further empirical research needs to be conducted to validate the proposed model and understand the impact of inconsistency in the literature. There is also a need to understand how small businesses can convert their abundant tacit knowledge to explicit knowledge, keeping in mind their limited resources. This could include the adoption of technological solutions (such as KMS) or non-technological solutions.

References

1. Wong, K.Y., Aspinwall, E.: An empirical study of the important factors for knowledge-management adoption in the SME sector. J. Knowl. Manag. **9**, 64–82 (2005). https://doi.org/10.1108/13673270510602773
2. Grover, V., Davenport, T.H.: General perspectives on knowledge management fostering a research agenda. J. Manag. Inf. Syst. **18**, 5–21 (2001). Grover, Davenport - 2001
3. Alavi, M., Leidner, D.: Review: knowledge management and knowledge management systems: conceptual foundations and research issues. MIS Q. **25**, 107–136 (2001)
4. Chou, S.W.: Knowledge creation: absorptive capacity, organizational mechanisms, and knowledge storage/retrieval capabilities. J. Inf. Sci. **31**, 453–465 (2005). https://doi.org/10.1177/0165551505057005
5. Becerra-Fernandez, I., Sabherwal, R.: Organizational knowledge management: a contingency perspective. J. Manag. Inf. Syst. **18**, 23–55 (2001). https://doi.org/10.1080/07421222.2001.11045676
6. Sveiby, K.: A knowledge-based theory of the firm to guide in strategy formulation. J. Intellect. Cap. **2**, 344–358 (2001). https://doi.org/10.1108/14691930110409651
7. Du Plessis, M.: Drivers of knowledge management in the corporate environment. Int. J. Inf. Manag. **25**, 193–202 (2005). https://doi.org/10.1016/j.ijinfomgt.2004.12.001
8. Gurteen, D.: Knowledge creativity and innovation. J. Knowl. Manag. **2**, 5–13 (1998). https://doi.org/10.1108/13673279810800744
9. Ngah, R., Jusoff, K.: Tacit knowledge sharing and SMEs' organizational performance. Int. J. Econ. Financ. **1**, 216–220 (2009). https://doi.org/10.5539/ijef.v1n1P216
10. Kaminski, P.C., de Oliveira, A.C., Lopes, T.M.: Knowledge transfer in product development processes: a case study in small and medium enterprises (SMEs) of the metal-mechanic sector from São Paulo, Brazil. Technovation **28**, 29–36 (2008). https://doi.org/10.1016/j.technovation.2007.07.001
11. Chong, S.C., Lin, B.: Exploring knowledge management (KM) issues and KM performance outcomes: empirical evidence from Malaysian Multimedia Super Corridor companies. Int. J. Technol. Manag. **43**, 285 (2008). https://doi.org/10.1504/IJTM.2008.020552

12. Sin Tan, K., Choy Chong, S., Lin, B., Cyril Eze, U.: Internet-based ICT adoption: evidence from Malaysian SMEs. Ind. Manag. Data Syst. **109**, 224–244 (2009). https://doi.org/10.1108/02635570910930118
13. Ministry of Business Innovation and Employment: The Small Business Sector Report, Auckland (2014)
14. Keating, R.J.: Free Trade Agreements, Exports and Small Business. http://sbecouncil.org/2015/03/04/free-trade-agreements-small-business-and-exports/
15. European Commission: What is an SME?. http://ec.europa.eu/growth/smes/business-friendly-environment/sme-definition/index_en.htm
16. Australasian SME Alliance: SME Facts. http://www.asmea.org.au/SMEFacts
17. Small Business & Entrepreneurship Council: Small Business Facts & Data. http://www.sbecouncil.org/about-us/facts-and-data/
18. Sveiby, K.E.: The New Organizational Wealth: Managing and Measuring Knowledge-Based Assets. Berrett-Koehler Publishers, San Francisco (1997)
19. Nonaka, I., Takeuchi, H.: The Knowledge-Creating: How Japanese Companies Create the Dynamics of Innovation. Oxford University Press, New York (1995)
20. Salojärvi, S., Furu, P., Sveiby, K.: Knowledge management and growth in Finnish SMEs. J. Knowl. Manag. **9**, 103–122 (2005). https://doi.org/10.1108/13673270510590254
21. Rubenstein-Montano, B., Liebowitz, J., Buchwalter, J., McCaw, D., Newman, B., Rebeck, K.: A systems thinking framework for knowledge management. Decis. Support Syst. **31**, 5–16 (2001). https://doi.org/10.1016/S0167-9236(00)00116-0
22. Bobbitt, L.: Implementing knowledge management solutions. In: International Knowledge Management Summit, San Diego, pp. 29–31 (1999)
23. McDermott, R.: Why information technology inspired but cannot deliver knowledge management. In: Knowledge and Communities, p. 272 (2000)
24. Jarillo, J.C.: Entrepreneurship and growth: the strategic use of external resources. J. Bus. Ventur. **4**, 133–147 (1989). https://doi.org/10.1016/0883-9026(89)90027-X
25. Durst, S., Runar Edvardsson, I.: Knowledge management in SMEs: a literature review. J. Knowl. Manag. **16**, 879–903 (2012). https://doi.org/10.1108/13673271211276173
26. Dollinger, M.J.: Environmental boundary spanning and information processing effects on organizational performance. Acad. Manag. J. **27**, 351–368 (1984). https://doi.org/10.2307/255929
27. Brush, C.G.: Marketplace information scanning activities of new manufacturing ventures. J. Small Bus. Manag. **30**, 41–53 (1992)
28. Brush, C.G., Vanderwerf, P.A.: A comparison of methods and sources for obtaining estimates of new venture performance. J. Bus. Ventur. **7**, 157–170 (1992). https://doi.org/10.1016/0883-9026(92)90010-O
29. Dollinger, M.J.: Environmental contacts and financial performance of the small firm. J. Small Bus. Manag. **23**, 24 (1985)
30. Curran, J., Blackburn, R.: Researching the Small Enterprise. Sage, London (2001)
31. Hessels, J., Parker, S.C.: Constraints, internationalization and growth: a cross-country analysis of European SMEs. J. World Bus. **48**, 137–148 (2013). https://doi.org/10.1016/j.jwb.2012.06.014
32. Abor, J., Quartey, P.: Issues in SME development in Ghana and South Africa. Int. Res. J. Financ. Econ. **39**, 218–228 (2010). ISSN 1450-2887
33. Mambula, C.: Perceptions of SME growth constraints in Nigeria. J. Small Bus. Manag. **40**, 58–65 (2002). https://doi.org/10.1111/1540-627X.00039
34. Cook, C., Cook, M.: The Convergence of Knowledge Management and Business Intelligence. Auerbach Publications, New York (2000)

35. Wong, K.Y., Aspinwall, E.: Characterizing knowledge management in the small business environment. J. Knowl. Manag. **8**, 44–61 (2004). https://doi.org/10.1108/13673270410541033
36. Egbu, C.O., Hari, S., Renukappa, S.H.: Knowledge management for sustainable competitiveness in small and medium surveying practices. Struct. Surv. **23**, 7–21 (2005). https://doi.org/10.1108/02630800510586871
37. Wickert, A., Herschel, R.: Knowledge-management issues for smaller businesses. J. Knowl. Manag. **5**, 329–337 (2001). https://doi.org/10.1108/13673270110411751
38. Wong, W.L.P., Radcliffe, D.F.: The tacit nature of design knowledge. Technol. Anal. Strateg. Manag. **12**, 493–512 (2000). https://doi.org/10.1080/713698497
39. Desouza, K.C., Awazu, Y.: Knowledge management at SMEs: five peculiarities. J. Knowl. Manag. **10**, 32–43 (2006). https://doi.org/10.1108/13673270610650085
40. McAdam, R., Reid, R.: SME and large organisation perceptions of knowledge management: comparisons and contrasts. J. Knowl. Manag. **5**, 231–241 (2001). https://doi.org/10.1108/13673270110400870
41. Kimpeler, S.: What is knowledge management in theory and practice? In: Proceedings of the Baltic-Net Conference on Knowledge Management in Networks and Innovation Systems in Regions in Transition (2001)
42. Razak, N.A., Rashid, W.E.W., Ma'amor, H., Asnawi, N.H., Ahmad, N.L., Achim, N.: Leveraging knowledge transfer in strategic human resource management. Int. J. Trade Econ. Financ. **4**, 168–172 (2013). https://doi.org/10.7763/ijtef.2013.v4.279
43. Penzer, E.: Big ideas come in small packages. Incentive **165**, 34–39 (1991)
44. Grant, R.M.: Toward a knowledge based theory of the firm. Strateg. Manag. J. **17**, 109–122 (1996). https://doi.org/10.2307/2486994
45. Joe, C., Yoong, P., Patel, K.: Knowledge loss when older experts leave knowledge-intensive organisations. J. Knowl. Manag. **17**, 913–927 (2013). https://doi.org/10.1108/JKM-04-2013-0137
46. Hargadon, A., Sutton, R.I.: Building an innovation factory. Harv. Bus. Rev. **78**, 157–166 (2000). 217 p.
47. Shen, W., Cannella, A.A.: Will succession planning increase shareholder wealth? Evidence from investor reactions to relay CEO successions. Strateg. Manag. J. **24**, 191–198 (2003). https://doi.org/10.1002/smj.280
48. Ip, B., Jacobs, G.: Business succession planning: a review of the evidence. J. Small Bus. Enterp. Dev. **13**, 326–350 (2006). https://doi.org/10.1108/14626000610680235
49. Rothwell, W.J.: Putting success into your succession planning. J. Bus. Strategy **23**, 32–37 (2003). https://doi.org/10.1109/EMR.2003.1207057
50. Motwani, J., Levenburg, N.M., Schwarz, T.V., Blankson, C.: Succession planning in SMEs: an empirical analysis. Int. Small Bus. J. **24**, 471–495 (2006). https://doi.org/10.1177/0266242606067270
51. Wang, Y., Watkins, D., Harris, N., Spicer, K.: The relationship between succession issues and business performance: evidence from UK family SMEs. Int. J. Entrep. Behav. Res. **10**, 59–84 (2004). https://doi.org/10.1108/13552550410521380
52. Obadan, J.A., Ohiorenoya, J.O.: Succession planning in small business enterprises in Edo State of Nigeria. https://eujournal.org/index.php/esj/article/viewFile/2048/1959
53. Sambrook, S.: Exploring succession planning in small, growing firms. J. Small Bus. Enterp. Dev. **12**, 579–594 (2005)
54. Awazu, Y.: Managing technology alliances: the case for knowledge management. Int. J. Inf. Manag. **26**, 484–493 (2006). https://doi.org/10.1016/j.ijinfomgt.2006.07.005
55. Cohen, S., Kaimenakis, N.: Intellectual capital and corporate performance in knowledge-intensive SMEs. Learn. Organ. **14**, 241–262 (2007). https://doi.org/10.1108/09696470710739417

56. Evangelista, P., Esposito, E., Lauro, V., Raffa, M.: The adoption of knowledge management systems in small firms. Electron. J. Knowl. Manag. **8**, 33–42 (2010)
57. Simonin, B.L.: Ambiguity and the process of knowledge transfer in strategic alliances. Strateg. Manag. J. **20**, 595–623 (1999). https://doi.org/10.1002/(sici)1097-0266(199907)20:7<595::aid-smj47>3.3.co;2-x
58. Wee, J.C.N., Chua, A.Y.K.: The peculiarities of knowledge management processes in SMEs: the case of Singapore. J. Knowl. Manag. **17**, 958–972 (2013). https://doi.org/10.1108/jkm-04-2013-0163
59. Durst, S., Wilhelm, S.: Knowledge management and succession planning in SMEs. J. Knowl. Manag. **16**, 637–649 (2012). https://doi.org/10.1108/13673271211246194
60. Wong, K.Y., Aspinwall, E.: A fundamental framework for knowledge management implementation in SMEs. J. Inf. Knowl. Manag. **3**, 155–166 (2004). https://doi.org/10.1142/s0219649204000766
61. Nonaka, I., Toyama, R.: The knowledge-creating theory revisited: knowledge creation as a synthesizing process. Knowl. Manag. Res. Pract. **1**, 2–10 (2003). https://doi.org/10.1057/palgrave.kmrp.8500001
62. Nonaka, I.: The knowledge-creating company. Harv. Bus. Rev. **85** (2007). https://doi.org/10.1016/b978-0-7506-7009-8.50016-1
63. Nonaka, I., Konno, N.: The concept of “Ba” (1998). http://home.business.utah.edu/actme/7410/Nonaka%201998.pdf
64. Hoe, S.L.: Tacit knowledge, Nonaka and Takeuchi SECI model, and informal knowledge processes. Int. J. Organ. Theory Behav. **9**, 490–502 (2006)
65. Bontis, N.: Managing organisational knowledge by diagnosing intellectual capital: framing and advancing the state of the field. Int. J. Technol. Manag. **18**, 433 (1999). https://doi.org/10.1504/IJTM.1999.002780
66. Baptista Nunes, M., Annansingh, F., Eaglestone, B., Wakefield, R.: Knowledge management issues in knowledge-intensive SMEs. J. Doc. **62**, 101–119 (2006). https://doi.org/10.1108/00220410610642075
67. Jelavic, M., Ogilvie, K.: Cultural perspectives on knowledge management in central and eastern Europe: the SECI model of knowledge conversion and ‘Ba’. J. Inf. Knowl. Manag. **09**, 161–169 (2010). https://doi.org/10.1142/S0219649210002607
68. Lee, S.M., Hong, S.: An enterprise-wide knowledge management system infrastructure. Ind. Manag. Data Syst. **102**, 17–25 (2002). https://doi.org/10.1108/02635570210414622
69. Davenport, T.H., De Long, D.W., Beers, M.C.: Successful knowledge management projects. Sloan Manag. Rev. **39**, 43–57 (1998). https://doi.org/10.1016/j.ygeno.2009.01.004
70. DiPasquale, J., McInerney, C.R.: Knowledge management in small- and medium-sized enterprises. J. Inf. Knowl. Manag. **09**, 341–353 (2010). https://doi.org/10.1142/S0219649210002723
71. Swap, W., Leonard, D., Shields, M., Abrams, L.: Using mentoring and storytelling to transfer knowledge in the workplace. J. Manag. Inf. Syst. **18**, 95–114 (2001). https://doi.org/10.1080/07421222.2001.11045668
72. Johannson, L.: The challenge of implementing ISO 14001 for small- and medium-sized enterprises—Surviving in the new global jungle. Environ. Qual. Manag. **7**, 9–19 (1997). https://doi.org/10.1002/tqem.3310070203
73. Omar Sharifuddin Syed-Ikhsan, S., Rowland, F.: Knowledge management in a public organization: a study on the relationship between organizational elements and the performance of knowledge transfer. J. Knowl. Manag. **8**, 95–111 (2004). https://doi.org/10.1108/13673270410529145

A New Distributed Brute-Force Password Cracking Technique

Emanuel Tirado, Brendan Turpin, Cody Beltz, Phillip Roshon, Rylin Judge, and Kanwal Gagneja(✉)

Computer Science and Information Technology, Florida Polytechnic University, Lakeland, FL, USA
kgagneja@floridapoly.edu

Abstract. In earlier incarnations of computing, security was not yet a concern. Now, because of the distribution of knowledge, malicious individuals have cleverly learned to take advantage of the loopholes. White hat hackers must constantly stay one step ahead of their black hat counterparts. When all other avenues of password cracking fail, brute force is the only option. Since the advent of secure hashing algorithms, passwords continue to become increasingly more difficult to crack. In this paper, we have presented an algorithm to crack passwords with brute force technique using parallel distribution. It is implemented on an IBM super computer to implement parallel distribution. It is a fact that with a distributed approach, diffusing intense computations across multiple nodes, millions of computations can be processed in a fraction of time.

Keywords: Distributed brute-force · MPI · Password cracking
Windows security · NTLM hash security

1 Introduction

In the Windows Operating System, arguably the most prevalent end-user platform, passwords are stored locally in the form of a hash value. When users set their passwords, the MD4 one-way hash function is responsible for converting the plain text into the hash. By using a brute-force approach, the hash value can be mapped back to the original plain text. The problem with this approach is that the number of combinations is very large.

According to the InfoSec Institute, the average password length is between 8 and 9 characters long. Assuming that a password only uses a combination of lower/upper case letters and numbers, the number of possible combinations can be in the order of 62^(8 or 9). This number can be even greater if you consider the addition of other ASCII symbols (e.g. !, @, #, %). The number of combinations can be reduced through the use of heuristics. A common approach is to only use words in a dictionary, or to only use syllables found in the English language.

1.1 Motivation

The motivation for this paper was primarily to analyze how distributed computing could affect the field of cyber security. With a dynamically distributed implementation,

R. Doss et al. (Eds.): FNSS 2018, CCIS 878, pp. 117–127, 2018.
https://doi.org/10.1007/978-3-319-94421-0_9

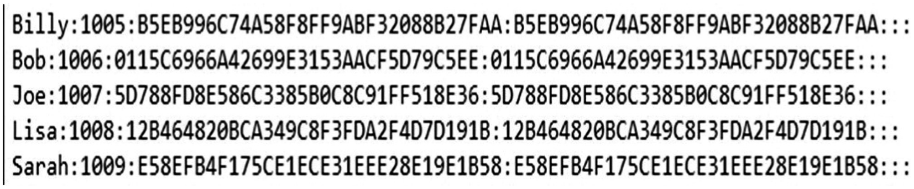
the time to crack a password could be drastically diminished, meaning that users with weak passwords are at a great risk, regardless of how securely their passwords are stored. We also wanted to stress the importance of robust encryption algorithms. Some modern encryption algorithms scale their intensity based on computing performance, forcing the hash function to take the same amount of time to process regardless of system performance. With local Windows account passwords, this is obviously not the case; we want users to be aware of the mechanisms that secure their data.

2 Windows Account Security

Microsoft has been using the same hashing algorithm since Windows 2000 for backwards compatibility purposes. These versions of Windows use the NTLM (NT LAN Manager) hashing algorithm to hash their passwords and are without the salt. In order to crack these passwords, we have implemented a distributed brute-force algorithm to check each permutation efficiently. In this section, we discuss the functions required to achieve our goal.

2.1 NTLM Hash Extraction

To crack the password, the program requires the hashed version of the password that resides in the Windows Security Account Management (SAM) file [1] as shown in Fig. 1. This hashed password can be obtained either by using a pre-existing tool that extracts the file while in the windows environment (e.g. an open source algorithm [2]), or from a bootable version of linux inside a USB flash drive that can access the file system directly. This file is colon delimited and contains the username, a unique identifier, an LM hash of the password, and an NTLM hash of the password.



```
Billy:1005:B5EB996C74A58F8FF9ABF32088B27FAA:B5EB996C74A58F8FF9ABF32088B27FAA:::
Bob:1006:0115C6966A42699E3153AACF5D79C5EE:0115C6966A42699E3153AACF5D79C5EE:::
Joe:1007:5D788FD8E586C3385B0C8C91FF518E36:5D788FD8E586C3385B0C8C91FF518E36:::
Lisa:1008:12B464820BCA349C8F3FDA2F4D7D191B:12B464820BCA349C8F3FDA2F4D7D191B:::
Sarah:1009:E58EFB4F175CE1ECE31EEE28E19E1B58:E58EFB4F175CE1ECE31EEE28E19E1B58:::
```

Fig. 1. Sample of dumped SAM file

3 Brute-Force Engine

Our solution consists of a distributed password cracking program built using MPI (XXXX). The brute-force engine is responsible for generating every possible permutation and comparing its hashed value to the extracted value.

3.1 Calculating Total Permutations

The first piece of information needed for our program is the total number of "guesses" the program will attempt. To do so, we wrote a function that takes into account several attributes, such as the minimum length (min), the maximum length (max), and the set

(x) of allowed characters (a–z, A–Z, 0–9). To determine the number of permutations (P), the following equation is used:

$$P(min, \max) = \sum\nolimits_{i=min}^{max} x^i$$

Where x = char set length

3.2 Enumerating Permutations

Each possible combination of characters is matched to a corresponding index. This approach will allow the program to be able to distribute the work across several computing nodes. We use an array of integers to represent every possible combination. Each item in the array corresponds to the index of the given character set. To illustrate:

```
0  --> {0, 0, 0} = "aaa"
1  --> {0, 0, 1} = "aab"
2  --> {0, 0, 2} = "aac"
...
62 --> {61, 61, 60}  = "998"
63 --> {61, 61, 61}  = "999"
```

3.3 Distribution Algorithm

The program needs to divide the work between nodes as evenly as possible. If the number of permutations is not evenly divisible by the number of nodes, nodes are given an additional permutation until there are no more "remainders". In the following example, we have a total of 1003 possible permutations, with our function, the work would be divided as follows:

```
- Node#0 --> start: 0, end: 99
- Node#1 --> start: 100, end: 199
- Node#2 --> start: 200, end: 299
- Node#3 --> start: 300, end: 399
- Node#4 --> start: 400, end: 499
...
- Node#9 --> start: 900, end: 1003
```

3.4 Brute-Force Algorithm

Initially, we distributed the entire set of permutations up front, but this approach proved to be inefficient for shorter length passwords. For example, if we were cracking a password from 3–8 characters in length, the function did not prioritize checking shorter character passwords before entering the next length level. In order to optimize this method, we now segment the distribution in stages based on length.

```
    //Main function of Brute-Force Algorithm
    int main (int argc, char* argv[])
    {
      struct BF_Arguments Arguments; int Rank_id;
      int    Count_of_workers;    MPI_Status    Status;
   MPI_Init(&argc, &argv);
      MPI_Comm_size(MPI_COMM_WORLD, &Count_of_workers); //Retrieve # of workers
 MPI_Comm_rank(MPI_COMM_WORLD, &Rank_id); //Retrieve rank id
      //Default argument values
      Arguments.Min_length = 5;
      Arguments.Max_length = 5;
       Arguments.Charset ="abcdefghijklmnopqrstuvwxyz-
   ABCDEFGHIJKLMNOPQRSTUVWXYZ 0123456789";
    //Parsing arguments
      argp_parse(&Arg_parser, argc, argv, 0, 0, &Arguments); Min_length = Argu-
   ments.Min_length;
      Max_length = Arguments.Max_length; Charset = Arguments.Charset;
   Extracted_hash = Arguments.Extracted_hash;
      //Argument validation
      if (strlen(Extracted_hash) != 32) {
        printf("Invalid                    NTLM                    hash.\n");
     MPI_Abort(MPI_COMM_WORLD, 1); }
      //Let  the  master  worker  do  its  thing...  if  (Rank_id  ==
   BF_MASTER)
      {
        BF_Master(Count_of_workers, &Status);
      }
      //Make the worker work BF_Worker(Rank_id, &Status);
      //Wait          for          everyone          to          finish
   MPI_Barrier(MPI_COMM_WORLD);
      //Fin. MPI_Finalize(); return 0;
    }
    static void BF_Master(int Count_of_workers, MPI_Status *Status)
    {
      int Dest_rank_id;
   uint64_t    Total_permutations    =    Get_total_permutations(strlen(Charset),
 Min_length, Max_length);

   uint64_t *Last_indices = Divide_workload(Total_permutations, Count_of_workers);
      uint64_t Start_index = 0;


      //Sending work
   for (Dest_rank_id = 0; Dest_rank_id < Count_of_workers; Dest_rank_id++)
      {
     uint64_t End_index = Last_indices[Dest_rank_id]; printf("Processor %d has
 been assigned permutations %llu through %llu. \n", Dest_rank_id, Start_index,
 End_index);
```

```
    MPI_Send(&Start_index,    1,    MPI_LONG_LONG_INT,    Dest_rank_id,
BF_TAG_FIRST, MPI_COMM_WORLD);
    MPI_Send(&End_index,    1,    MPI_LONG_LONG_INT,    Dest_rank_id,
BF_TAG_LAST, MPI_COMM_WORLD);


      Start_index = End_index + 1;
    }
  }
```

4 User Interface

To increase the utility of the password cracker, we have created a Graphical User Interface (GUI) to assist in making the process more intuitive as shown in Fig. 3. The user can parse the Windows SAM file, select the account to be cracked, and initiate the password cracker with the click of a button as shown in Fig. 2. The GUI application will check for the completion of this job, parse the cracked password from the output file, and display the password in plain text to the user.

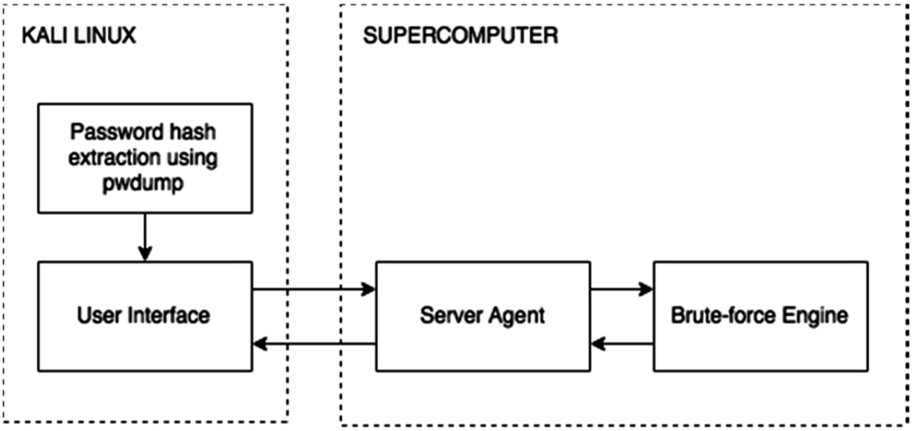


Fig. 2. Communication diagram

In order to effectively implement this GUI, we created a few different features. The GUI itself was created using the Java Swing library and interacts with a server hosted on the HPC using a client-server architecture as shown in Fig. 2. The client sends the server the parameters of the operation and the server then schedules an MPI job based on the provided parameters. When the job completes, the server will return the output to the client, which will then be displayed to the user on screen.

5 MPI Challenges and Optimizations

Brute-forcing a password can be seen as a needle-in-the- haystack type problem. That is, as soon as one of the worker nodes finds a solution, there is no need to continue further. This is unlike Map-Reduce patterns of problem solving where the work is distributed and then aggregated once finished.

One major challenge we faced was implementing a shared flag that signals to the other workers that it is time to stop. There were several ways we considered tackling this problem and ultimately arrived at a solution that worked for us.

First, we looked into using the MPI broadcast function, MPI_Bcast. With this function, a worker designated as the root can pass on information to the rest of the workers. For this function to work, every worker calls the same Bcast function. The following Fig. 3 illustrates of how MPI_Bcast works:

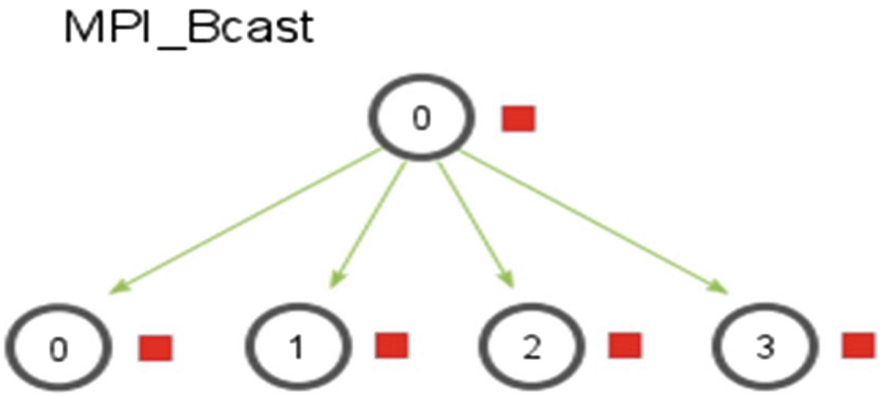


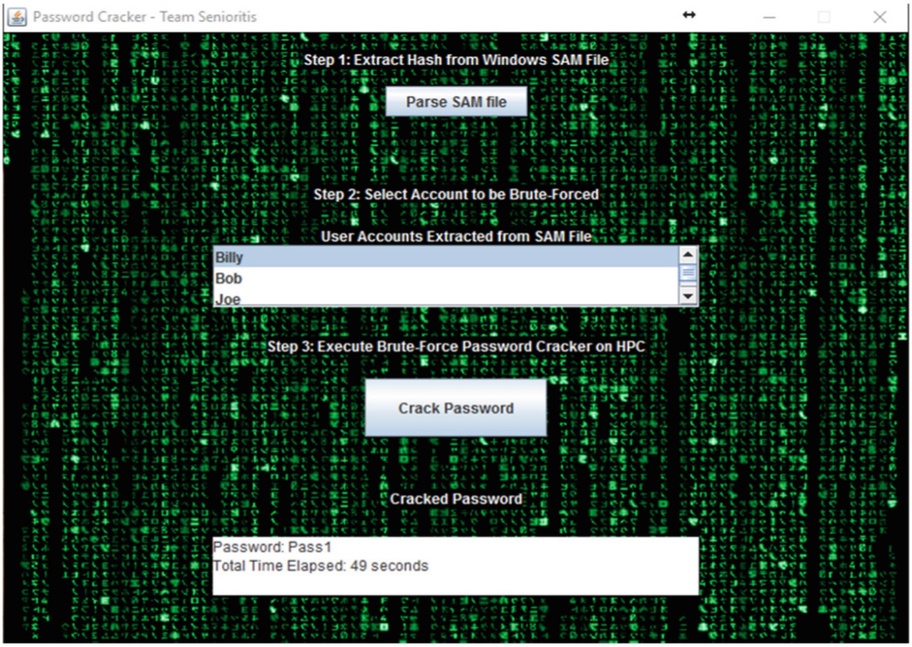
Fig. 3. MPI_Bcast

For our purposes, we needed to have the ability for any worker to become the broadcaster. This cannot be accomplished using MPI_Bcast alone (Fig. 4).

Our solution to this problem was to use MPI_Gather in conjunction with MPI_Bcast. MPI_Gather is used to collect information from all workers as shown in Fig. 5 below. Each worker calls MPI_Gather during every iteration and let the master node know if a solution was found. The master evaluates the results and then broadcasts a signal to inform the other workers if they should stop or continue.

This approach solves the problem, but it also creates a performance issue. Both MPI_Bcast and MPI_Gather are blocking operations. That is, the processes are put on hold until the communication operations are completed. This directly impacts our hashing rate which is critical to the application.

A more succinct approach is the MPI_Allgather function. This function allows workers to exchange information with one another as shown in Fig. 6. After calling MPI_Allgather, each worker checks the responses from its peers and stops if a solution was found by any of their peers.


Password Cracker - Team Senioritis
Step 1: Extract Hash from Windows SAM File
Parse SAM file
Step 2: Select Account to be Brute-Forced
User Accounts Extracted from SAM File
Billy
Bob
Joe
Step 3: Execute Brute-Force Password Cracker on HPC
Crack Password
Cracked Password

```
Password: Pass1
Total Time Elapsed: 49 seconds
```

Fig. 4. User interface

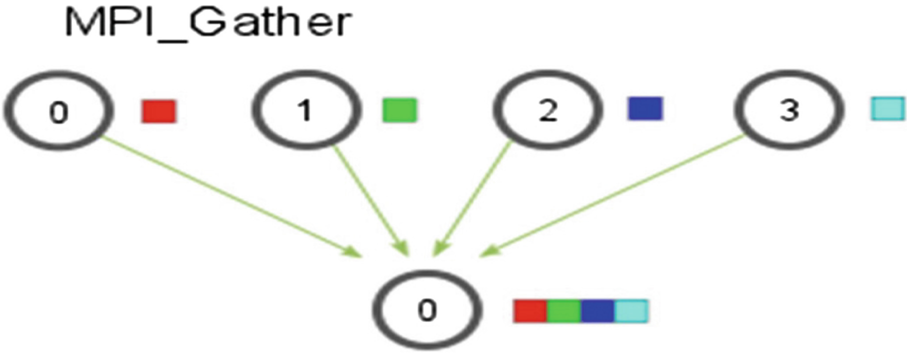


Fig. 5. MPI_Gather

This approach eliminates the need for the MPI_Bcast function, reducing the communication overhead significantly. Despite this improvement, the hashing rate continued to be severely impacted.

A possible workaround was to only perform the MPI_Allgather operation periodically (e.g. for instance, every million hashes). This minimizes the communication overhead drastically while only wasting a few seconds of CPU time.

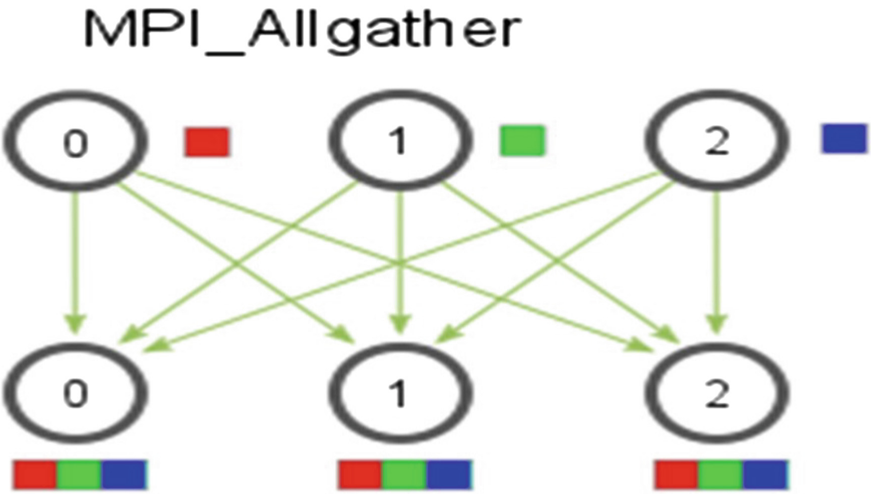


Fig. 6. MPI_Allgather

Finally, a fourth approach emerged, it consists of making a call to MPI_Abort as soon as a solution is found by one of the workers. This method is fast and eliminates the need to have any communication between the workers after the work distribution stage. A small experiment was executed comparing these different approaches and the results show that the MPI_Abort method is the fastest method, but not significantly better than the method that utilizes MPI_Allgather with periodic checks.

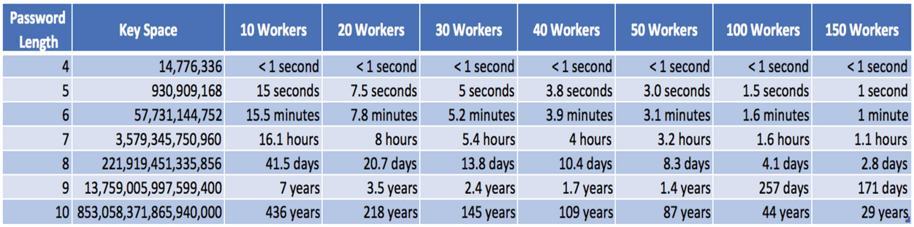
In a later revision of the code, we switched to MPI_Allreduce. This function works similarly to MPI_Allgather, but instead it performs a sum operation to count the number of solutions found by all nodes. The workers stop when this number is greater than zero. No significant performance impact was observed.

6 Results and Analysis

The “Hash Comparison Rate” is defined as the number of hash comparison completed in a second. After running 10 trials, we determined that the average hash second per computing node is 6,194,329 hash comparisons per second. Table 1 contains the time estimates for cracking passwords of a particular length. An experiment was performed using four MPI workers trying to crack a five digit password (9999). To find the result, the application must calculate 916,132,831 hashes.

We were able to crack a 5 character password in about 8 s. For comparison, we attempted to crack the same password on a professional password cracking software Cain & Abel [1]. Cain was able to crack this password in 1 min and 23 s. This demonstration successfully proves the need for stronger encryption and password rules in modern applications. As computing professionals, it is our ethical duty to ensure user information is kept confidential.

Table 1. Time estimates for cracking passwords of a particular length



Password Length	Key Space	10 Workers	20 Workers	30 Workers	40 Workers	50 Workers	100 Workers	150 Workers
4	14,776,336	< 1 second	< 1 second	< 1 second	< 1 second	< 1 second	< 1 second	< 1 second
5	930,909,168	15 seconds	7.5 seconds	5 seconds	3.8 seconds	3.0 seconds	1.5 seconds	1 second
6	57,731,144,752	15.5 minutes	7.8 minutes	5.2 minutes	3.9 minutes	3.1 minutes	1.6 minutes	1 minute
7	3,579,345,750,960	16.1 hours	8 hours	5.4 hours	4 hours	3.2 hours	1.6 hours	1.1 hours
8	221,919,451,335,856	41.5 days	20.7 days	13.8 days	10.4 days	8.3 days	4.1 days	2.8 days
9	13,759,005,997,599,400	7 years	3.5 years	2.4 years	1.7 years	1.4 years	257 days	171 days
10	853,058,371,865,940,000	436 years	218 years	145 years	109 years	87 years	44 years	29 years

The following Table 2 presents the time taken to execute and the number of hashes performed in unit time for all the developed approaches. It is different for all the approaches.

Table 2. Time taken to execute the developed approaches

Approach	Total time	Hashes/second
MPI Gather/broadcast	6 min 23 s	2.4 million/s
MPI Allgather	4 min 40 s	3.3 million/s
MPI Allgather (w/periodic checks)	59 s	15.5 million/s
MPI Abort	57 s	16 million/s

Choosing a hashing algorithm that offers security against brute-force attacks is another important aspect to consider. For a brute-force attack to be effective, the hashing algorithm should allow millions of comparisons every second. A secure hashing algorithm will be "slow" by design to counteract this. For a user, if the authentication process takes an extra 10 ms, it would be near impossible to detect, but for an attacker this means that the number of hashes that can be computer per second are limited severely.

The National Institute of Standards and Technology (NIST) provides a Cryptographic Toolkit in which they define what constitutes a secure hashing algorithm. There are two publications in which they define three different families of secure hashing algorithms. Publication FIPS 180-4 specifies seven hash algorithms: SHA-1 (Secure Hash Algorithm-1), and the SHA-2 family of hash algorithms: SHA-224, SHA- 256, SHA-384, SHA-512, SHA-512/224, and SHA-512/256.

In a more recent publication, FIPS 202, they introduce the new SHA-3 family of permutation-based functions. This publication specifies four fixed-length hash algorithms: SHA3- 224, SHA3-256, SHA3-384, and SHA3-512; and two closely related, "extendable-output" functions (XOFs): SHAKE128 and SHAKE256. In a newsletter published in March of 2016, the NIST recommend that for Federal Agencies to stop using SHA-1. There are other algorithms that help prevents GPU brute-force attacks by using operations that are not typically supported by GPUs.

Another factor to consider is that attackers are constantly looking for weaknesses in these algorithms. What we consider to be safe today, will most than likely change in the near future. The best approach is to stay informed.

7 Future Work

Cracking a password greater than 7 characters is extremely time consuming. For this reason, we feel a GPU implementation of this would yield much greater results. The SIMD (Single Instruction, Multiple Data) architecture of a GPU is designed to handle a large amount of identical, trivial operations, producing higher throughput. CPU's are not optimized for this form of calculation, therefore consuming more power than necessary; a GPU cluster would be more efficient all-around.

Another potential improvement would be to randomize the permutation assignments. This way, permutations on the higher end of each node's assigned permutation range would not consistently perform as "worst-case scenario". We would require additional statistical evidence to back up the claim that this would improve average cracking time, but believe there would be at least a slight improvement.

While brute-force password cracking is required when the password is sufficiently secure, there are other preferred methods of password cracking available. Dictionary attacks and rainbows tables are much quicker methods, yet do not guarantee success. If we were to integrate these methods into our application, utilizing them prior to brute-force, our overall average performance would tremendously increase.

8 Conclusions

In today's world, hearing about database breaches has become a normal occurrence. In recent years, several large corporations like Home Depot, LinkedIn, and Yahoo have had user information leaked. For software developers, mitigating the damage that can be inflicted when a data breach occurs should be paramount. Following a set of best practices can help make it harder for attackers to steal our information. This paper presents a distributed password cracker. While this distributed password cracker is not fully optimal, we are extremely pleased with the results.

References

1. Cain and Abel. http://www.oxid.it/cain.html. Platform MPI User's Guide. IBM. 2012 MPI Communication Diagrams. http://www.mpitutorials.com
2. Du, X., Gagneja, K.K., Nygard, K.: Enhanced routing in heterogeneous sensor networks. In: IEEE Computation World 2009, Athens, Greece, pp. 569–574, 15–20 November 2009
3. Evanoff, L., Hatch, N., Gagneja, K.K.: Home network security: beginner vs advanced. In: ICWN, Las Vegas, USA, 27–30 July 2015
4. Gagneja, K.K., Nygard, K.: Heuristic clustering with secured routing in heterogeneous sensor networks. In: IEEE SECON, New Orleans, USA, pp. 9–16, 24–26 June 2013

5. Gagneja, K.K.: Secure communication scheme for wireless sensor networks to maintain anonymity. In: IEEE ICNC, Anaheim, California, USA, 16–19 February 2015
6. Gagneja, K.K.: Pairwise post deployment key management scheme for heterogeneous sensor networks. In: 13th IEEE WoWMoM 2012, San Francisco, California, USA, pp. 1–2, 25–28 June 2012
7. Gagneja, K.K.: Knowing the ransomware and building defense against it - specific to healthcare institutes. In: IEEE MobiSecServ, Miami, USA, 11–12 February 2017
8. Kanwal, G.: Pairwise key distribution scheme for two-tier sensor networks. In: IEEE ICNC, Honolulu, Hawaii, USA, pp. 1081–1086, 3–6 February 2014
9. Nygard, K., Gagneja, K.: Energy efficient approach with integrated key management scheme for wireless sensor networks. In: ACM MOBIHOC, Bangalore, India, pp. 13–18, 29 July 2013
10. Nygard, K., Gagneja, K.K.: Key management scheme for routing in clustered heterogeneous sensor networks. In: IEEE NTMS 2012, Security Track, Istanbul, Turkey, pp. 1–5, 7–10 May 2012
11. NTLM Algorithm: Openwall Community Wiki. 17 February 2010. Accessed 16 Mar 2017
12. Peleus: Ramblings. NetSec. Accessed 16 Mar 2017
13. Max, R., Gagneja K.K.: Raspberry Pi webserver. In: ESA, Las Vegas, USA, 27–30 July 2015
14. Arvinderpal, S., Gagneja, K.K.: Incident response through behavioral science: an industrial approach. In: IEEE CSCI, LasVegas, USA, 7–9 December 2015

Identifying Drawbacks in Malicious PDF Detectors

Ahmed Falah(✉), Lei Pan, Mohamed Abdelrazek, and Robin Doss

School of Information Technology, Deakin University, Burwood, VIC 3125, Australia
{afalah,l.pan,mohamed.abdelrazek,robin.doss}@deakin.edu.au
https://www.deakin.edu.au

Abstract. Despite the continuous countermeasuring efforts, embedding malware in PDF documents and using it as a malware distribution mechanism is still a threat. This is due to its popularity as a document exchange format, the lack of user awareness of its dangers, as well as its ability to carry and execute malware. Several malicious PDF detection tools have been proposed by the academic community to address the PDF threat. All of which suffer some drawbacks that limit its utility. In this paper, we present the drawbacks of the current state of the art malicious PDF detectors. This was achieved by undertaking a survey of all recent malicious PDF detectors, followed by a comparative evaluation of the available tools. Our results show that Concept drifts is major drawback to the detectors, despite the fact that many detectors use machine learning approaches.

Keywords: Malicious PDF detection · Comparative evaluation
Concept drift

1 Introduction

It has been getting increasingly popular to embed malware in documents. PDF in particular, which is used as an alternate distribution mechanism. This is to counter users' increasing awareness of the dangers of executables and other malware distribution approaches. In comparison, not as many are aware of the capabilities of PDF (and other *seemingly benign file types*) and its ability to carry malicious code. Users are blindly assuming it as a plain document, especially when combined with a little social engineering, to trick the target into ignoring any warning signs that might be detected unconsciously. A security-related psychology study in [14] found that for non-security professionals, the human brain is more likely to pick up (cyber) danger indicators unconsciously than it can consciously.

Figure 1 shows a trend regarding the number of PDF-related reported vulnerabilities (as common vulnerability exposures (CVEs)) each year. 2009 seemed to be the peak for maldocs and exploitable documents, given the high number of reported CVEs. There was a quiet period between 2009 and 2016. But more

R. Doss et al. (Eds.): FNSS 2018, CCIS 878, pp. 128–139, 2018.
https://doi.org/10.1007/978-3-319-94421-0_10

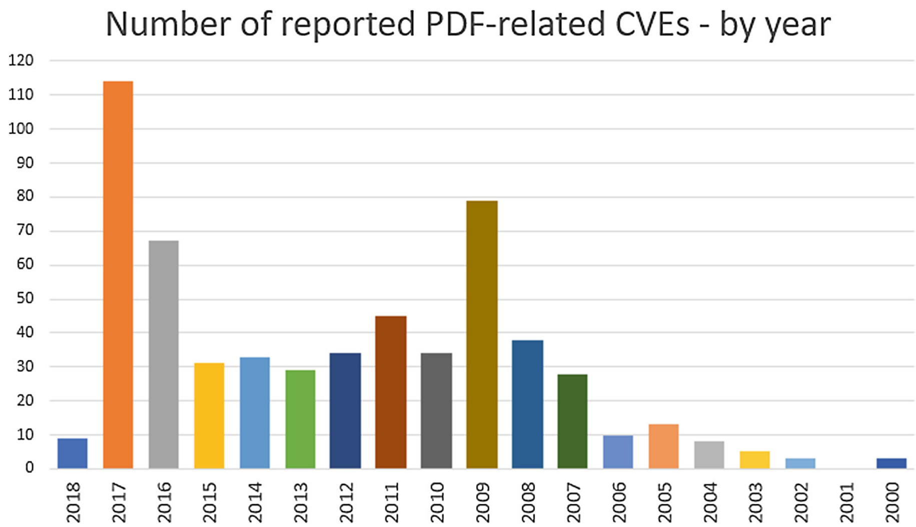

Number of reported PDF-related CVEs - by year

Fig. 1. Number of reported PDF-related CVEs each year. 2016 and 2017 saw an increase in reported CVEs, compared to previous years. Figures collected from [5]

than doubling the number of reported CVEs (31 in 2015, 67 in 2016), which continued in 2017, increasing by nearly 60% with 114 reported CVEs and 8 reported already in the first 2 weeks of 2018. Not only the number of reported CVEs is increasing, but also the severity of these CVEs is increasing, as shown in Fig. 2, according to the NVD [15], where the number of "high" severity increased by over 50% from 2016 to 2017. Figure 2 contains data taken in the last 2 years using the CVSS version 3.0 score. Data of the previous years is still in version 2.0 format. Table 1 shows the NVD scores range for each severity rank.

Besides user awareness, PDF is widely used in business, making it an ideal malware distribution mechanism, because it works across platforms, devices and operating systems, in particular, its ability to execute a wide variety of code, such as JavaScript and ActionScript.

In the September 2017 threat report [12], McAfee stated that malware writers and cyber criminals are moving away from binary and executable into non-executable, script-based malware. This is due to its advantages over

Table 1. NVD's CVE severity score range (CVSS version 3.0)

Ranking	Score
Low	0.1–3.9
Medium	4.0–6.9
High	7.0–8.9
Critical	9.0–10.0

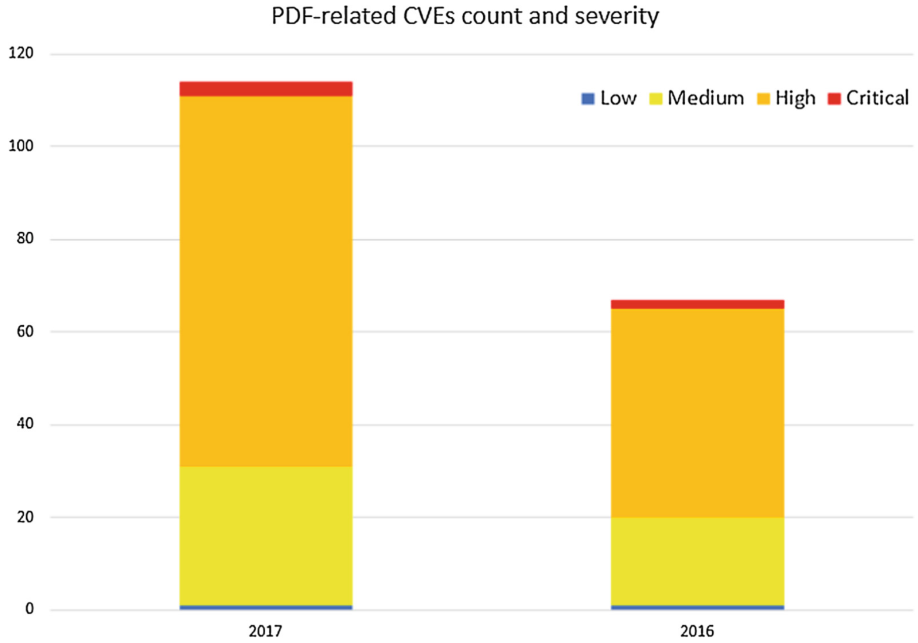


Fig. 2. Severity of PDF-related CVEs in 2016 and 2017. The count of CVEs rated by the NVD as "high" has more than doubled in 2017, compared to 2016. Data obtained from [15]

executables, such as: (1) ease of antivirus evasion, (2) efficiency, (3) easier obfuscation.
Table 2 highlights the current techniques in malicious PDF detection:

1. Static based detection approaches are more utilized than dynamic approaches, specifically, it is used in 4 out of 5 tools reviewed in this work. This focuses on detecting malicious indicators in document structure and metadata, or content (i.e malicious JavaScript). However, there exists a wide range of techniques, such as classifier evasion, parser confusion attacks, to counter static based approaches.
2. Machine learning is used with static based detection (3 out of 4 static based approaches are machine-learning based), and is not utilized in dynamic-based detection approaches.

This is counter intuitive, seeing that recent PDF standard improvements have limited the malicious capabilities of PDF, which leads into the primary installer/dropper role (according to [12]). In this case, a malicious PDF includes a script which can be automatically triggered to download another malware. Such behavior is detected more accurately by adopting dynamic approaches.

Another unexplored area in PDF detection is its utilization in social engineering phishing, where a PDF file does not contain any malicious contents, instead, a file presents a malicious URL to the victim, as well as some message to motivate the victim to click the link. That link leads to a malicious web page that performs a malicious act (such as download malware or steal credentials). In this scenario, a PDF file plays a vital role in the compromise, without actually including any malicious contents.

Furthermore, the current detection approaches are all almost always client- or web-based, where a user is required to manually submit a PDF file for inspection. Submitting a file for inspection requires a considerable amount of effort, and potentially advanced computer skills. This laborous process prevents these tools to be used widely by ordinary users. Situations become worse when there are many files to be scanned regularly. This is the expected case in business environments, where users expect (and need) the security, without the high interaction overhead.

We conducted a set of experiments to identify drawbacks in malicious PDF detection tool. 2 tools where trained with 2 datasets collected over different periods. The first dataset was collected before 2013 (taken from [4]), and the other collected recently and provided by VirusTotal. Our experiments show that concept drifts is a key challenge, where tools trained with data collected in previous years, did not accurately detect the testing dataset. Where detection accuracy decreased from around the 90–100 percentile to the 70–80 percentile. This happened because the testing dataset collected in recent years contains PDF file of other standards, such as the PDF/A standard.

To summarize, the previous section highlighted the following research gaps in the field of malicious PDF detection:

1. The current malicious PDF detection suffer from concept drift and other factors that decrease the detection efficiency and reliability, such as being limited to a specific PDF version or standard of inspected files.
2. The current tools operate at client-level only, and require considerable effort to submit a file for inspection.
3. The tool designers do not consider expected change in PDF distribution mechanism, such as IoT devices (i.e. smart meters) automatically generating reports in PDF (among other formats) and sending them directly to recipients, without going through conventional distribution methods, such as email.

Our contributions in this paper are:

- Conducted a comparative evaluation to identify concept drifts in current state of the art malicious PDF detectors, and other factors that cause drop in performance.
- Identified drawbacks in current state of the art malicious PDF detectors.

2 Literature Review

Between 2000 and 2017, 572[1] PDF-related vulnerabilities were published on the CVE database [5], 114 of which were reported in 2017 alone, and 67 in 2016, that is 31% of the total vulnerabilities in 2 years only. Despite the several malicious PDF detection methods that were proposed dating as back as 2007 spanning over the past 10 years [7–11,16,18,20,22], embedding malicious code within PDF files is becoming increasingly popular among cyber criminals. This is because of its versatility, portability, and wide spread, as well as supporting features that allow malicious code execution. This section will briefly review the features that enable malicious code embedding within PDF documents, then the most recent detection methods will be discussed, and will conclude by discussing social engineering and the role it plays in parallel with malicious PDF.

Enablers: Reviewing the PDF standard [2] shows the rich content allowed in the files, which is part of the reason PDF has become the de-facto file exchange standard in enterprises. The main enabler according to [3,10] is the ability to embed JavaScript within PDF files to perform various tasks, which is notorious for its exploitability, such as these shown at BlackHat [6]. Besides JavaScript, [10] also lists ActionScript as a tool.

Moving away from technical enablers, malicious PDF writers exploit the benign appearance of PDF files. The malicious potential of PDF files is known in the security communities, but non-expert, average users are not aware. Thus the probability of opening a malicious PDF file by a person is much higher than opening files in other formats such executables, regardless of the distribution mechanism (email attachment, USB stick, download) and the presence of anti-malware applications.

Behavior: It is possible to perform sophisticated attacks through PDF, according to [9], such as heap spraying, mapped memory search and DLL injection. [1] shows the reader application is frequently updated and patched, and exploitable embedding formats are blacklisted. These security updates are driving PDF utilization into one of the following malicious roles: (1) The dropper role where the PDF file will download and install a malware from the Internet. (2) Leverage in social engineering attacks, such as including a malicious URL and tricking the user into clicking it, in a phishing-like approach.

Detection: Table 2 summarizes the detection tools reviewed in this section. The table shows that the static approach is more preferred than dynamic approaches. This is because of its speed, efficiency, and low overhead. In comparison, dynamic approaches are slower and more expensive, but could be more accurate. The table also shows that machine learning is utilized in static approaches only, but dynamic approaches do not rely on machine learning detection. Figure 3

[1] Search was conducted using the "PDF" keyword only. [22] reports much higher numbers using assumably the "adobe acrobat reader" keyword.

Table 2. Summary of the reviewed maldocs detectors.

Tool	Year	Method	Focus	ML
PDFrate [16]	2012	Static	Structure[a]	Yes
Unnamed [9]	2014	Both	JavaScript	No
Slayer [10]	2015	Static	Content & structure	Yes
Hidost [19]	2016	Static	Structure	Yes
PlatPal [22]	2017	Dynamic	Behaviour differences	No

[a] Document structure and document metadata are used interchangeably by various works

provides a high level overview of all operations that take place in malicious PDF detection. When a static approach is taken, a detection tool will look at the various tags used in a file, then a classifier makes a decision. Some tools parse JavaScript and review the content of each tag, rather than making a decision based on the tags only. When we take a dynamic approach, the behavior of a file is monitored before a decision is made. Below is a review of the most recent academic detectors.

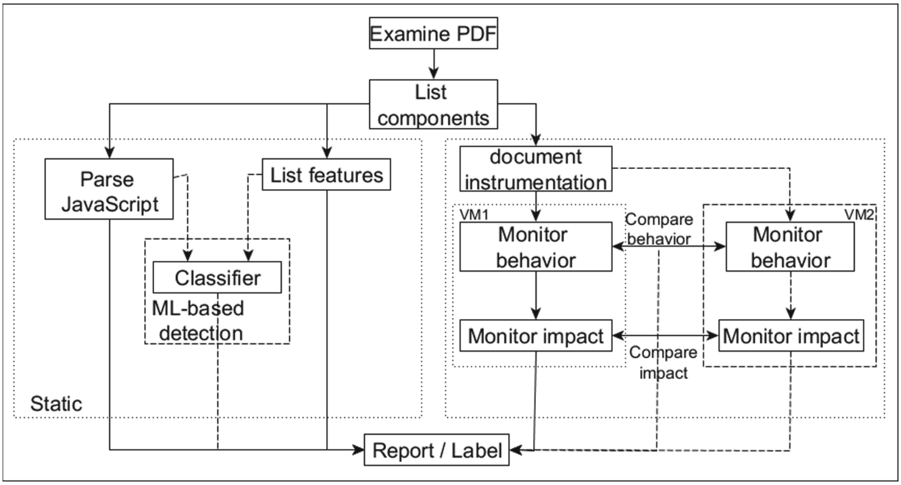


Fig. 3. High level overview of the malicious PDF detection process.

PDFrate [16] examines over 200 features extracted from document structure and meta data and utilizes random forests to binary classify PDF files. It then classifies malicious documents as opportunistic (relies on mass distribution) or targeted (targets specific individuals or organization, utilizing social engineering to lure the victim into interacting with the document). To counter mimicry attacks, the authors suggest removing the top feature that enable such attacks, resulting in negligible classification errors, a problem that was later addressed in [17]. [18] takes a similar approach to [16], basing their detection on differences in the structure between malicious and benign files. The work was later on improved in [19], where the classifier examines the logical structure and the file content, as well as extending it to cover multiple hierarchical file formats, such as Adobe Flash.

Differently, by focusing on content instead of structure, [9] proposes a content-aware detection approach. Utilizing document instrumentation to monitor JavaScript execution at run-time for certain behaviors such as malware dropping, suspicious memory consumption, suspicious network access, and process creation. The evaluation dataset used provide insight on the trends on malware writers, where every single malicious file out of the 7370 used, contains JavaScript, which justifies focusing exclusively on detecting malicious JavaScript behavior in the work.

[10] highlights the drawbacks of the previous works [9,16], where a structure only detection approach is susceptible to manipulation and mimicry attacks, and a JavaScript only detection is incapable of detecting any non-JavaScript malicious content. To address this, the authors build upon the previous two works, proposing a system that extracts both content- and structure-based information in order to build a classifier that leverages adaptive boosting decision trees and over 100 features. The authors reported resilience of their classifier against three types of attacks: JavaScript injection, EXE- and PDF embedding, making no mention of resilience to mimicry attacks.

Detectors that rely on JavaScript extraction are vulnerable to a new class of attacks introduced by [3], called Parser confusion attack. The attack exploits the weaknesses of the current JavaScript parser, which includes implementation bugs, designers errors, omissions and ambiguities. The attack attempts to hide the malicious payload embedded within a PDF file by encoding and obfuscating the objects, malicious JavaScript and reference. The authors suggest three mitigation technique for the proposed attack: (1) exploit detection at runtime (2) improving JavaScript parsers, and (3) deployment of the proposed reference extractor. Besides these mitigation suggestions, [22] goes a different direction to address this attack class, as well as other techniques. The authors propose a detection technique based on platform diversity. They assume that benign PDF files will behave similarly on different platforms, while malicious files, especially targeted attacks, are designed to attack a specific platform, therefore showing several behavioral differences when examined on non-targeted platform. This behavior difference stems from differences in how various platform handle various aspects, including (1) system call semantics (2) calling convention and argument passing (3) library dependencies (4) memory layout (5) heap management (6) executable formant (7) filesystem semantics (8) expected programs on the target platform, according to [22]. Exploiting specific vulnerabilities requires tailor-made input that utilizes some or all of the factors listed above, which leads to failed, or behaviorally different, execution on various platform.

3 Experiment

The aim of this experiment is to identify concept drift that occurs due to the aging of classifiers. This occurs when new malware samples utilize new techniques that were not examined during the training of the classifiers, resulting in questionable outputs and unreliable classification.

The experiment evaluated the performance of 2 of the tools explained in Sect. 2: PDFrate and Slayer. Both employ static detection approaches, utilization machine learning.

3.1 Datasets

In preparation for the experiment, malicious and benign PDF files were collected from the following sources:

1. Contagio: a large, publicly available dataset that dates back to 2013, among the rest, this dataset contains 9000 benign PDF files, and over 10,000 malicious PDFs. Used for training and evaluation.
2. VirusTotal: provided 10,500 malicious PDF files. Used for training and evaluation.
3. TPN: contains 1000 open-source PDF files. Used mainly for training.
4. Personal: this dataset was used for evaluation, and was collected from personal files, as well as Google searches.

To perform the experiment, the datasets explained above were divided into several sub-datasets for training and evaluation. Table 3 summarizes the datasets used.

Training: 900 benign and 900 malicious files were used from several dataset for training each instance of Slayer.

Evaluation: 10 × 100 benign and malicious files were used from several datasets to evaluate Slayer and PDFrate.

Table 3. Datasets used in the pilot experiment.

#	Source	Label	Status	Files	Purpose
1	Contagio	Benign	Old	900	Training
2	Contagio	Malicious	Old	900	
3	TPN	Benign	Recent	900	
4	VirusTotal	Malicious	Recent	900	
5	Contagio	Benign	Old	1000	Evaluation
6	Contagio	Malicious	Old	1000	
7	TPN	Benign	Recent	100	
8	VirusTotal	Malicious	Recent	1000	
9	Personal	Benign	Recent	900	

3.2 Procedure

To identify concept drift, 2 instances of slayer were trained. The first one was trained with recently collected data (called Slayer2017 henceforth). The second was trained with an old dataset, called slayer2013. Each of these classifiers was trained as follows:

- **Slayer2017:** Trained with 900 benign and 900 malicious files from the VirusTotal, TPN and Personal datasets.
- **Slayer2013:** Trained with 900 benign and malicious files from the Contagio dataset.

PDFrate does not need training as it is an online tool, and utilize 3 classifiers, trained with several datasets, according to the author and creator, as follows:

- Classifier trained with the Contagio dataset, called PDFrate (Contagio) in this experiment, trained with 10,000 benign and malicious files.
- Classifier trained with data collected from the network of the George Mason university, trained with 100,000 benign and malicious files, called PDF (GMU).
- Community classifier: trained with files submitted to the PDFrate service and is retrained frequently, called PDFrate (community).

Each instance of the tools was evaluated with evaluation datasets explained in Table 3 (both old and recent files). The experiment was repeated 10 times, each iteration used 100 benign files and 100 malicious files.

3.3 Results

A pilot experiment was conducted, using a fraction of the available datasets, summarized in Table 4. Further more, only 2 tools were tested: PDFratefrom [16] and Slayer [10], as they are both currently available.

Table 5 shows a summary of the results for all tools examined. Both instances of Slayer showed high detection accuracy when evaluating 2013 files (old dataset). All PDFrate classifiers achieved near perfect detection accuracy when testing the 2013 dataset. When evaluating a more recent dataset, all classifiers showed decreased performance, where the detection accuracy dropped to 74%–77%.

Table 4. Available datasets.

Name	# of files	Label	Source
Contagio	9,000	Benign	[4]
Contagio	10,000+	Malicious	[4]
VirusTotal	10,500	Malicious	[21]
TPN	1000	Benign	[13]
Personal	900	Benign	Personal & search

Table 5. Summary of the experiment's results. Both tools performed accurately when evaluating the old dataset (˜2013), but performance dropped significantly when evaluating a dataset collected more recently (˜2017).

Tool	**Evlautation datase**	**Accuracy**	**Missclass rate**	**FP rate**	**Percision**
Slayer2017	**2013**	90%	10%	19%	99%
	2017	74%	24%	45%	93%
Slayer2013	**2013**	93%	6%	12%	98%
	2017	78%	21%	33%	89%
PDFrate(contagio)	**2013**	100%	0%	0%	100%
	2017	77%	23%	35%	88%
PDFrate(GMU)	**2013**	100%	0%	0%	100%
	2017	77%	23%	40%	93%
PDFrate(Community)	**2013**	100%	0%	0%	100%
	2017	76%	24%	39%	90%

4 Discussions

Results shown in Table 5 provide insight on the nature of PDF detection. As malware writers started adapting new and improved techniques to embed their malicious code, obfuscate it, or evade detection, classifiers were not able to match such improvements. This could be the result of old training datasets, resulting in aging classifiers, or, the feature sets utilized are no longer relevant.

The PDF standard is continuously improved, introducing new features, and limiting access to older (specifically; more exploitable, dangerous) features. Therefore, the feature set examined by a specific tool must also be frequently revisited, to introduce new relevant features, and exclude irrelevant features. Otherwise, the classifier will suffer from overfitting, or worse, being built for a particular version of the PDF standard, limiting its benefits significantly.

To prove the above point, a number of PDF/A files were included in the 2017 benign evaluation dataset, which make around 40% of that dataset. This change caused all classifiers to perform severely unreliably, as shown in Table 6.

Table 6. Detection accuracy of 2017 benign dataset, which contains around 40% PDF/A files.

	Slayer 2017	Slayer 2013	PDFrate (contagio)	PDFrate (GMU)	PDFrate (community)
Accuracy	54.30%	66.60%	65%	60.13%	61%

Slayer 2013 achieved similar results to PDFrate (Contagio), where both tools were trained using the same dataset (Contagio), with the difference being that

Slayer was trained with 1800 total files, when PDFrate was trained with a total of 10,000 files. Several factors could lead to this, including: (1) Slayer covers all aspects of static analysis, where it examines content, structure, and parses JavaScript, producing features that cover more aspects, compared to PDFrate that considers structure and metadata only[2]. (2) The algorithm used in the classifier: Slayer employs an adaptive boosting algorithm, while PDFrate uses a bagging algorithm: Random Forests.

5 Conclusion

Several tools have been proposed since the appearance of malware-embedded-documents. Specialized tools started to appear around 2012, utilizing different techniques and approaches, ranging from static to dynamic, tools that barely look at metadata, to more advanced that extract and examine JavaScript and other content types, to those that perform real-time monitoring. Despite that, no tool is yet to offer a 1-package solution that counters malicious PDF file, and each tools is susceptible one or more attack type, such as obfuscation, parser confusion, and evasion.

In this paper, we attempted to identify drawback in current state of the art malicious PDF detectors. This was done via a survey of the tools, followed by a comparative evaluation of the available tools. The experiment attempted to identify concept drift in 2 classifiers: PDFrate and Slayer. It was found that the classification accuracy significantly dropped when trained with old dataset, and encountered newer samples collected in later years. The accuracy also significantly dropped when using different PDF formats and standards, such as PDF/A, where the classifiers frequently miss-classified around 50% of benign samples as malicious. Other findings of this paper include the following drawbacks of malicious PDF detectors: (1) Current tools work at the client-level only (user machines), and do not consider the distribution mechanism. (2) Current tools require high level of user interaction in order to submit a file for evaluation.

Acknowledgements. We would like to thank Mustafa Al-Saegh for helping with dataset cleaning and preparation. We would also like to thank VirusTotal, the owner of the Contagio dataset and TPN for providing access to their files. Finally, we are grateful to the authors and creators of PDFrate and Slayer, for providing access to their tools.

References

1. Adobe: Adobe reader security patches (2017). https://helpx.adobe.com/security/products/reader.html
2. Adobe: PDF technology center (2017). http://www.adobe.com/devnet/pdf.html

[2] Neither article Slayer [10] and PDFrate [16] covers the feature set in full details, thus preventing full analysis of the observation.

3. Carmony, C., Hu, X., Yin, H., Bhaskar, A.V., Zhang, M.: Extract me if you can: abusing PDF parsers in malware detectors, In: NDSS (2016)
4. Contagio: Contagio malware dump (2017). http://contagiodump.blogspot.com.au
5. CVE: PDF-related vulnerabilities (2017). https://cve.mitre.org/cgi-bin/cvekey.cgi?keyword=PDF
6. Esparza, J.M.: PDF attack - a journey from the exploit kit to the shellcode (2014). https://www.blackhat.com/docs/eu-14/materials/eu-14-Esparza-PDF-Attack-A-Journey-From-The-Exploit-Kit-To-The-Shellcode.pdf
7. Laskov, P., Šrndić, N.: Static detection of malicious JavaScript-bearing PDF documents. In: Proceedings of the 27th Annual Computer Security Applications Conference, pp. 373–382. ACM (2011)
8. Li, W.-J., Stolfo, S., Stavrou, A., Androulaki, E., Keromytis, A.D.: A study of malcode-bearing documents. In: M. Hämmerli, B., Sommer, R. (eds.) DIMVA 2007. LNCS, vol. 4579, pp. 231–250. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-73614-1_14
9. Liu, D., Wang, H., Stavrou, A.: Detecting malicious JavaScript in PDF through document instrumentation. In: 2014 44th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), pp. 100–111. IEEE (2014)
10. Maiorca, D., Ariu, D., Corona, I., Giacinto, G.: A structural and content-based approach for a precise and robust detection of malicious PDF files. In: 2015 International Conference on Information Systems Security and Privacy (ICISSP), pp. 27–36. IEEE (2015)
11. Maiorca, D., Giacinto, G., Corona, I.: A pattern recognition system for malicious PDF files detection. In: Perner, P. (ed.) MLDM 2012. LNCS (LNAI), vol. 7376, pp. 510–524. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-31537-4_40
12. McAfee: Mcafee september 2017 threat report (2017). https://www.mcafee.com/au/resources/reports/rp-quarterly-threats-sept-2017.pdf
13. Trent Nelson: PDF collection (2017). https://github.com/tpn/pdfs
14. Neupane, A., Saxena, N., Maximo, J.O., Kana, R.: Neural markers of cybersecurity: an fMRI study of phishing and malware warnings. IEEE Trans. Inf. Forensics Secur. **11**(9), 1970–1983 (2016). https://doi.org/10.1109/TIFS.2016.2566265
15. NIST: National vulnerable database (2017). https://nvd.nist.gov
16. Smutz, C., Stavrou, A.: Malicious PDF detection using metadata and structural features. In: Proceedings Of The 28th Annual Computer Security Applications Conference, pp. 239–248. ACM (2012)
17. Smutz, C., Stavrou, A.: When a tree falls: using diversity in ensemble classifiers to identify evasion in malware detectors. In: NDSS (2016)
18. Šrndić, N., Laskov, P.: Detection of malicious PDF files based on hierarchical document structure. In: Proceedings of the 20th Annual Network and Distributed System Security Symposium (2013)
19. Šrndić, N., Laskov, P.: Hidost: a static machine-learning-based detector of malicious files. EURASIP J. Inf. Secur. **2016**(1), 22 (2016)
20. Tabish, S.M., Shafiq, M.Z., Farooq, M.: Malware detection using statistical analysis of byte-level file content. In: Proceedings of the ACM SIGKDD Workshop on CyberSecurity and Intelligence Informatics, pp. 23–31. ACM (2009)
21. VirusTotal: Virustotal (2017). https://www.virustotal.com
22. Xu, M., Kim, T.: PlatPal: detecting malicious documents with platform diversity. In: USENIX Security Symposium (2017)

Shifting the Burden: An Ineffective 'Quick Fix' to the New Zealand Tire Problem

Aldrich Rasco(✉) and David Sundaram(✉)

Department of Information Systems and Operations Management,
University of Auckland, Auckland, New Zealand
{aras613, d.sundaram}@auckland.ac.nz

Abstract. In New Zealand, we dispose off around 5 million tires, and 70% end up in a landfill. There, the tires occupy a considerable amount of space. Often unaccounted for once disposed off, the tires eventually attract pests and other contaminants. This situation is highly dangerous to nearby eco-systems, housing animals, and people alike. When tires are stacked in one area, there is a severe risk of fire and increased soil pollution. However, behind these real consequences, there lies a more significant problem with how tires are facilitated overall. There is no standardized process to dispose or recycle tires to prevent future build up. New Zealand's deteriorating concern for tires is symptomatic of a more substantial problem which is our diminishing respect for sustainability. While the country prides itself on a clean and green image, it is without doubt our environmental awareness is deteriorating. The purpose of this research is to reclaim our consciousness and bring attention to a problem that is growing more prominent and visible across the country. The paper aims to shed insight with regards to the viability of modeling a sustainable zero waste supply chain from the perspective of tire conservation. Firstly, through the identification of the dynamic issues at play, we suggest potential solutions to our problems. Secondly, by demonstrating the viability of theoretical models, practical action is one step closer. Given that activity on tire conservation has been lagging for over a decade, this research would be of interest to academics, as well as crucial environmental decision-makers in New Zealand.

Keywords: Tire conservation · Tire recycling · Retreading
Zero waste economy · Sustainable zero waste supply chain
Shifting the burden archetype

1 Introduction

Any responsibility for tire conservation is exhaustive; especially when the motivation to recycle is mostly predicated on environmental concerns alone. Dealing with the sheer scale of tires requires much more than encouragement and passion for gaining reasonable progress. Financial incentives prove to be a proactive accelerator when tasks are exhaustive. While there are profitable solutions to such a problematic tire problem, they are not too lucrative, or if they are, implementation is too complicated or overwhelming. One lucrative venture is the concept of Tom's (www.Toms.com): their business model is based on the idea of having unemployed citizens carve out footwear

R. Doss et al. (Eds.): FNSS 2018, CCIS 878, pp. 140–146, 2018.
https://doi.org/10.1007/978-3-319-94421-0_11

from used tires. In New Zealand, executing a similar vision is not unreasonable, especially when the possibilities for end products for tires are countless. Soundproofing rubber, cement, infrastructural walls and de-vulcanized tires are a few examples [1]. These examples affirm the point that businesses based around tires are feasible. As a raw material: tires cost close to nothing, which is beneficial for ventures that aim to transform tires into profitable sales. Tire stockists such as bike shops pay to get rid of their tires. Since tire procurement is cheap, the task itself can be a further point of income. Combine this effort with any revenue from product sale and the vision seems viable. However not all tire conservators are entrepreneurs, although they may have elements of this vision. They may understand the potential profitability of using tires – yet, often they do not connect the dots to facilitate such a venture.

In tire conservation, motivation is a problem which is symptomatic of our deteriorating concern over sustainability. In the industry, financial incentives are a useful driver however if environmental consideration does not guide it then progress towards profitable recycling ventures may stagger. In fact, money can be a driver in the opposite direction. The industry exhibits such a motivation problem in ‘shifting the burden.’ Here, tire responsibility is forgotten as leaders settle for the consolation prize of what little profit they can eke from storing tires. Tires are often bought out to headline a dream of facilitating a lucrative tire recycling venture. As the challenges get harder and more complicated; and as the sheer number of tires and tire responsibility increases, tire recycling ventures opt out of responsibility to shift the burden elsewhere. There is little profit in storing tires. Landfills that do so, temporarily hold the tires and eventually clip the ticket to profit from other dreamers who have potential solutions to the problem. Still, defeat by the overwhelming tire numbers is a common theme which makes the burden-shifting a literal ‘merry go round’ with ventures and councils suggesting they know the right fix [2]. At the heart of all burden shifting is the utilization of a ‘quick fix’. While temporary relief can be achieved by pushing the problem away, progressing towards a more sustainable solution such as recycling the tires halts. From a utilitarian perspective, as problems get pushed under the carpet with very little progress or return, we can deem the task unproductive. The research attempts to highlight these problems through modeling the situation and suggesting potential solutions to the fundamental issues. Through understanding the practical problems, we must also state the research gap. There is very little academic literature in system dynamic modeling of sustainable zero waste supply chains which attempt to recycle or reuse products at the end of their life.

2 ‘Shifting the Burden’ Macro-dynamic

Figure 1 illustrates a causal loop diagram of shifting the burden that usually applies to landfill owners. Storing tires elsewhere is a ‘quick fix’ that temporarily relieves responsibility for any tire build up. Because someone else would be responsible for the tires, pushing the problem away is a temporary solution; responsibility is evaded. The positive relationship of the tire build up and proactive recycling emphasizes that as the buildup becomes more prominent, the need for a solution is becoming more prominent – albeit a temporary one. Therefore, there are balancing dynamics with both the quick fix (storing tires elsewhere) and the actual solution (proactive recycling). Both solutions would relieve the buildup distinctly: proactive recycling will have a more sustainable effect in the long term, compared to the quick fixes' temporary implications. When the quick fix is selected, environmental deterioration passively occurs as time passes. Storing tires elsewhere does not truly contain environmental risk. However, the proactive recycling of it will. As fewer tires are recycled and taken out of overcrowded spaces, risk can be avoided or minimized. On the contrary, when tires are stored elsewhere, the disadvantages such as rats and other contaminants ensue. Any sense of relief felt with a quick fix is temporary along with the 'risks' that may be appeased just because tires would have some time away from ecosystems due to transportation [3].

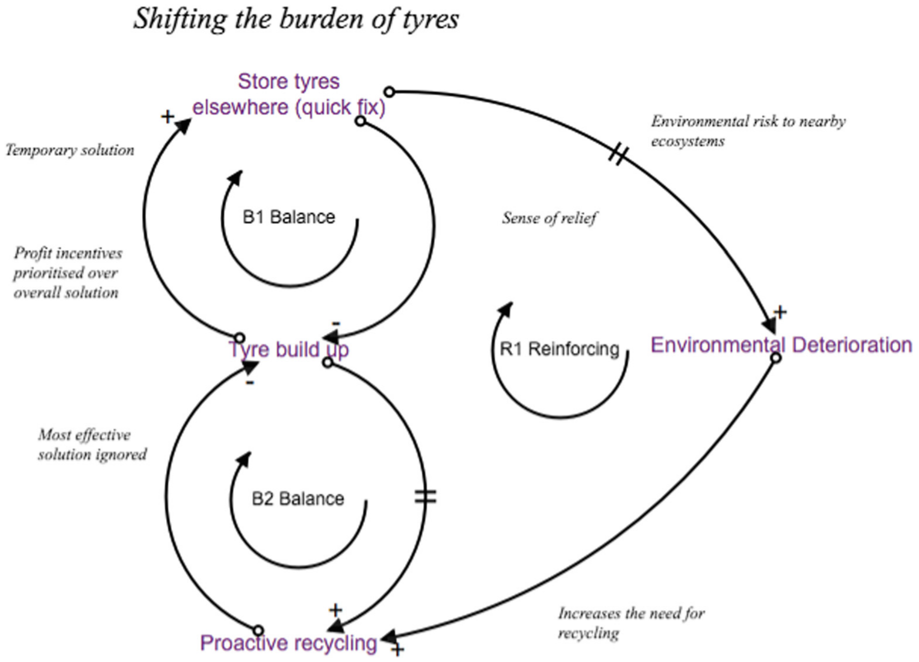

Shifting the burden of tyres
Store tyres elsewhere (quick fix)
+
Temporary solution
B1 Balance
Environmental risk to nearby ecosystems
Sense of relief
Profit incentives prioritised over overall solution
-
Tyre build up
-
R1 Reinforcing
+
Environmental Deterioration
Most effective solution ignored
B2 Balance
+
Proactive recycling
+
Increases the need for recycling

Fig. 1. Shifting the burden of responsibility in tire conservation

The reinforcing nature (R1) of this dynamic increases the need for recycling. In turn, the situation portrays poor mental models regarding attitudes towards recycling and sustainability. Short term profit incentives through 'clipping the ticket' while passing the burden of tires are prioritized over finding effective solutions to the tire problem. The ignorance towards proactive recycling is further symptomatic towards our degenerating concern for sustainability. The following section will delve deeper into the 'quick fix' so that its analysis can help us derive effective solutions to the situation.

3 The 'Quick Fix' Micro-dynamic of 'Shifting the Burden'

To be able to make effective recommendations for solution over the shifting the burden situation, we must delve deeper into the quick fix itself. Figure 2 breaks down the quick fix micro-dynamic.

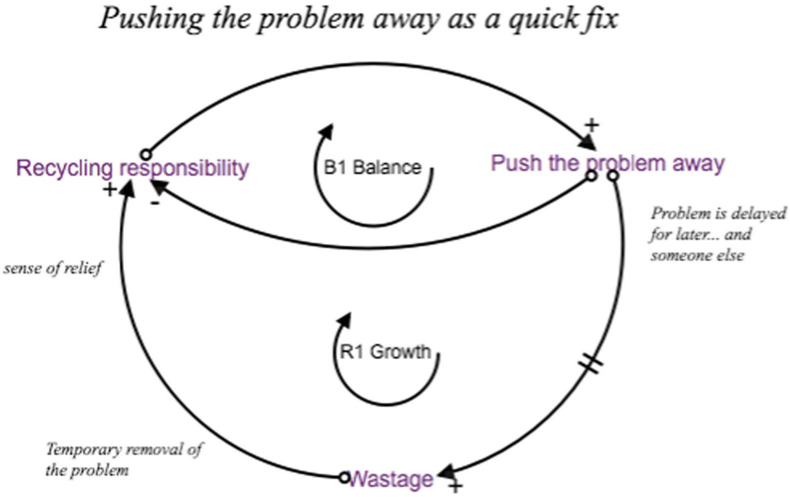

Pushing the problem away as a quick fix
Recycling responsibility
B1 Balance
Push the problem away
+
+
-
Problem is delayed for later... and someone else
sense of relief
R1 Growth
Temporary removal of the problem
Wastage
+

Fig. 2. Breaking down the quick fix micro-dynamic

If we model society as one systematic supply chain, it is easy to see how continuously shifting the burden creates wastage of some sort like time, effort and resources spent on realizing solutions for the tire problem and eventually giving up on this pursuit. By focusing on pushing the burden itself, we see that the dynamic becomes a fix that ultimately fails because it prolongs the problem for society and wastes resources from individual units. From a utilitarian perspective, the value of pushing the tire burden would be less if ventures decided to continue to pursue a sustainable solution (proactive recycling in Fig. 1: 'shifting the burden') or at least made efforts to collaborate with similarly minded ventures. From a utilitarian perspective, again, this process could be less wasteful with more consistency and standardization within the overall tire recycling supply chain. A system that has a consistent inflow and outflow of tires would be able to identify process bottlenecks more quickly, and as such, the wastage in ideation and collaboration can be avoided at micro levels and in individual ventures.

This dynamic is at the heart of the utilitarian waste created in this situation. R1 will show that pushing the problem away establishes a sense of relief through the temporary removal of the waste problem for a venture (this is similar to what happens at the macro-dynamic of shifting the burden). Similarly, a balancing act is required (B1) which plays into responsibility and the conscience of 'quick fixers.' As recycling responsibility increases so does the need to push the problem away. Therefore, a predictable ‘relieved’ mental model is observed for ‘quick fixers’. However, the quick fix will not continuously result in relief as time passes. It will diminish the more it is used in practice; eventually, conscience should catch up to quick fixers and cause some discomfort through the constant ignoring of sustainable proactive recycling.

This dynamic is also important for noting down the effects that it leads to in the overall scheme of things. In the tire situation, this fix is a common procedure - which means that several ventures or councils take actions which further exacerbate the problem. Since it has become common to ‘shift the burden,’ there has been little progress in solving the tire problem.

4 Solutions

To address the issues found at the heart of shifting the burden and the ‘quick fix’, it is crucial that sensible solutions are recommended. This section aims to recommend viable solutions that would construct a more consistent tire conservation process. Figure 3 summarizes our findings.

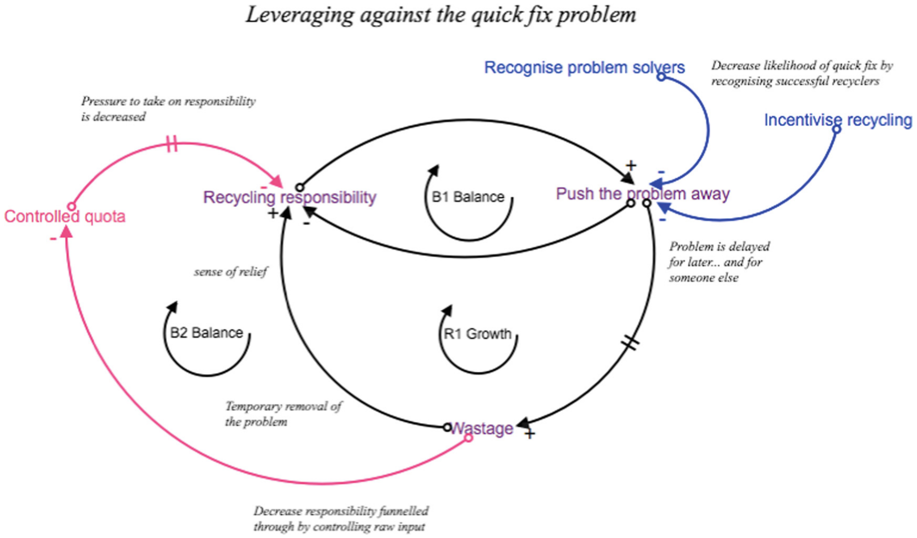

Leveraging against the quick fix problem
Recognise problem solvers
Decrease likelihood of quick fix by recognising successful recyclers
Incentivise recycling
Pressure to take on responsibility is decreased
Controlled quota
Recycling responsibility
B1 Balance
Push the problem away
Problem is delayed for later... and for someone else
sense of relief
B2 Balance
R1 Growth
Temporary removal of the problem
Wastage
Decrease responsibility funnelled through by controlling raw input

Fig. 3. Solutions to the quick fix problem

4.1 Scoping: A Controlled Quota

Figure 3 proposes several solutions to combat the quick fix, which ultimately should have leading effects on the macro-dynamic that is ‘shifting the burden.’ The first solution is about implementing a controlled quota over the number of tires entering society. When the raw number of tires circulating are reduced, recycling responsibility should be statistically lower. One key reason why the tire burden is shifted consistently is that the sheer number of tires or the responsibility that landfill owners have are just too big for them to handle. When the scope is decreased, they can deal and apply solutions rightfully in more doable portions. Therefore, as responsibility is decreased, the wastage created in ‘quick fix’ scenarios would also be smaller. When the impact of the ‘quick fix’ becomes smaller, the macro-dynamic of ‘shifting the burden’ becomes less problematic because overall the number of tires for recycling is less. Although the habit is not ultimately fixed, a smaller number of tires overall is easier to handle.

4.2 Habits: Recognition and Incentivizing

‘Pushing the problem away’ in the ‘quick fix’, or ‘shifting the burden’ has an important attitude problem. To work against habits, we must adhere to positive reinforcement such as recognizing key problem solvers and incentivizing recycling. The New Zealand government has already tried negative reinforcement through taxing, fining and sending out abatement notices to landfills which do not proactively aim to commit recycling initiatives over their tires [4]. We believe the reason for this failure is because recycling is closely intertwined with motivation and incentives. Negatively punishing an exhaustive mental model towards recycling will only deter future initiatives to do so. Furthermore, such a deterrent can be realized beyond sanctioned individuals. If current ventures and willing up-and-comers are going to be or will be, punished for trying, then future initiatives will be suppressed, and growth is slower and less predictable. Recognizing problem solvers will help inspire heroes within sustainability to attempt their ventures. Incentivizing any form of recycling can be a reasonable gateway to furthering effort towards the tire problem. The solution is based on the idea that when inspiration, visibility, and motivation is increased then habit can be fundamentally changed for the better, and for the best interests of sustainability and ultimately society.

5 Conclusion

Overall, this research aims to model and simulate sustainable zero waste supply chains so that the tire problem in New Zealand can be solved. We identified two key problems: ‘shifting the burden’ macro-dynamic and explored in depth the ‘quick fix’ micro-dynamic. We also explored two potential solutions to these problems: namely a controlled quota, recognition of problem-solver, and incentivizing recycling. Further work needs to be conducted in terms of parametrization and validating the models proposed. We hope that publishing these models will raise awareness amongst academics, practitioners, and society regarding the problems we face and how we could potentially solve them.

References

1. Tyre Recycling Waikato—Tyre and Rubber Recycler—Hamilton, NZ. http://tyrerecyclingwaikato.co.nz/. Accessed 4 May 2018
2. Environment Minister offers grants for tire recycling ideas - NZ Herald. https://www.nzherald.co.nz/nz/news/article.cfm?c_id=1&objectid=11535390. Accessed 4 May 2018

3. Government announces tire recycling plan—Newshub. http://www.newshub.co.nz/home/new-zealand/2017/06/government-announces-tyre-recycling-plan.html. Accessed 13 Mar 2018
4. 1 News Piles of used tires proving mounting problem across NZ—1 NEWS NOW—TVNZ. https://www.tvnz.co.nz/one-news/new-zealand/piles-used-tyres-proving-mounting-problem-across-nz. Accessed 12 Mar 2018
5. Shand, M.: Millions of tires a hazard, Waikato Regional Council decides—Stuff.co.nz. https://www.stuff.co.nz/business/97643158/millions-of-tyres-a-hazard-waikato-regional-council-decides. Accessed 12 Mar 2018

A New Method for Sharing the Public Keys in Opportunistic Networks

Samaneh Rashidibajgan[1(✉)] and Robin Doss[2]

[1] Rostock University, Albert-Einstein Strasse 22, Rostock, Germany
samaneh.rashidibajgan@uni-rostock.de

[2] Deakin University, Melbourne Burwood Campus, Burwood, Australia
robin.doss@deakin.edu.au

Abstract. Opportunistic Networks do not have a fundamental infrastructure, and different nodes in these networks have the role of a sender, receiver and a router. Intermediate nodes should route messages to neighbors and they should extract sufficient information for this purpose while the content of the message is still hidden. In order to achieve this aim, messages should be encrypted. While there is not a constant path between two specific nodes in OppNet, it is not possible to use traditional solutions such as a trusted third party for sharing the public key. In this paper, a new confidentiality structure is proposed in order to encrypt the messages and sharing nodes' public key in Opportunistic Networks. By this new structure, intermediate nodes may route messages while they are not able to extract payload of a message.

Keywords: Opportunistic networks · Key management
Messages encryption

1 Introduction

Opportunistic Networks (OppNet) are a subset of Delay Tolerant Networks (DTN) [1] which are trying to provide reliable transmission in an intermittently connected environment, as well they are offered long and unpredictable delays. In these networks, an end-to-end connection is not available. OppNets are usually established in wildlife tracking [2], search and rescue [3], underwater sensor network [4], and military environments or other places where a fundamental infrastructure is not available. In the OppNet, the Internet connections are not required and messages are sent from a source to the destination through different intermediate nodes.

There are two main forwarding methods in OppNet [5,6]:

- Context based forwarding: In this approach, each node has a profile which is built by the node's interests, and each message has a header which contains one of the interests of a message sender. When a node in the network receives a message, it compares its profile with the header of the message, and if these

R. Doss et al. (Eds.): FNSS 2018, CCIS 878, pp. 147–161, 2018.
https://doi.org/10.1007/978-3-319-94421-0_12

two are matched, the node will be a destination; otherwise this message is forwarded to neighbors which can be interested in the message or it is directed to a route which makes it nearer to the destination.
- Content based forwarding: receivers advertise their interests and intermediate nodes save these interests in the respective tables. A received message through the intermediate nodes is forwarded according to the related table to a neighbor that may be interested in this message.

Context and Content of a message referring to the senders' or receivers' interests profile, and this message should be sent via various intermediate nodes which many of them are not trusted. Maintaining confidentiality of messages, and anonymity for receivers and senders is a serious concern in such networks. Thus, it is important to introduce a methodology in OppNet for maintaining confidentiality for messages as proof of trustworthiness. The message encryption and proper a key management method are some solutions for this issue [7].

Traditional encryption and sharing key methods are not adequate for OppNet. Due to the topology of OppNet, a trusted third party cannot be an appropriate solution to produce and share the public and private keys in OppNet, and key management is a challenge in such a network [8]. The receiver of a message is not clear for the sender of a message, and it is not possible to use the public key of the receiver for encryption of a message according to the standard cryptography methods. On the other hand, messages are sent via various nodes and it is not safe to send them as a text. A method to encrypt a message while intermediate nodes cannot read it, but they can route it, is essential in OppNet. According to [6] the header and the payload of a message should be encrypted in different ways in OppNet. As a result, intermediate nodes will be able to route a message while they cannot extract the concept of message's payload.

In common networks, a sender knows the public key of the receiver, and it uses this key for encryption of a message; but in OppNet, nodes even do not know who is the receiver. In order to overcome this problem, a new method is proposed in this paper which the sender can share its public key in the header of a message while it is not accessible for intermediate nodes. For this purpose, two parts for attributes in the profile tables of nodes are used in this paper. In this way, when a node is going to make a message according to one of its interests, it creates the message and shares its public key by placing it in a part of the packet. When a node receives a message, and finds out that it is the receiver, it can achieve the public key of the sender and uses it in order to connect to the sender.

In this study, the following results will achieve:

- A message will not be sent as a text which will not be readable for all members of the network.
- The public key for a group of people who are interested in a special topic is shared. Therefore, it is a secret key for a specific session between a group of people.

- Intermediate nodes route messages without knowledge about its payload, or recognizing the sender or receiver of a message, and the process of encryption and decryption is not done in intermediate nodes.
- The network will be resisted against the dictionary attacks for groups which are sharing special codes as their interest items in their profile table.

In addition, the following issues are not considered in this study:

- Usual messages, which are related to the social events and are related to the guessable topics in the profile table, do not resist against the dictionary attacks.
- These usual messages are not resisted against Man in the middle attacks too, but we assume that a trusted function is used, and messages are sent to the nodes which are partly trusted. Thus, we suppose that these trusted nodes will not use Man in the Middle attacks.
- Subjects related to the physical layer of the network are not considered in this research.

Furthermore, in this paper, we assume that multi-copy of messages are sent to some different neighbors.

The rest of this paper is organized as follows: a description of related works in the literature is summarized in Sect. 2, and in Sect. 3, context and content routing protocols and their challenges are described. The problem statement is described in Sect. 4. Section 5 describes the proposed model. The algorithm is evaluated in Sect. 6 and finally, conclusion is followed in Sect. 8.

2 Related Works

A Multiple Layer Commutative Encryption (MLCE) for providing privacy in the content base opportunistic networks were used in [5]. In this algorithm, data are encrypted several times with various keys. This algorithm is based on a tree structure for neighbor nodes, and nodes share the respective keys with their parents, grandparents, children and grandchildren in this tree.

In [6], authors proposed an algorithm based on both IDbased and policy-based encryption algorithms. This protocol has four security primitives: encrypt_header, encrypt_payload, match_ header, decrypt_payload. A hash function for encrypt_header was used in this algorithm. Each node hashes its profile with a similar hash function for match_ header and compares its profile items with the header of the received messages. Authors assume that in the setup phase, every node has access to a Trusted Third Party (TTP) for receiving public and private keys. The policy_based encryption has been applied for encrypt_payload.

A self-organized key based on Ant Algorithms for Ad-hoc networks was proposed in [9]. In this algorithm, every node produces a set of public-private keys, and neighbors certify the public key for each other. Also, each node saves the trust level of other nodes. A node sends ants toward the destination and ants try to find the most trustworthy certificate chains. When ants find a trust path between the source and a destination, they leave the traces of pheromones along the path. This technique is not resistant against Sybil attacks. Furthermore, a certain path between a source and a destination is considered which is impossible in OppNet.

Privacy_Enhance Opportunistic Networking (PEON) was introduced in [10]. In this algorithm, a message is encrypted in different layers. There is a chain of intermediate nodes between the source and a destination and each node is aware of only the next node. In order to reduce delays and overhead of the network, they considered some groups of nodes and each group shares a public key pair.

Cananiss et al. [11] used a chaining algorithm in order to secure messages. According to this paper, each intermediate node encrypts and decrypts a message before forwarding it to the final destination. There is multiple layers encryption and intermediate nodes do not have access to the content of a message. Also, a message is broken into several fragments and these fragments are forwarded in parallel to several intermediate nodes in this algorithm.

An onion based algorithm is introduced in [12]. There are two important assumptions in the onion encryption techniques: relays can communicate with each other, and users know the list of available routers. This structure cannot be useful in a network which nodes have limited knowledge about other relays. In order to tackle the problem, authors in [12] proposed that each node chooses some neighbors and makes a circle between them, so it does not need knowledge about all relays. Intermediate nodes use a Bloom filter to communicate with each other without disclosing their identities. The Distributed Hash Tables (DHT) are exploited to distribute onion keys.

An onion based anonymous routing is proposed in [13] which group onion idea is used in this research. Messages are encrypted/decrypted layer-to-layer by different groups from a source to the destination, so every node in the same onion group can encrypt/decrypt the corresponding layer. By this scheme, the source and destination of a message remain unknown for intermediate nodes. For onion routing, limited knowledge about the next and previous router is necessary.

A self-organize public key management is proposed in [14] based on the digital signature. Nodes have a list of digital signatures which contains nodes identifiers. In a fix time intervals, nodes certificates are signed by other neighbors and nodes do not have knowledge about the size of chain.

3 Challenges of Routing Algorithms in OppNet

In this section, an overview of two main routing algorithms in OppNet is provided in order to check out various challenges to maintaining confidentiality for messages in these networks.

3.1 Context Based Routing

In context based protocols, each node has a profile table which is contained user's interests. Senders make a message which has two parts: (1) Header or control information (can be two simple words like Soccer Sport), (2) Payload (can be a complex and long text). When a match occurs between the header of a message and the profile table of a node in the network, this node is the destination. Senders are not interested in revealing of this header for all intermediate nodes. Furthermore, intermediate nodes, in order to route messages, should have limited knowledge about the encrypted profile table of nodes which are visited frequently.

Therefore we need:

1. An encryption method to encrypt the header of messages and the profile table of each node. The header of a message should be encrypted by a method which intermediate nodes can compare it with their profile tables in order to find whether it is the receiver or not.
2. An encryption method to encrypt the payload of a message. (the encryption of a message payload and the header of a message should be different while the intermediate nodes should be able to analyze the header of a message for routing packets, but they should not be able to recognize the concept of the payload part of a message).
3. Intermediate nodes should be able to establish their tables based on the encrypted profile context of frequently visited nodes, search on it and make a decision to save or drop a message.

3.2 Content Based Routing

In content based routing protocols, senders do not have knowledge about receivers. A receiver advertises its interests which contain two parts: (1) Control information (can be two simple words like Soccer Sport), (2) Payload (can be a complex and long text).

Each intermediate node has a table and it saves received advertisements about other nodes on it and compares them with control information of messages and route messages in a correct direction.

In order to optimize bandwidth usage, similar advertisements are merged and they are recognized as one interest with several receivers. When there will be a match between the control information of a message and an advertisement, this message will be sent to all receivers which had advertised the text. It means that a packet can have several receivers.

Receivers do not want other nodes to know about their advertisements, thus these advertisements should be encrypted and intermediate nodes should establish their tables based on this encrypted content. Furthermore, a publisher (sender) wants to encrypt both control information and the published content.

Therefore followings are required:

1. An encryption method to encrypt advertisements of receivers and control information of senders.
2. An encryption method to encrypt the payload of sender's messages. (the encryption of a message payload and its control information should be different while intermediate nodes should be able to analyze control information

of sender's messages and advertisements of receivers for routing packets, but they should not be able to recognize the concept of the payload part of a message).

4 Problem Statement

The Public Key Infrastructures (PKIs) are usually used for public keys distribution, the key management and authentication of the owner of a key as well. Certificate Authority (CA) is the most common solution for this purpose. The trusted third party is widely used in all of these authentication systems. It is notable that, there is an organization for users authenticating and sharing public keys. The topology of OppNet is varying frequently, thus the trusted third party based authentication systems are not applicable in OppNet.

There are three problems in using a trusted third party in OppNet: first, the topology of these networks is not a fix. Nodes are moving and they may connect and disconnect to the network frequently, thus it is not easy for a node to connect to a PKI. Second, the destination of a message is not known, therefore each intermediate node must compare the header of a message with the respective table and consequently decision is taken according to this comparison in order to send the message in the best direction heading to the node's destination. Because nodes do not know about the receiver, they can not use the receiver's public key for encryption of a message. Third, even if it will be possible to use a PKI, receiving authentications and keys takes a long time and causes a delay which leads to a possible invalid key. As a result, it is important to develop a method for sharing public keys of users. In this paper, a new algorithm for sharing the public key is proposed. The public key of a sender is sent in the header of a message instead of using the public key of the receiver. Every node, which finds similarity between its profile table and a message's header, can achieve the public key and decrypt the message.

To be more precise, we have considered a scenario which contains a network composed of a set of n nodes $\{N_i\}_{1 \leq i \leq n}$. It is assumed that these nodes are students of a department in an university, and they walk around the university according to their semester schedule. The profile table of node i is defined as P_i and it contains a set of attributes $\{A_j\}_{1 \leq j \leq m}$, and each attribute A_j has two components of G_j which is a general category of an interest and L_j which is a limited concept of an interest in a profile. Thus each A_j uses a couple words, for example, "Soccer Sport" can be a A_j which $G_j = Sport$ and $L_j = Soccer$. The pair (G_{ji}, L_{ji}) is the attribute j of node i and is recognized as $A_{j,i}$. Node i has the profile P_i, and it is the concatenation of all attributes of node i: $P_i = A_{1,i} \| ... \| A_{n,i}$ (Table 1).

Each node has a key chain, and there is a couple of public and private keys for each profile attribute in this key chain $\{Pk_j, PV_j\}_{1 \leq j \leq m}$. Thus, the pair $\{Pk_{ji}, PV_{ji}\}$ are the public key and the private key of attribute j of node i respectively (Table 2).

Each message M in the network has a header $Header(M)$ and a payload $Payload(M)$. When a sender node $N_S (1 \leq S \leq n)$ wants to send a message

M to a receiver node $N_R (1 \leq R \leq n)$, N_S makes a message as follows: $M = Header(M) \| Payload(M)$.

Table 1. Profile table for node i (P_i)

	G_i	L_i
A_{1i}	G_{1i}	L_{1i}
A_{2i}	G_{2i}	L_{2i}
...	...	...
A_{ji}	G_{ji}	L_{ji}
...	...	...
A_{mi}	G_{mi}	L_{mi}

Table 2. Keys chain for node i

	Public key	Private key
1	PK_{1i}	PV_{1i}
...	...	...
j^{th}	PK_{ji}	$PVji$
...	...	...
m^{th}	PK_{mi}	$PVmi$

5 Proposed Model

Nodes profile should be hidden from other nodes in OppNet, so the header of a message (control information) should be protected. Furthermore, the intermediate nodes should be able to route a message based on their routing tables. For this purpose, a Hash Function (HF) is used and both G_i and L_i of attribute A_i are Hashed as $HF(G_i)$ and $HF(L_i)$.

In proposed algorithm, when nodes are in the communication range of each other in the network, they use a Trust function in order to find out whether a neighbor is trusted or not; In this methodology, the trust function proposed in [15] has been exploited. The direct observation and a Game Theory strategy are applied in order to recognize trusted nodes in this trust function.

If a message is sent as a text to the network, all the intermediate nodes will be able to read it, therefore, it is better to encrypt messages to protect the message confidentiality. While the receivers of a message are not known, each node considers a separate public key for each interest in the profile table and add this public key in the header of a message. Actually, the public key is shared between nodes with a similar interesting field in their profile tables. In the following equations, E is an encryption method which is considered as $E(Payload, Key)$, E_S and E_{AS} are a Symmetric and an Asymmetric encryption algorithm respectively. The structure of a message when node A wants to send a message is as follows:

$$HF(G_{j,A}) \| E_S(PK_{j,A}, HF(L_{j,A})) \| E_{AS}(Payload, PV_{j,A}) \tag{1}$$

so we have

$$Header(M) = HF(G_{j,A}) \| E_S(PK_{j,A}, HF(L_{j,A})) \tag{2}$$

$$Payload(M) = E_{AS}(Payload, PV_{j,A}) \tag{3}$$

where $PK_{j,A}$ and $PV_{j,A}$ are the public key and the private key of j^{th} attribute of profile table of node A respectively.

Each node uses an asymmetric encryption in order to encrypt the payload of a message with private key $(PV_{j,A})$ of related attribute in its profile table; Also They use $HF(L_{j,A})$ as a key for symmetric encryption and they encrypt related public key $(PK_{j,A})$ with this item and add $HF(G_{j,A})$ to it.

In the context based routing, when node B receives a message, compares $HF(G_{j,A})$ of the message with its $HF(G_B)$ part of its attribute table, and if it finds a match in $HF(G_{l,B})$, tries to decrypt $E(PK_A, HF(L_{j,A}))$ with $L_{l,B}$. If it is decryptable, node B is a receiver $(HF(L_{j,A}) = HF(L_{l,B}))$ and it can achieve the public key of node A and read the payload. Otherwise, it will forward this message according to the $HF(G_{j,A})$ of a message to other neighbors.

In the content based routing, node B advertises its $HF(G_{l,B})$ and intermediate nodes save this Attribute in their tables. When a node receives a message, it forwards the message according to their routing tables. When receiver B takes a message related to a $HF(G_{l,B})$, it tries to achieve the public key by decrypting $E_S(PK_A, H(L_{j,A}))$ with its $HF(L_{l,B})$. If it can decrypt $E_S(PK_A, HF(L_{j,A}))$, it is the receiver and $HF(L_{j,A}) = HF(L_{l,B})$; otherwise, node B will discard this message. This process is shown in Fig. 1.

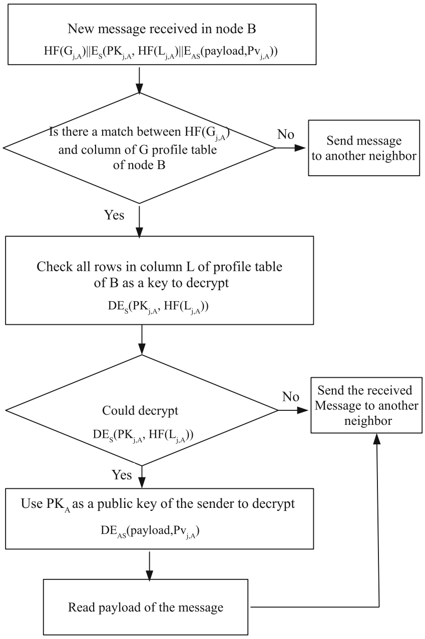

New message received in node B
$HF(G_{j,A})||E_S(PK_{j,A}, HF(L_{j,A})||E_{AS}(payload,Pv_{j,A}))$
Is there a match between $HF(G_{j,A})$ and column of G profile table of node B
No
Send message to another neighbor
Yes
Check all rows in column L of profile table of B as a key to decrypt
$DE_S(PK_{j,A}, HF(L_{j,A}))$
Could decrypt
$DE_S(PK_{j,A}, HF(L_{j,A}))$
No
Send the received Message to another neighbor
Yes
Use PK_A as a public key of the sender to decrypt
$DE_{AS}(payload,Pv_{j,A})$
Read payload of the message

Fig. 1. Process of sharing the public key between two nodes in OppNet

In order to make it more clarify, the concept is explained with an example which you can find it in the Figs. 2 and 3. Assume that node A wants to send a message which is related to jth of its profile attributes with the title of Soccer Sport.

$$(G_{j,A}, L_{j,A}) = (HF(Sport), HF(Soccer)) \tag{4}$$

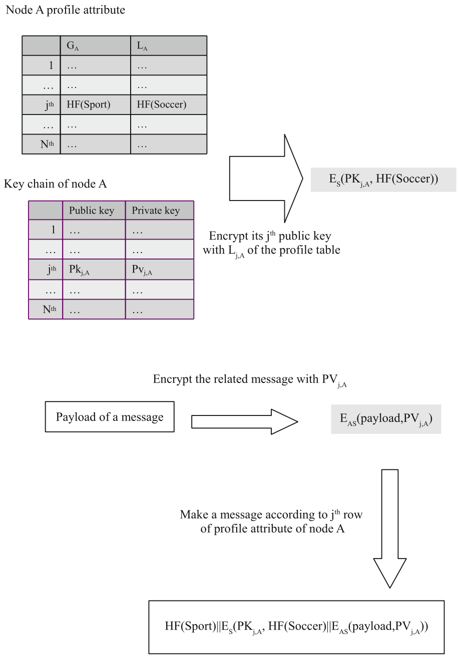
$$G_{j,A} = HF(Sport) \tag{5}$$

$$L_{j,A} = HF(Soccer) \tag{6}$$

So, the message is produced as follow:

$$HF(Sport) \| E_S(PK_{j,A}, HF(Soccer)) \| E_{AS}(Payload, PV_{j,A}) \tag{7}$$

When node B receives this message, compares $HF(Sport)$ with all parts of $HF(G_B)$ in its attribute table. If there will be a match in l$^{\text{th}}$ line of $HF(G_B)$, it uses $HF(L_{l,B})$ in order to decrypt $E_S(PK_{j,A}, HF(Soccer))$. If it can decrypt this header, it means $HF(L_{l,B}) = HF(Soccer)$, and node A and B have a similar interest, so node B can find the public key of node A $((PK_{j,A}))$, and it can decrypt the payload of the message; otherwise, the message will be sent to other neighbors who are interested in this subject. While the header of a message and attributes of a node are hashed, intermediate nodes cannot know these attributes.



Node A profile attribute

	G_A	L_A
1	…	…
…	…	…
j$^{\text{th}}$	HF(Sport)	HF(Soccer)
…	…	…
N$^{\text{th}}$	…	…

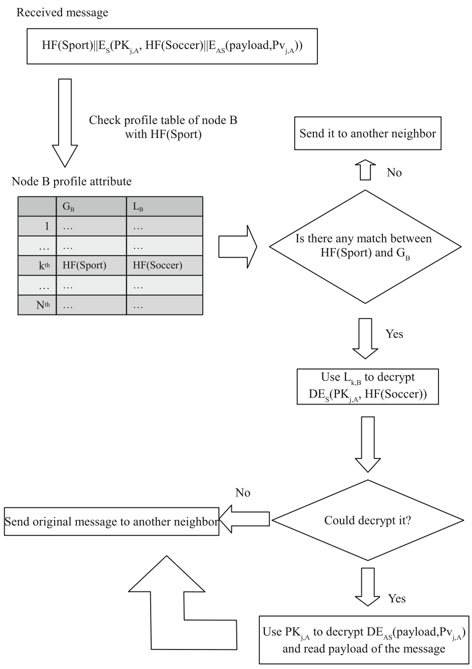
Key chain of node A

	Public key	Private key
1	…	…
…	…	…
j$^{\text{th}}$	$Pk_{j,A}$	$Pv_{j,A}$
…	…	…
N$^{\text{th}}$	…	…

$E_S(PK_{j,A}$, HF(Soccer))
Encrypt its j$^{\text{th}}$ public key with $L_{j,A}$ of the profile table
Encrypt the related message with $PV_{j,A}$
Payload of a message
E_{AS}(payload,$PV_{j,A}$)
Make a message according to j$^{\text{th}}$ row of profile attribute of node A
HF(Sport)||$E_S(PK_{j,A}$, HF(Soccer)||E_{AS}(payload,$PV_{j,A}$))

Fig. 2. Process of making a message and encrypt it in the sender

Using simple words in the profile table does not resist against the dictionary attacks. It can be used in a social network between ordinary people, but we


Received message
HF(Sport)||E_S($PK_{j,A}$, HF(Soccer)||E_{AS}(payload,$Pv_{j,A}$))
Check profile table of node B with HF(Sport)
Node B profile attribute
Is there any match between HF(Sport) and G_B
No
Send it to another neighbor
Yes
Use $L_{k,B}$ to decrypt DE_S($PK_{j,A}$, HF(Soccer))
Could decrypt it?
No
Send original message to another neighbor
Yes
Use $PK_{j,A}$ to decrypt DE_{AS}(payload,$Pv_{j,A}$) and read payload of the message

	G_B	L_B
1	…	…
…	…	…
k^{th}	HF(Sport)	HF(Soccer)
…	…	…
N^{th}	…	…

Fig. 3. Process of decryption a message in the receiver

propose to use special codes as $HF(G)$ and $HF(L)$ for particular groups. For example, these attributes can be set as special codes between related groups in the army, and members of the group whose have these codes, are able to read these messages. These groups can be defined by different levels of security. For example, different codes for various groups of soldiers, lieutenant, captain, colonel, general and etc. can be defined in an army. People with a similar code who are members of a group can decrypt messages. Furthermore, people from different groups, according to their duties and degrees, can have some or all of the related code, and achieve more messages. In Fig. 4, different groups in an army are represented, and each square shows a group and its subsets. Members of nodes in a group will have access to all codes in their groups and subset groups (square). Figure 5 illustrates the profile table of different groups.

These codes can be given to the qualified members of a group, and it will be possible they receive the public key and decrypt secret messages. For example, a General has the public key of Colonels, Captain, Lieutenant, and soldiers. Members of every group have the code of its group and subset groups. Furthermore, the public key can be changed frequently, and the messages will be safe against some attacks like the dictionary attacks.

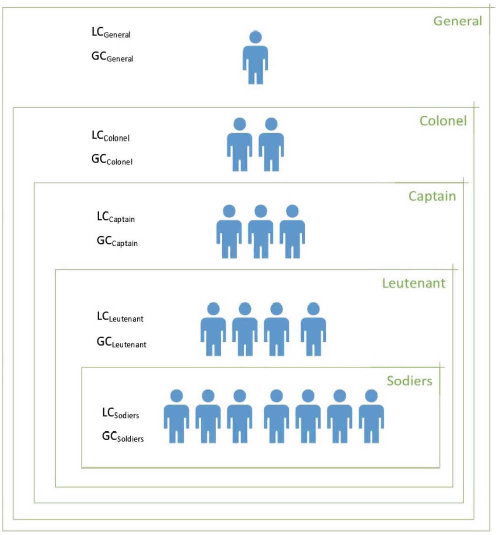


Fig. 4. Different groups with various Limit and General parts in an OppNet

6 Evaluation

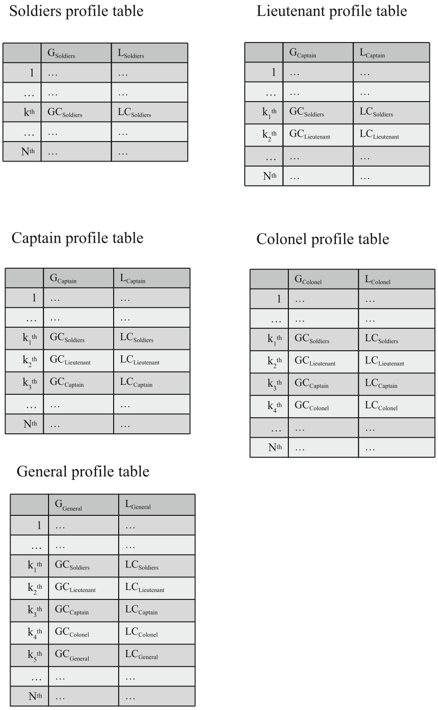
In this section, the security of the proposed encryption method is discussed.

In the proposed algorithm, every node evaluates the trust degree of its neighbors and nodes select a neighbor with the most top degree of reliability to send a message to it. Furthermore, the attributes of the profile table of users are hashed, thus nodes do not access the text of attributes.

Two parts for attributes are considered in order to be more precise and resist against the dictionary attacks. If these attributes are set with special codes (For example 32 character with high security level), intermediate nodes cannot find the public key by the dictionary attack.

An intermediate node compares $HF(G_{j,A})$ of a message with its neighbors General part of the hashed attributes in its routing table and if the answer shows that there is a match, the node can forward this message to the best neighbor to make the message closer to the destination.

From the performance point of view, a sender needs to encrypt twice, two parts of a message and the receiver should decrypt corespondent, but intermediate nodes will not need to encrypt or decrypt a message. Intermediate nodes evaluate just $HF(G_{j,A})$ part of a message with their profile table and route it according to this. It provides a faster process of routing in the intermediate nodes. Furthermore, the process of double encryption of a message can be done for the first time. After a match comparison and connection of two nodes and sharing the public keys, these nodes can use an asymmetric algorithm for encryption of messages in the rest of a communication.



Soldiers profile table

	$G_{Soldiers}$	$L_{Soldiers}$
1	…	…
…	…	…
k^{th}	$GC_{Soldiers}$	$LC_{Soldiers}$
…	…	…
N^{th}	…	…

Lieutenant profile table

	$G_{Captain}$	$L_{Captain}$
1	…	…
…	…	…
k_1^{th}	$GC_{Soldiers}$	$LC_{Soldiers}$
k_2^{th}	$GC_{Lieutenant}$	$LC_{Lieutenant}$
…	…	…
N^{th}	…	…

Captain profile table

	$G_{Captain}$	$L_{Captain}$
1	…	…
…	…	…
k_1^{th}	$GC_{Soldiers}$	$LC_{Soldiers}$
k_2^{th}	$GC_{Lieutenant}$	$LC_{Lieutenant}$
k_3^{th}	$GC_{Captain}$	$LC_{Captain}$
…	…	…
N^{th}	…	…

Colonel profile table

	$G_{Colonel}$	$L_{Colonel}$
1	…	…
…	…	…
k_1^{th}	$GC_{Soldiers}$	$LC_{Soldiers}$
k_2^{th}	$GC_{Lieutenant}$	$LC_{Lieutenant}$
k_3^{th}	$GC_{Captain}$	$LC_{Captain}$
k_4^{th}	$GC_{Colonel}$	$LC_{Colonel}$
…	…	…
N^{th}	…	…

General profile table

	$G_{General}$	$L_{General}$
1	…	…
…	…	…
k_1^{th}	$GC_{Soldiers}$	$LC_{Soldiers}$
k_2^{th}	$GC_{Lieutenant}$	$LC_{Lieutenant}$
k_3^{th}	$GC_{Captain}$	$LC_{Captain}$
k_4^{th}	$GC_{Colonel}$	$LC_{Colonel}$
k_5^{th}	$GC_{General}$	$LC_{General}$
…	…	…
N^{th}	…	…

Fig. 5. Profile table of different groups in an OppNet

In order to protect all communications of a node, we proposed that each node uses a separate private and public keys for different attributes. When node A connects to node B, $PK_{j,A}$ and $PV_{j,A}$ are used which are created for this connection, and when the session is ended, these keys will be expired. In this manner frequently changing of $PK_{j,A}$ and $PV_{j,A}$ is possible.

More than one message is often sent to OppNet. We suggest that three copies of each message have to be created, and each of these copies is sent from a different paths/neighbors. Thus, if a node tries to change one of them when several messages are delivered to the receiver, a mismatch is found, and this can indicate that some parts of a message are changed by an attacker.

To put in a nutshell, in our proposed scheme, messages are not sent as a text, although they are encrypted by senders' private key, and the related public key is shared between some groups of people whose are interested in a specific topic. Intermediate nodes are able to route a message without extracting knowledge about the title of a message or the payload of it.

7 Comparison with Previous Solutions

We compared the proposed algorithm with discussed algorithms in Sect. 2 in Table 3 in the following terms:

Table 3. comparison of proposed algorithm with some other algorithms

	Network knowledge	TTP	Different payload and header decryption	Context based	Content based	Known receiver	Other
MLCE [5]	✓	×	×	×	✓	×	×
ID\policy based [6]	×	✓	✓	✓	×	×	×
Ant based [9]	×	×	×	✓	×	✓	×
PEON [10]	×	✓	×	✓	×	✓	✓
Chaining algorithm [11]	✓	×	×	✓	×	✓	×
Peer-to-peer onion-based [12]	×	×	×	✓	×	✓	×
Group onion-based [13]	✓	×	×	✓	×	✓	×
Digital signature [14]	✓	×	×	✓	×	✓	✓
Proposed algorithm	×	×	✓	✓	✓	×	×

- Is it necessary for nodes to have a limited knowledge about the network topology and position of the next or previous hop?
- Is it necessary for nodes to connect to the third party for sharing their public keys?
- Is header and payload of messages encrypted separately in order to make the process of routing easier for intermediate nodes?
- Does the method cover context based routing?
- Does the method cover content based routing?
- Is it necessary for the sender of a message to have knowledge about the receiver of the message?
- Did the algorithm consider additional security infrastructure like secure side channel?

The check mark sign shows a positive answer to each question and cross sign shows a negative answer to each question in Table 3.

In OppNet, nodes move frequently and often topology of the network is changed. In the majority of algorithms in the literature, nodes need to have a knowledge or at least a limited knowledge about the topology of the network and the position of the neighbors, or they need to have access to a third party for sharing public keys. Therefore, the most of these algorithms are not suitable for OppNet.

In the proposed algorithm, messages are encrypted and public keys are shared without knowledge about the network or the destination of a message and also without considering a third party in the network. Furthermore, proposed algorithm covers both content and context based routing algorithms in OppNet. The payload and header of messages are encrypted separately to make the process of routing easier in the intermediate nodes.

8 Conclusion

In this paper, a new method to maintain confidentiality for Opportunistic networks is presented. A Trust Algorithm is applied and according to that, the priority of messages are defined. Therefore, each node sends the message to a neighbor with a higher degree of trust. For more security, two parts for each attribute in the profile table of a node are taken into consideration, one of them is the general concept of the message and another one is the limited concept of this, it is remarkable that, both of these parts were hashed. The General part of an attribute is used as a header of a message which gives the capability of routing the message to the destination according to it, and another part of an attribute is used as a key for encryption of the public key of a sender. Therefore, only the node that has the similar attribute can decrypt the message. Also, we compared the proposed algorithm with some other algorithms in the state of art.

References

1. Farrell, S., Cahill, V., Geraghty, D., Humphreys, I., McDonald, P.: When TCP breaks: delay-and disruption-tolerant networking. IEEE Internet Comput. **10**(4), 72–78 (2006)
2. Juang, P., Oki, H., Wang, Y., Martonosi, M., Peh, L.S., Rubenstein, D.: Energy-efficient computing for wildlife tracking: design tradeoffs and early experiences with zebranet. ACM SIGARCH Comput. Archit. News **30**(5), 96–107 (2002)
3. Huang, J.-H., Amjad, S., Mishra, S.: CenWits: a sensor-based loosely coupled search and rescue system using witnesses, pp. 180–191 (2005)
4. Detweiller, C., Vasilescu, I., Rus, D.: An underwater sensor network with dual communications, sensing, and mobility, pp. 1–6 (2007)
5. Shikfa, A., Onen, M., Molva, R.: Privacy in content-based opportunistic networks, pp. 832–837 (2009)
6. Shikfa, A., Önen, M., Molva, R.: Privacy in context-based and epidemic forwarding, pp. 1–7 (2009)
7. Hegland, A.M., Winjum, E., Mjolsnes, S.F., Rong, C., Kure, O., Spilling, P.: A survey of key management in ad hoc networks. IEEE Commun. Surv. Tutor. **8**(3), 48–66 (2006)

8. Attar, A., Tang, H., Vasilakos, A.V., Yu, F.R., Leung, V.C.: A survey of security challenges in cognitive radio networks: solutions and future research directions. Proc. IEEE **100**(12), 3172–3186 (2012)
9. Memarmoshrefi, P., Seibel, R., Hogrefe, D.: Autonomous ant-based public key authentication mechanism for mobile ad-hoc networks. Mob. Netw. Appl. **21**(1), 149–160 (2016)
10. Le, Z., Vakde, G., Wright, M.: PEON: privacy-enhanced opportunistic networks with applications in assistive environments. In: 2nd International Conference on Pervasive Technologies Related to Assistive Environments, pp. 76–83 (2009)
11. Cabaniss, R., Kumar, V., Madria, S.: Multi-party encryption (MPE): secure communications in delay tolerant networks. Wirel. Netw. **21**(4), 1243–1258 (2015)
12. Palmieri, P., Pouwelse, J.: Key management for onion routing in a true peer to peer setting. In: Yoshida, M., Mouri, K. (eds.) IWSEC 2014. LNCS, vol. 8639, pp. 62–71. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-09843-2_5
13. Sakai, K., Sun, M.-T., Ku, W.-S., Wu, J., Alanazi, F.S.: Performance and security analyses of onion-based anonymous routing for delay tolerant networks. IEEE Trans. Mob. Comput. **16**(12), 3473–3487 (2017)
14. de Diogo, A., Albini, L.C.P.: Fully distributed public key management through digital signature chains for delay and disrupt tolerant networks. In: 13th International Conference on Mobile Ad Hoc and Sensor Systems, pp. 316–324 (2016)
15. Rashidibajgan, S.: A trust structure for detection of sybil attacks in opportunistic networks, pp. 347–351 (2016)

Wearable IoT Security and Privacy: A Review from Technology and Policy Perspective

Onyeka D'Mello[1], Mathilde Gelin[1], Fatma Ben Khelil[1], Rojen Erik Surek[1], and Huihui Chi[1,2](✉)

[1] MSc Big Data and Business Analytics, ESCP Europe, Paris, France
{onyeka.dmello,mathilde.gelin,fatma.ben_khelil,rojen_erik.suerek, huihui.chi}@edu.escpeurope.eu
[2] Department of Information and Operations Management,
ESCP Europe, Paris, France

Abstract. The continuing increase in the number of Internet of Things (IoT) devices around the world calls for the need to assess privacy and security vulnerabilities of IoT devices. In this paper, we discuss the extent to which individuals and organizations have utilized the IoT-enabled devices to connect and share data. We also explain the different types of security loopholes that need urgent attention along with other ethical issues that arise from IoT devices. While major application of the IoT is its incorporation into wearable technology, we review its current practices and implications. Moreover, this paper also highlights some of the legal policies and regulations, their values, and challenges regarding data privacy. Finally, we discuss various data analytics solutions for cyber-security coupled with their value and the challenges.

Keywords: Internet of Things (IoT) · Cyber-security · Wearables
Security and privacy concerns · Regulations · Policies
Data analytics · Predictive analytics · Big data

1 Introduction

The recent years have seen an increase in the number of smart devices that can connect to the Internet with ease. This paper focuses on the study of the Internet of Things abbreviated as IoT in the 21st century. With the ever-expanding population of IoT devices, the need to address their susceptible security becomes crucial and demanding.

The research on the Internet of Things (IoT) is an interesting subject because it is a common phenomena in today's world. Almost everyone in the modern society has access to smart digital devices such as smart phones, smart TVs, smart watches, and smart technologically-driven cities among others. Furthermore, the sub-topics of legal policies and regulations surrounding IoT security is also interesting because the society needs to maintain their legal and moral standards in this digital age [17].

R. Doss et al. (Eds.): FNSS 2018, CCIS 878, pp. 162–177, 2018.
https://doi.org/10.1007/978-3-319-94421-0_13

Previous literature shows that the availability and access to IP-enabled devices (IoT) continue increasing exponentially every year. Additionally, it shows that there is a need to address the susceptible security concerns that come alongside the IoT devices [30]. Our research highlights various policy regulations around the world that govern the use of the smart technologies. However, previous research conducted in the past provides limited information addressing the shortcomings of challenges associated with data analytics and policy regulations in improving the security of data and devices in the digital era.

Using a sample of the companies that use data analytics and selected IoT devices, it has been analyzed that there is a dire need to develop policy regulations and quantitative techniques to improve IoT security. There exist some challenges when formulating and implementing policies for improving the security of the IoT devices.

This research contributes to the literature in the field of Information and Communication Technology by showing that both data analytics and policy regulations play a critical role in the improvement of security across IoT devices. Future studies need to address the empirical analysis of all the existing guidelines relating to data analytics and privacy concerns around the world from 2017 to 2021[1].

The remainder of the paper is organized as follows. In Sect. 2, we describe the research methodology in detail. In Sect. 3, we introduce the wearable IoT and related devices. We emphasize security and privacy issues in IoT based on different types of attacks in Sect. 4. In Sect. 5, we provide a summary of this paper and conclude.

2 Research Methodology

We present a comprehensive review of published research on wearable Iot security and privacy issues. To operationalize this, we searched for published papers in international peer-reviewed journals or books in electronic bibliographical sources mainly by keywords or combination of keywords such as wearable Internet of Things (IoT), Radio Frequency Identification (RFID), wearable devices, security, privacy. We then expanded our search by using additional keywords obtained from the results of our initial search like electroactive fabrics, cyberattack. This resulted in 40 papers after filtering by categories, topic relevance, time of publication, and contributions.

The distribution of papers across journals shows that the papers were mainly published in journals that cover interdisciplinary topics such as Decision Support Systems, European Journal of Information Systems, and Operations Research. The distribution of published papers across years 2001–2018 is shown in Fig. 1, indicating the increasing attention paid to this general area. To the best of our knowledge, this paper is the first of its kind to simultaneously review the security and privacy issues in wearable IoT from the technology and policy perspective.

[1] Morgan, S. (2017), Top 5 Cybersecurity facts, figures and statistics for 2017, CSO.

3 Wearable IoT, Technology and Devices

'Internet of Things', commonly referred to as IoT, had its name coined in 1998 by Ashton of Procter & Gamble who described it to be a network of IP-enabled devices with the ability to connect and exchange data [1]. Ranging from everything "SMART" - smart homes, smart cars, smart watches, smart cities, IoT is assumed to consist of approximately 20.8 billion connected devices by the end of 2020[2]. Subsequently, IoT devices that can be worn by individuals on their bodies are referred to as wearable devices, although it can go around by several names.

To put the vulnerability of IoT devices into perspective, recent years have seen dozens of medical devices potentially vulnerable to cyber-attack threats by researchers. Millions of smart-TVs are vulnerable to click fraud, bot-nets, data theft and even ransomware. In the world of smart cars, Fiat Chrysler recalled 1.4 million vehicles after researchers demonstrated a proof-of-concept attack where they managed to take control of the vehicle remotely. In the UK, attackers managed to hack key-less entry systems to steal cars. In retrospect and according to recent reports, it has been predicted that cyber-crime damage costs are estimated to hit a total of $6 trillion annually by 2021. Furthermore, cyber-security investment and spending are to exceed $1 trillion (see footnote 2) in the next years.

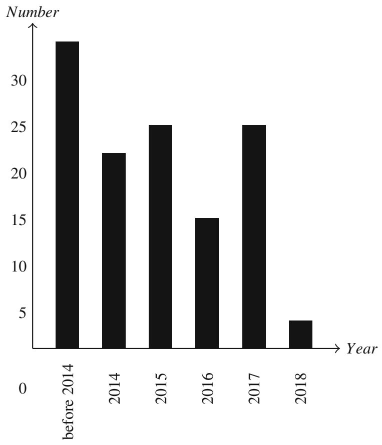


Fig. 1. Distribution of number of publications by year

[2] Wood, P. (2016). 2016 Symantec Internet Security Threat Report, https://www.symantec.com.

3.1 Wearable IoT

Wearable technology is often touted as one of the greatest applications of the IoT with good reason. Wearable technology has the potential to transform the way people live. So what is 'Wearable Technology' one may ask? Wearable technology, which goes around by several names such as 'Wearable Electronics', 'Wearable Connected Devices', 'Wearable IoT (WIoT)' or just simply as 'Wearables', includes small electronics devices that people can wear on their bodies with ease of comfort [36,37]. In the broadest sense, any computer device that is carried with a person to assist them could conceivably be called a 'Wearable'. In particular, glasses, jewelry, watches, head-bands, contact lenses and even clothing. Alternatively, one might also come across a more invasive form of this concept as in the case of implanted devices used to measure electrical activities from the body [38]. Coker (2015)[3] described few examples of Wearable IoT devices (some currently being developed) that can be implanted inside a human body (Table 1):

Table 1. Wearable IoT applications on human being

Applications
Implantable Birth Control
Cyber-Pills
Smart Tattoos
Verified Self
Smart Monitoring and Healing Chips

Ultimately, whether a device is worn on or incorporated into the body, the purpose of wearable technology is to create constant, convenient, seamless, portable, and mostly hands-free access to electronics and computers [23].

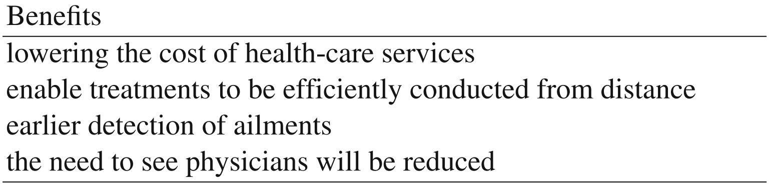
The implications and uses of wearable technology are far-reaching and can influence the fields of health and medicine, fitness, aging, disabilities, education, transportation, enterprise, finance, gaming and music [32]. However, it is in the fields of health-care, medicine, and fitness where wearable technology potentially has its greatest influence. There are beliefs that wearable devices, over the next 10 years, will transform healthcare sector by [8] (Table 2):

The most successful wearable devices as of today are smart watches and health and fitness trackers [33,35]. In fact, over 170 million units of wearable wrist-wear devices are forecast to be shipped in 2020. According to another forecast, sales of smart watches alone are going to reach 141 million units worldwide with Apple's watchOS being the most used smart wrist wear operating system to be used[4].

[3] Top Five Implantable Wearables. Technowize. Retrieved 25 November 2017, from https://www.technowize.com/top-five-implantable-wearables/.

[4] Gordon, K. Statistics & Facts on Wearable Technology. The Statistical Technology. Retrieved 2017-11-18, from https://www.statista.com/topics/1556/wearable-technology/.

Table 2. Benefits of wearable devices in healthcare



Benefits
lowering the cost of health-care services
enable treatments to be efficiently conducted from distance
earlier detection of ailments
the need to see physicians will be reduced

Wearable technology usages can be broadly categorized into two major categories as shown in Table 3[5]:

The market for wearable technology looks promising as the number of connected Wearable devices worldwide is expected to grow from an estimate of 325 million in 2016 to over 830 million in 2020. A little over than 2.5 times in a span of only 4 years!

3.2 Devices to Enhance User Experience

Visitors of the Walt Disney World in the US can now encounter the MyMagic+ program. MyMagic+ incorporates a wearable MagicBand that uses a number of technologies, all designed to enhance the user's experience and provide useful data to Disney. The MagicBand can connect to a number of systems in the theme park and can assist visitors to make reservations for rides electronically in order to avoid long waiting times using the MagicBand. They can also purchase on-site meals which can be electronically charged to their Disney Hotel room using this band. This largely improves user experience and in return the profits for the business. The MagicBand allows Disney to easily track the movements and actions of park visitors so that staff and services can be efficiently allocated to meet emerging needs.

The Disney example depicts how powerful Wearables can be in a controlled space where number of variables are limited. Speaking of tracking movements, some companies are inventing Wearable Technology to track a companion animal's movements and health, and even to track the activities of their infants and pre-schooling children. It is often debatable if one's privacy is being compromised in exchange for an improved user experience or the ease of life [4].

3.3 Wearable Electroactive Fabrics and Bio-monitoring Devices

Some of wearable devices are capable of recording biomechanical variables from its users. The system included in the wearable device is able to record the vital signs and movement of its user. Research has made improvement in the development of smart textiles: devices that are capable of recording several human vital

[5] "Understanding Wearable Technology – Aspencor Tech". Aspencor Tech. Aspencor Tech. from http://medgizmo.info/news/understanding-wearable-technology.

Table 3. Two categories of wearable technology usages

Personal usage	Business usage
A fitness/sport tracker	A treatment for hearing impairments
A gauge for alertness and energy levels	Remote treatment of speech and voice disorders such as those in patients with Parkinson's disease
A fashion statement	Synchronize data and communication from other gadgets
A navigation tool	Specific health issue monitoring, such as stress management
A communication gadget	Improve user experience
A media device	

signs and wearable motion-capture systems. The use of those devices impacts important tool for promoting sustainable development and progress in different fields such as healthcare, ergonomics, art and sport [3,24].

3.4 Self-tracking Technologies in the Workplace

Companies are willing to use wearable devices at the workplace, in order to increase the productivity by increasing the wellness and health of their employee, and also in order to measure and quantify their behavior and performance[6]. Wearable devices can take many forms such as armbands, badges, rings and smart watches, using Bluetooth, infrared sensors and accelerometers. In the workplaces' use of these devices, companies store data on their employee regarding stress level, heart rate, physical activity and body temperature, altogether becoming great implication for the company due to the huge amount of data created per day and the privacy of the information [25]. The use of wearable devices is raising questions about legal, privacy and data protection issues. Because this ability of gathering data is new, it is also unregulated. But most important thing about insecurity comes from that employees, in this specific case, use their own devices at work: data security standards are not respected [29].

3.5 Wearable Medical Devices

Wearable medical devices, such as continuous health monitoring devices for individuals, have generated a vast quantity of data. There has been a proposition on an IoT architecture that intends to store and process the data for healthcare applications [14,26]. More concretely, the architecture that has been proposed

[6] Self-tracking technologies in the workplace: Quantifying health, behavior and productivity, Human Resource Management International Digest, 25(5), 10–12, Retrieved from http://www.emeraldinsight.com/.

cover MetaFog-Redirection (MF-R) and Grouping & Choosing (GC). In this same proposition, logistic regression has been conducted based on prior records from a certain heart disease database and data retrieved from health sensors on patients. Based on this regression analysis a prediction model can be created that uses the current body sensor health data of blood pressure, heart rate and blood sugar level in order to predict the risk of heart disease.

Implanted medical Wearables have diversified uses too. Health professionals are adopting the cyber-implant technology among their patients in order to track diseases in real-time. These devices are fed and they retrieve health data directly into smart phones. An example of this is the 'Bionic Pancreas' which is used to monitor blood-sugar levels for diabetics (see footnote 3). Cyber-Pills with microprocessors are being developed by British Researchers which communicate directly from inside the body to a smart phone to help health specialists monitor the users regular medication intake and its possible side-effects.

Slender Smart Tattoos made of computer fibers are being used to track body functions and processes. Individuals can also use their fingers to unlock or enter codes with the aid of an NFC chip inserted into their fingertips using tattoo-like procedures.

4 Security and Privacy Issues in IoT

Our IoT world represents a danger to users because of high cyber-risk. Users are sharing highly personal information as in the example of home devices (alarms, clocks, lights, doors and garage openers). They can be extremely dangerous because the crucial information embedded and shared via IoT devices represents an attraction for hackers: since in every perfect IoT ecosystem, there is a danger in security [21]. Other sectors as media and telecom technologies are targeted by hackers and lead to a real combat against cyber-risks due to the high value of the data shared and created. The other industries that are prone to being hacked are healthcare and life sciences, infrastructures and smart cities, transports and urban mobility and finally industrial systems and sensors [39].

Company leaders try their best to take actions against threats and their impacts at three levels of an organization: they prevent and anticipate IoT related cyber-threats before they take hold; they monitor and neutralize threats that are already operating, and finally, they restore regular operations as soon as possible after treat.

Organizations need to find a balance between cyber-risk management and innovation. This means that the use of IoT and the increased use of information not only increase the possibilities of creating value for the organization, but they also increase the possibilities of cyber-risks. When data is overprotected, it hinders innovation and creation of values but at the same time, if data is left open and unprotected, this would leave the organization vulnerable to cyber-risks.

There are no global risk standards governing the IoT at the moment because of the novelty of IoT; however, it does not mean that organizations between them - public or private, share awareness and operate strategically and cooperatively to ensure the immense value of data. There is a danger of security breach because the shared responsibility does not always work. Saif voiced that IoT solutions need to be implemented in such a way that they blend organization-specific operational capabilities with multi-layered cyber-risk management techniques [21].

Li and Xu [13] envisaged IoT as a multilayer network and they call the intelligent tags and sensors the “sensing layer” which could be devices such as RFID tags, readers, WSNs, BLE devices that are acquiring the information of the devices and/or their immediate environment. In implementing the sensing layer, organizations will need to take diverse security threats and vulnerabilities into account; more specifically unauthorized access, selfish threats, spoofing attacks, malicious code, DoS, transmission threats and routing attack [28]. In order to secure users, the authors proposed some measures to mitigate security risks: (i) implementing security standard for IoT and ensuring that all devices are produced by meeting specific security standards; (ii) building a trustworthy data sensing system and reviewing the security of all devices; (iii) forensically identifying and tracing the source of users; (iv) and finally that software or firmware at IoT end-node should be securely designed.

Privacy Concerns Related to Big Data: The volume of data in the world is increasing drastically. By 2021, Big Data will be of worth $66.9 billion which increases concerns about privacy and security of the data[7]. Currently, the number of cyber-crime victims is increasing on a daily basis, and people are urging the government to undertake actions to fight against these threats in order to provide full trust in the utilizing and sharing of their data. For example, the more data is contained in a single source, the easier it is to be cyber-hacked. Companies need to accept the greater responsibilities for personal information they have on people and could use third-party providers to help them store their data in clouds and other areas [34].

4.1 Different Types of Cyber-Attacks on IoT

Distributed Denial of Service (DDoS) Attacks. Nowadays, technology enables industries to use IoT embedded into small devices, allowing the integration of physical things into an information network. IoT faces a lot of challenges due to its low power, low processing, and low memory because of its small-sized housing. A multitude of attacks can impact an IoT network and Denial of Service attacks (DoS) is known to be the most sought attacking method. An attack is referred to as ‘Distributed’ Denial of Service attack when the attack is diffused from different sources (DDoS) [22]. These attacks can block usage of the IoT device for the users and can drive network resources or consumption of the bandwidth to be unavailable or modified. For the healthy functionality of IoT, data needs to be confidential during its transmission; it needs to maintain its integrity because it should be the same, sent as received; it needs to be available for the users, and lastly, it needs to be authentic with the right identity claimed.

[7] McCabe, B. Privacy concerns about the monetization of Big Data, Linked In.

The five DDoS Attack types are respectively called the '*UDP flood*', which leads to the inaccessibility of the target host resources; the '*ICM/PING flood*', which leads to a significant overall system slowdown; the '*SYN flood*' and the '*Ping of Death*', both of which lead to a denial of service; and the '*Zero-Day DDoS*', which cannot be described because it has never been seen before. DDoS attacks globally change the expected functionality of the IoT and can lead to several adverse impacts on the users.

IoT is vulnerable to DDoS attacks even from an architectural perspective. The architecture of IoT is divided into three layers called the *Perception layer*, which collects ubiquitous data from the physical environment; the *Network layer*, which processes the data; and the *Application layer*, which contains the business logic for the user. On each of those layers, different varieties of DDoS attacks can befall. On the Perception Layer, the main reader technology RFID can be hacked. It will be unable to communicate with the reader, or completely disabled. It could also lose its authentication capability and synchronization between the system and the tag. On the Network Layer, attacks can disrupt the authentication availability, fake replicate request instead of original ones, consume enormous amounts of the victim's resources, and amplify the traffic for breakdown. On Application Layer, attacks can create infinite loopholes to disable the accessibility of network resource and create infinite waiting time for reply, along with communication paths that replay data packets or insert infected data packets [31].

Social Engineering. Social Engineering can be perceived as an act of manipulation of people through their personal information but more globally it is an art or a science of skilfully maneuvering human beings to take action in some aspects of their lives [7].

There are different types of social engineering which could be classified as friendly or malicious. The first type, which is the most malicious, is the Hackers. Because of the complexity of today's software, hackers are turning towards social engineering skills, mixed with the use of hardware and personal skills. The second type is quite similar but represents the friendly approach: the Penetration Testers. These are individuals who are meant to follow and think like a hacker in order to disrupt the client's security by mimicking actions that of a hacker. Spies, identity thieves, disgruntled employees, scam artists are considered as types of social engineering which can cause harm to other people. Executive recruiters, sales-people, governments, doctors, psychologists, and lawyers are also other types of social engineer, which are not meant to harm people.

The basic goal of malicious social engineering is the same as hacking in general: to gain unauthorized access to systems or information in order to commit fraud, network intrusion, industrial espionage, identity theft, or simply to disrupt the system or network [6]. The Internet boom has its share of industrial engineering attacks as in start-ups as well, but attacks generally focus on larger entities.

Man in the Middle (MITM) Attacks. This type of attack encompasses the concept of intercepting (read, insert and modify) legitimate communications between two separate users or IP-connected-systems by a middle agent - the hacker [5]. The hacker uses an amalgamation of Eavesdropping and Alteration techniques to create a web of deceit and tricks the two systems into thinking they are communicating with each other. In this scenario, the hacker has control of the original communication and transmits messages to the two separate nodes [10]. According to a Europol news reported in 2015, 49 suspects were arrested for performing MITM attacks to sniff out and intercept payment requests from emails. Investigations uncovered international fraud totaling 6 million euros[8].

For the MITM attack to work, the hacker would need to find an unsecured or poorly secured WIFI router. Then he injects malware into the connected-device, which installs itself into the victim's web-browser without the victim's knowledge. This malware can then record and route all information being exchanged between the victim and specific targeted websites (e.g. financial institutions) to the hacker's computer. According to the McAfee Threat Reports of 2014, MITM attacks comprise 66% of total Top Network Attacks [5]. The MITM method of cyber-attack is gaining popularity among cyber-thieves due to its ease of execution and its arduous nature of being detected as these attacks can be accomplished without any trails left behind for the breach.

In Meyer's paper, 'A man-in-the-middle attack on UMTS' [16], he displayed a man-in-the-middle attack on the Universal Mobile Telecommunication Standard (UMTS), one of the recently developing 3G portable advances. The assault enabled an interloper to mimic a legitimate GSM base station to an UMTS supporter paying little respect to the way that UMTS confirmation and key understanding are utilized. Accordingly, a gatecrasher could listen stealthily on all versatile station-started traffic. Since the UMTS standard requires shared confirmation between the portable station and the system, so far UMTS systems were thought to be secured against man-in-the-middle assaults. The system confirmation characterized in the UMTS standard relies upon both the legitimacy of the validation token and the honesty insurance of the consequent security mode command. Meyer demonstrated that both of these instruments are essential keeping the end goal to keep a man-in-the-middle assault in mind. As an outcome he demonstrated that an assailant can mount a pantomime assault since GSM base stations do not bolster trustworthiness insurance and possible victims to the attack are all mobile stations that support the UTRAN and the GSM air interface simultaneously.

Data and Identity Theft. Today in the US alone, there are 25 connected IoT devices per 100 inhabitants. It is safe to say that people are prepared to accept any reality as long as it is presented to them in a digitalized manner. In addition, people readily accept information from smart devices as a fact of life.

[8] Abel, R. (2015, June 10). Europol arrest dozens for a scam that laundered six million euro, SC Media. https://www.scmagazine.com/europol-arrest-dozens-for-a-scam-that-laundered-six-million-euro/article/534058/.

However, it is believed that the biggest threat to the success of IoT devices lies in ID theft-related crimes [40].

ID theft is the action of unauthorized use of personal information (which is stored and used in an array of digital forms) by another individual for various gains. These gains range from espionage, revenge, terrorism, illegal immigration or assuming a new identity to evade criminal charges. The nature of personal data are names, addresses social security numbers, date of birth, driver's licenses, passport numbers, and financial data. Various frauds range from: fraudulent unemployment claims, fraudulent tax returns, fraudulent loans, home equity fraud and payment card fraud. Original users can also endure the burden of increased loan interest rates; they can suffer involuntary payment with credit card fraud, and they can be denied from utility services, civil suits or criminal investigation [20].

Nonetheless, whatever the underlying objective maybe, it all boils down to some sort of financial gain for the thief. Since IoT's foundation is built on identity related services and (hence) any communication between devices is therefore based on the same identity, ID theft operated via a digital channel could easily be categorized as a cyber-crime. Vidalis and Angelopoulou propose a vulnerability assessment model that attempts to understand how an environment can be influenced by this type of attack. This can be established by the use of Vulnerability Trees to measure how the environment can be affected by the introduction of smart devices. This further can help in making appropriate and informed decisions in terms of management of such crimes. The user is known to constitute the biggest and least complex vulnerability of a system [27]. As a response to these breaches, governments all around the world have enhanced laws that require organizations to notify individuals when their own information has been hacked.

Recently, the European Union has launched the General Data Protection Regulation (GDPR) which will force companies to protect the personal information that they have on each individual. The policies' goal is to protect natural persons with regard to the processing of personal data and on the free movement of such data[9].

Botnets Attacks. Mobile botnet attacks are systems that are combined to distribute malware. They are used by criminals to exploit online-banking data and steal private information. The botnet operators control them via Command-and-Control-Servers. Mobile devices have their own constraints such as limited processing, less data storage capabilities and heterogeneity of operating systems (OS) (Android, Apple, Windows etc.), that restricts the security solutions to be programmed efficiently. Botnet is a network of compromised machines. The aim of botmaster is to disturb true blue administrations over the Internet or cheat private data. Botnets are advancing as a genuine danger towards focusing on cell phone gadgets. The motive of this attack is somewhat similar to that of

[9] Reform of Data EU Protection Rules, Building a European Area of Justice. Retrieved from http://ec.europa.eu/justice/data-protection/reform/index_en.htm.

traditional botnet attacks to access the assets, translate substance of portable client gadget and exchange control to the botnet initiator. In the long run, this programmer will probably perform pernicious and unapproved exercises including illicit telephone calls, ceasing control panel, sending emails, initialization of worm code and unauthorized file access or photos. 'Andbot' is a mobile bot, which utilizes URL transition and it is considered as a stealthy, minimal effort, and flexible bot, which utilizes botmaster for unlawful in mobile environment.

4.2 Three Main Types of Attacks on Wearable Devices

Wearables can fall victim to an array of security breaches. Marrington et al. broadly categorize these attacks into the categories below [15]:

Unauthorized Access to Wearable Devices. This is the classic case of the Sinkhole Attack where unauthorized users gain illegal access to the wearer's wearable device adversely affecting their privacy. The concept of this attack requires a base station (e.g. a health monitoring application on a user's smart phone that is connected to the operating system of the fitness tracking device worn by the user) and a Wireless Sensor Network (WSN). The small nodes that make up a WSN sense and send data to the base station. In the Sinkhole Attack, the hacker infiltrates a node(s) (preferably one that is closer to the base station, rather than all the nodes in the network), which causes the compromised node to attract all traffic from its neighboring nodes using fake routing information [11]. All packets of data then pass through the infiltrated node before reaching the base. This hinders the base station's ability to receive complete and unaltered data.

This can have adverse effects on people and organizations who work with health-related wearable devices. Hospitals and other medical institutions rely on wearables to collect patient's health information and track their behavioral habits; compromised medical data can jeopardize the wearer's physical safety.

Attacking the Wearable Device Availability. All wearable devices depend on inbuilt battery-packs for their sustainability and operation. This dependency gives rise to the possibility of the Denial Of Service (DoS) attacks. As mentioned above, this type of attack encapsulates the concept wherein the Wearables' OS is overwhelmed by malicious requests brought on by the attacker. This ultimately results in system crashes and draining the device's battery.

FitBit allows users to automatically upload its data to the user's online social networking account on a daily basis. This enables hackers to intercept data reported by FitBit to launch the DoS attack. To prove this, Rahman et al. [18] have built FiteBite which is a suite of tools that exploits vulnerabilities in the FitBit. The FitBit authorizes the dummy hacker to continuously query the victim FitBit (initially once every 15 min and subsequently on an average of 4 times per minute) in its vicinity, hence draining the FitBit's battery at an alarmingly fast rate. In order to avoid suspicions, the FiteBit uploaded the victim

FitBit's data into the web server once every 15 min. Rahman et al. concluded via their experiment, that during the attack-free mode, the victim FitBit's battery lasted for 29 days. In the 15 min upload mode, the battery lasted for 7 days and 18 h whereas, in the attack mode, the battery lasted for only 32.71 h (just a little over 1 day). This summarized that FitBit drained its battery 21 times faster prior to the Battery-Draining-Denial-of-Service Attack on the test.

False Data Injection on Wearables. A false data injection attack implies that data contained, and transmitted by a wearable device is forged [15]. In these attacks, the hacker may target Internet traffic as a point of attack. Wearable devices are made to transmit data to a central database using Internet protocols. In the case of transmission over the Internet attacks, the hacker modifies data transmitted over the Internet protocols and injects modified data, which eventually reflects on the target website [19].

Another exploitation of weak communication channels relates to attacks carried over Bluetooth protocols. In some instances, the hacker may also target data transmitted over WIFI protocols. The attacker can pair a device to the wearable one and avoid authentication in subsequent pairings when using Bluetooth and WIFI technologies [2]. Such a system can act as a conduit for the attacker to overwrite information transmitted between the wearable device and the target recipient, which he subsequently uses to inject false data.

Physical attacks can also facilitate data injection in wearable devices. In physical attacks, the hacker records data that has not been performed by the legitimate owner of the wearable device. In such cases, the wearable device records fabricated facts on the memory component of the device [12].

Hackers can also exploit vulnerabilities in application functions. For example, the failure of a developer to use HTTPS for application functions creates a vulnerable point for hackers to exploit. It was noted that the author fails to verify data contained in HTTPS POST requests, which are often used to upload data over the internet [9].

5 Conclusion

The accelerating emergence of IoT devices has resulted in a vast quantity of sensitive data entering the digital sphere. Thus, all this data are subjected to the risk of unwarranted infringements. Organizations are more likely to identify security incident earlier if they utilize big data cyber-security data analytics. However, due to the volume of abundant data that needs to be analyzed it is still highly challenging. Hence, this requires the usage of analytics solutions that can scale to the huge storage, memory and computation requirements.

Machine learning applied to security data and user behavioral analytic (UBA) are presented as the most promising methods of data analytics. Together with these technologies, measures should also be taken to have access to skilled labor to conduct statistical analysis to get valuable insights. However, there are a lack of people who can perform advanced degree analytics. A key approach for organizations and firms to improve detection of security threats is to utilize readily available frameworks such as Apache Hadoop and inexpensive hardware, which enable the user to collect, store and analyze huge amounts of security data across the whole enterprise in real time.

If the current frontier of cyber-security is predictive analytics, the next one involves automated actions. Often organizations want to investigate problems identified by analytics before taking corrective action, which means that the most effective cyber-security environments will be complex hybrids of human and machine intelligence. The combination of automated and analytics-driven alerts and human interventions will be extremely important for effective security.

Current worldwide policies and regulations related to IoT devices and data protection are insufficient. So far, worldwide organizations are using some guidelines, e.g. the OECD guidelines on the Protection of Privacy and Trans-Border Flows of personal Data or the guidelines of the association Online Trust Alliance. The Federal Trade Commission and the Department for Homeland Security in the United States have also only given non-binding guidelines that cover IoT devices and associated data. In the EU, the General Data Protection Regulation (2016/679) will enforce regulations to device manufacturers and provide a wider data protection for the consumer starting in 2018. A similar forcing regulation is necessary in United States, which is a major and international actor in data creation. This legal discrepancy needs to be eliminated since legal problems can arise when data about a person or entity is transmitted through different jurisdictions with dissimilar data protection laws. The discrepancies on the practices of how to respect security and data protection between organizations, within the same country, also need to be eradicated.

References

1. Ashton, K.: That Internet of Things thing. RFID J. **22**(7), 97–114 (2009)
2. Ching, K.W., Singh, M.M.: Wearable technology devices security and privacy vulnerability analysis. Int. J. Netw. Secur. Appl. **8**, 19–30 (2016)
3. De Rossi, D.: Electroactive fabrics and wearable biomonitoring devices. Autex Res. J. **3**(4), 6 (2003)
4. Fernandez, P.: Wearable technology: beyond augmented reality. Libr. Hi Tech News **31**(9), (2014)
5. Gangan, S.: A review of man-in-the-middle attacks. arXiv preprint arXiv:1504.02115 (2015)
6. Granger, S.: Social engineering fundamentals, part I: hacker tactics. Secur. Focus **18** (2001)
7. Hadnagy, C.: Social Engineering: The Art of Human Hacking. Wiley, Hoboken (2010)
8. Hiremath, S., Yang, G., Mankodiya, K.: Wearable Internet of Things: concept, architectural components and promises for person-centered healthcare. In: 2014 EAI 4th International Conference on Wireless Mobile Communication and Healthcare (Mobihealth), pp. 304–307. IEEE (2014)
9. Hilts, A., Parsons, C., Knockel, J.: Every step you fake: a comparative analysis of fitness tracker privacy and security. Open Eff. Rep. **76** (2016)

10. Hossain, M.M., Fotouhi, M., Hasan, R.: Towards an analysis of security issues, challenges, and open problems in the Internet of Things. In: IEEE World Congress, pp. 21–28 (2015)
11. Kibirige, G.W., Sanga, C.: A survey on detection of sinkhole attack in wireless sensor network. arXiv preprint arXiv:1505.01941 (2015)
12. Kim, D., Park, S., Choi, K., Kim, Y.: BurnFit: analyzing and exploiting wearable devices. In: Kim, H., Choi, D. (eds.) WISA 2015. LNCS, vol. 9503, pp. 227–239. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-31875-2_19
13. Li, S., Xu, L.: Securing the Internet of Things. Syngress, Rockland (2017)
14. Manogaran, G., Lopez, D., Thota, C., Abbas, K.M., Pyne, S., Sundarasekar, R.: Big data analytics in healthcare Internet of Things. In: Qudrat-Ullah, H., Tsasis, P. (eds.) Innovative Healthcare Systems for the 21st Century. UCS, pp. 263–284. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-55774-8_10
15. Marrington, A., Kerr, D., Gammack, J.: Managing Security Issues and the Hidden Dangers of Wearable Technologies, 1st edn, pp. 21–22. IGI Publishing, Hershey (2016)
16. Meyer, U., Wetzel, S.: A man-in-the-middle attack on UMTS. In: Proceedings of the 3rd ACM Workshop on Wireless Security, pp. 90–97. ACM, October 2004
17. Piramuthu, S., Zhou, W.: RFID and Sensor Network Automation in the Food Industry: Ensuring Quality and Safety Through Supply Chain Visibility. Wiley, Hoboken (2016)
18. Rahman, M., Carbunar, B., Banik, M.: Fit and vulnerable: attacks and defenses for a health monitoring device. arXiv preprint arXiv:1304.5672 (2013)
19. Rieck, J.: Attacks on fitness trackers revisited: a case-study of unfit firmware security. arXiv preprint arXiv:1604.03313 (2016)
20. Romanosky, S., Acquisti, A., Sharp, R.: Data breaches and identity theft: when is mandatory disclosure optimal? (2010)
21. Saif, I.: Cyber Risk in an Internet of Things World. Deloitte, New York (2017)
22. Sonar, K., Upadhyay, H.: A survey: DDOS attack on Internet of Things. Int. J. Eng. Res. Dev. **10**(11), 58–63 (2014)
23. Tehrani, K., Michael, A.: Wearable technology and wearable devices: everything you need to know. Wearable Devices Mag. (2014)
24. Thibaud, M., Chi, H., Zhou, W., Piramuthu, S.: Internet of Things (IoT) in high-risk environment, health and safety (EHS) industries: a comprehensive review. Decis. Support Syst. **108**, 79–95 (2018)
25. Townsend, M., Le Quoc, T., Kapoor, G., Hu, H., Zhou, W., Piramuthu, S.: Real-time business data acquisition: how frequent is frequent enough? Inf. Manag. **55**, 422–429 (2017)
26. Tu, Y.J., Zhou, W., Piramuthu, S.: Identifying RFID-embedded objects in pervasive healthcare applications. Decis. Support Syst. **46**(2), 586–593 (2009)
27. Vidalis, S., Angelopoulou, O.: Assessing identity theft in the Internet of Things. IT Converg. Pract. (INPRA) **2**(1), 15–21 (2014)
28. Wang, Z., Hu, H., Zhou, W.: RFID enabled knowledge-based precast construction supply chain. Comput.-Aided Civil Infrastruct. Eng. **32**, 499–514 (2017)
29. Zhou, W.: RFID and item-level information visibility. Eur. J. Oper. Res. **198**(1), 252–258 (2009)
30. Zhou, W., Kapoor, G., Piramuthu, S.: RFID-enabled item-level product information revelation. Eur. J. Inf. Syst. **18**(6), 570–577 (2009)

31. Zhou, W., Yoon, E.J., Piramuthu, S.: Varying levels of RFID tag ownership in supply chains. In: Meersman, R., Dillon, T., Herrero, P. (eds.) OTM 2011. LNCS, vol. 7046, pp. 228–235. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-25126-9_33
32. Zhou, W., Piramuthu, S.: Consumer preference and service quality management with RFID. Ann. Oper. Res. **216**(1), 35–51 (2014)
33. Zhou, W., Piramuthu, S.: Security/privacy of wearable fitness tracking IoT devices. In: 2014 9th Iberian Conference on Information Systems and Technologies (CISTI), pp. 1–5. IEEE, June 2014
34. Zhou, W., Piramuthu, S.: Information relevance model of customized privacy for IoT. J. Bus. Ethics **131**(1), 19–30 (2015)
35. Zhou, W., Piramuthu, S.: IoT and supply chain traceability. In: Doss, R., Piramuthu, S., Zhou, W. (eds.) FNSS 2015. CCIS, vol. 523, pp. 156–165. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-19210-9_11
36. Zhou, W., Piramuthu, S.: Effects of ticket-switching on inventory management: actual vs. information system-based data. Decis. Support Syst. **77**, 31–40 (2015)
37. Zhou, W., Piramuthu, S.: Effect of ticket-switching on inventory and shelf-space allocation. Decis. Support Syst. **69**, 31–39 (2015)
38. Zhou, W., Piramuthu, S., Chu, F., Chu, C.: RFID-enabled flexible warehousing. Decis. Support Syst. **98**, 99–112 (2017)
39. Zhou, W., Piramuthu, S.: IoT security perspective of a flexible healthcare supply chain. Inf. Technol. Manag., 1–13 (2017)
40. Zhou, W., Piramuthu, S.: Identification shrinkage in inventory management: an RFID-based solution. Ann. Oper. Res. **258**(2), 285–300 (2017)

Thalos: Secure File Storage in Untrusted Clouds

Luca Maria Castiglione and Simon Pietro Romano(✉)

DIETI - Dipartimento di Ingegneria Elettrica e delle Tecnologie dell'Informazione,
Università degli Studi di Napoli "Federico II", Naples, Italy
luc.castiglione@studenti.unina.it, spromano@unina.it

Abstract. In this paper we present Thalos, an architecture for the secure storage of files in the presence of untrusted third parties. Namely, Thalos has been conceived at the outset as a system for protecting both the confidentiality and the privacy of users who rely on an untrusted remote server for storing their files. The system ha been designed as a browser-enabled client-server application and its implementation has been conducted by leveraging the Model-View-Controller pattern. The paper discusses the rationale behind our work, as well as briefly presents the design and implementation phases by focusing on the main use cases that Thalos is capable to support.

1 Introduction

Nowadays, there is a growing interest towards the possibility of remotely storing our files while making them readily available across multiple devices. People do not normally manage their own storage servers; thus, they need to rely on third-party, cloud-based storage services like Google Drive or Dropbox, which have rapidly gained momentum in the technology market. We might ask ourselves how much secure are these kinds of services [1] and what would happen to our files if someone seized storage servers or hacked into them. More in general, we might wonder whether to trust those companies at all.

This paper presents Thalos as a solution to the above-mentioned issues.

Thalos is an extremely robust storage service that is made secure by design. The chosen cryptographic algorithms and the way they are applied offer to the final users the opportunity to securely store their files remotely, while denying any attempt to access them without the proper authorization. Thalos design, indeed, makes it impossible for anyone who has physical or virtual access to the servers to decrypt files without the right key. It also prevents any possibility of establishing an exact match between one specific file and its owner. Thalos relies on local elaborations to perform encryption: everything outside the owner's computer is hard encrypted with asymmetric algorithms (AES 4096 bit key), according to OpenPGP standards. Due to the most known critical issues that belong to read and write operations, in fact, cryptography is executed locally on the machine of the user who owns the original contents. About that, in no way does Thalos memorize keys or pass-phrases in browser cookies or anywhere else.

R. Doss et al. (Eds.): FNSS 2018, CCIS 878, pp. 178–192, 2018.
https://doi.org/10.1007/978-3-319-94421-0_14

Thalos will be provided as a service that can be easily used, in theory, by any device connected to the Internet. Prospective users can easily register an account by using their email address and choosing a username and a password; sessions keep track of the users across the application. Once a user is registered, a first key pair can be generated. The following keys are created: (i) **Master Key:** the private key of the cryptographic key pair. It belongs to the user that can unlock it through a pass-phrase chosen during the creation process; (ii) **Public key:** this is the public key of the pair and is stored on a remote database. It will also be used for secure file sharing in future improvements.

Once a key pair is generated it is possible to add a basket to one's own basket list. Baskets are virtual file containers (they can be thought of as very simple virtual file systems). Each basket is described by a basket description file which basically stores information about contained files, including name, type, size and a pointer to the encrypted file on disk (attribute id) as it can be seen in Fig. 1.

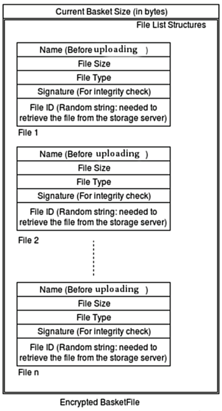

Current Basket Size (in bytes)
File List Structures
Name (Before uploading)
File Size
File Type
Signature (For integrity check)
File ID (Random string: needed to retrieve the file from the storage server)
File 1
Name (Before uploading)
File Size
File Type
Signature (For integrity check)
File ID (Random string: needed to retrieve the file from the storage server)
File 2
Name (Before uploading)
File Size
File Type
Signature (For integrity check)
File ID (Random string: needed to retrieve the file from the storage server)
File n
Encrypted BasketFile

Fig. 1. Thalos basket description

Together with the basket, two new keys are generated: (i) *Basket Private Key*: used to decode the basket description and each file which belongs to the basket itself; (ii) *Basket Public Key*: used to encode the basket description and each file which belongs to the basket itself.

Basket description files are stored remotely and are encrypted with the basket private key. Furthermore, a base file is associated with each user. It is remotely stored and is encrypted with the Master Key of the user to whom it belongs. A basefile contains the basket private keys of the baskets owned by the user it is associated with.

2 Using Thalos

In the following we will briefly illustrate how our usual friends Alice and Bob can securely store their files using Thalos. Figure 2 illustrates Alice's example usage path with Thalos. The involved sequence of steps is reported below:

1. Alice retrieves from remote her base file which is encrypted with her Master Key Pair;
2. Alice decrypts locally in her laptop the base file and gets the keys needed to unlock her own baskets;
3. Alice retrieves from remote the encrypted description of the "UAV" basket;
4. Alice decrypts the description and gets the entire file list which includes pointers to the actual files on disk;
5. Now Alice can securely download any files she wants to access.

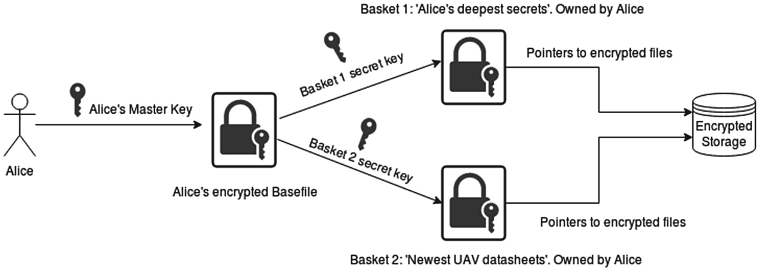

Basket 1: 'Alice's deepest secrets'. Owned by Alice
Basket 1 secret key
Pointers to encrypted files
Alice's Master Key
Alice
Alice's encrypted Basefile
Basket 2 secret key
Encrypted Storage
Pointers to encrypted files
Basket 2: 'Newest UAV datasheets'. Owned by Alice

Fig. 2. Alice's Thalos use path

In much the same way, Fig. 3 sketches Bob's interactions with Thalos:

1. Bob retrieves from remote his base file which is encrypted with his Master Key Pair;
2. Bob decrypts locally in his laptop the base file and gets the keys needed to unlock his own baskets;
3. Bob retrieves from remote the encrypted description of the basket containing his deepest secrets;
4. Bob decrypts the description locally in his laptop and gets the entire file list which includes pointers to the actual files on disk;
5. Now Bob can securely add a brand new secret to his list, staying assured that no one will ever steal it from Thalos.

Alice's and Bob's files are stored (encrypted) on the same hard drive along with other users files. It is impossible to find a reverse path which leads from a file to its owner.

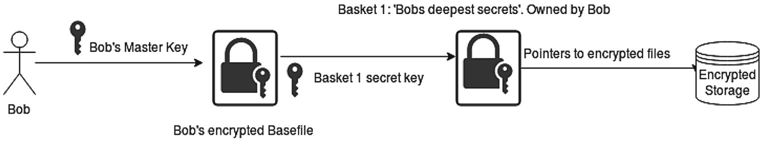


Fig. 3. Bob's Thalos use path

2.1 Protecting Master Keys in Thalos

The *Master Key* appears to be both the bottleneck and the weakness of our storage system. Indeed, the key is strictly related to the device used to read and write remote content and moving it among different laptops or smartphones might constitute a security issue. In addition, in case of a device being lost or stolen, the whole remote content would be exposed to unauthorized users. This problem has been analyzed and a mitigation has been found in what we have called the *Multiple Key Management System*. In the newest version of Thalos, indeed, a user is allowed to generate and manage more than one Master Key (MK). Once a MK has been compromised, the victim can simply disable the access rights deriving from the impaired credentials. The Multiple Key Management System has been designed by taking advantage of the hierarchical structure of Thalos. At the end of a key addition process, on the remote system many basefiles will exist, one for each key actively owned by the end user. To avoid malicious exploitations of this feature, the creation of an additional key is carried out via a feedback from an already existing key, following these steps:

1. Standard user authentication (login and password) from the ‘new device’. Up to this moment the ‘new device’ will be logged-in and it will not be able to decrypt user's content, yet.
2. Key Pair Generation within the context of the ‘new device’. New private Master Key will be stored locally while the public key will be sent to the Thalos Server along with a new key association request. The server forwards the request through a push notification towards the set of already associated devices.
3. The user will accept the request from an already associated device. Using the old (locally stored) private key, the basefile is first decrypted and then re-encrypted with the new public key.
4. From now on, the user can access his files from multiple devices.

The process introduces a minimal redundancy in the system since encryption of a single file is carried out using the keys of the baskets. On the other hand, key compromising is a well-known issue of asymmetric encryption and the Multiple Key Management System is able to completely protect the end user provided that the deletion of an impaired key is carried out as soon as possible.

3 From Theory to Practice: Thalos Software Description

In practice, Thalos shows up as a Web Application that can be reached through any modern Internet browser (it has been successfully tested on Firefox and Google Chrome) and allows users to create an account, generate a master key pair and, eventually, securely manage their files.

More in details Thalos has been developed following a Client-Server pattern where the client role is played by a Web Browser.

3.1 Server Side Architecture

Since the project runs on NodeJS [3], server routines are programmed in server-side Javascript. Furthermore, the server has been designed following a Model View Controller paradigm as showed by the architectural view in Fig. 4. The application uses PUG [4] as view engine to dynamically render HTML pages and SEQUELIZE [6] as ORM (Object to Relational Mapping) tool to dynamically map views inside a relational database and manage migrations.

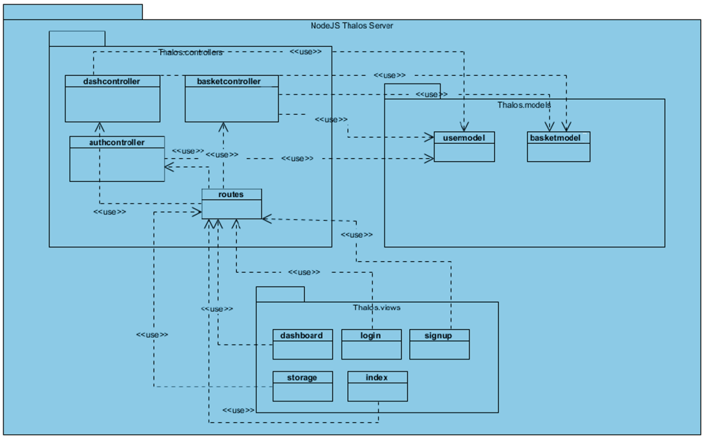
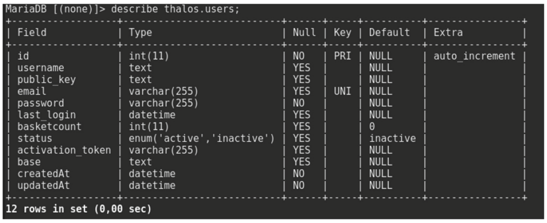


Fig. 4. Thalos architecture

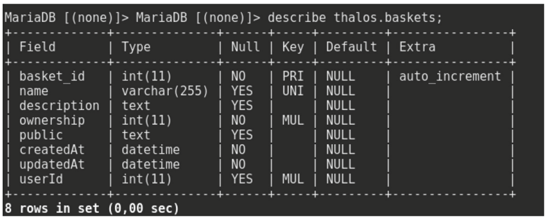
Models Description. Two models are used in this application, namely ‘user’ and ‘basket’. As it can be easily guessed by their name, the former is needed to manage users and the latter to manage baskets. Tables content is described in Figs. 5 and 6, respectively. Particular fields are:

- `users.public_key`: stores the user public key (from master key pair);
- `users.base`: stores the encrypted base file associated with the user;
- `baskets.description`: stores the encrypted basket description.



```
MariaDB [(none)]> describe thalos.users;
+------------------+---------------------------+------+-----+----------+----------------+
| Field            | Type                      | Null | Key | Default  | Extra          |
+------------------+---------------------------+------+-----+----------+----------------+
| id               | int(11)                   | NO   | PRI | NULL     | auto_increment |
| username         | text                      | YES  |     | NULL     |                |
| public_key       | text                      | YES  |     | NULL     |                |
| email            | varchar(255)              | YES  | UNI | NULL     |                |
| password         | varchar(255)              | NO   |     | NULL     |                |
| last_login       | datetime                  | YES  |     | NULL     |                |
| basketcount      | int(11)                   | YES  |     | 0        |                |
| status           | enum('active','inactive') | YES  |     | inactive |                |
| activation_token | varchar(255)              | YES  |     | NULL     |                |
| base             | text                      | YES  |     | NULL     |                |
| createdAt        | datetime                  | NO   |     | NULL     |                |
| updatedAt        | datetime                  | NO   |     | NULL     |                |
+------------------+---------------------------+------+-----+----------+----------------+
12 rows in set (0,00 sec)
```

Fig. 5. Users table



```
MariaDB [(none)]> MariaDB [(none)]> describe thalos.baskets;
+-------------+--------------+------+-----+---------+----------------+
| Field       | Type         | Null | Key | Default | Extra          |
+-------------+--------------+------+-----+---------+----------------+
| basket_id   | int(11)      | NO   | PRI | NULL    | auto_increment |
| name        | varchar(255) | YES  | UNI | NULL    |                |
| description | text         | YES  |     | NULL    |                |
| ownership   | int(11)      | NO   | MUL | NULL    |                |
| public      | text         | YES  |     | NULL    |                |
| createdAt   | datetime     | NO   |     | NULL    |                |
| updatedAt   | datetime     | NO   |     | NULL    |                |
| userId      | int(11)      | YES  | MUL | NULL    |                |
+-------------+--------------+------+-----+---------+----------------+
8 rows in set (0,00 sec)
```

Fig. 6. Baskets table

Views Description. Views are written according to PUG syntax [4]. PUG engine dynamically renders HTML pages. Data coming from controllers are sent to the view as messages through flash [7].

Controller Description. The express [5] framework has been used with NodeJS in order to manage HTTP requests. Express manages incoming connections through the use of routes: when an HTTP request is incoming, it calls the associated callback, if it exists. The following controllers have been defined for Thalos:

- **passportController:** Defines strategies for user login and registration.
- **dashController:** Manages operations on user dashboards. It allows users to upload and download keys and base files, as well as to create baskets. The following interfaces are exposed:
  - `addBasket:` responds to POST requests. Retrieves user data from the current session and updates the basket tables with the POST parameters received along with the request. Returns a JSON object containing the result.
  - `addPublicKey:` responds to POST requests. Retrieves user data from the current session and updates the users table with the new public key and the new base file, both received from POST parameters. Returns a JSON object containing the result.
  - `getBasefile:` responds to POST requests. Retrieves user data from the current session. Returns a JSON object containing the result.

- **basketController:** Manages operations on baskets, like download/upload of a description, download/upload of a file. This controller exposes the following interfaces:
  - `getFile:` responds to POST requests. Returns file selected by file id. Result is returned as a JSON object.
  - `updateBasket:` retrieves user data from the current session and updates the basket tables with POST parameters received along with the request. Returns a JSON object containing the result.
  - `deleteBasket:` responds to POST requests. Deletes a basket. Returns a JSON object containing the result.
  - `getBasket:` responds to POST requests. Retrieves user data from the current session. Returns a JSON object containing the result.
- **authController:** Manages users' authorizations and exposes the following interfaces:
  - `login:` responds to GET requests and commands the PUG engine to show the login page.
  - `signup:` responds to GET requests and commands the PUG engine to show the signup page.
  - `validateUser:` changes user status from 'inactive' to 'active'; this allows the user to login. It's a kind of 'antispam' filter.

Controllers do not implement or call any kind of encryption algorithm since the data they work with are already encoded.

3.2 Client Side Architecture

In order to execute all encryption operations locally to the user machine, particularly in the user browser, the client side part of the project has been written entirely in Javascript. About this, the client side routines require the OpenPGP.js library [2] to perform their duty. The code is divided in three main categories, according to the functions that are carried out.

- **Operations on dashboard:**
  - `genkey:` given a pass-phrase generates a keypair. The public key is sent to the server through AJAX as user public key. The private one is the user Master Key. A downloadable file is generated on the fly and a link is displayed.
  - `addBasket:` given the Master Key, the Master Key pass-phrase and a basket pass-phrase, it generates a new basket for the user who requested it. Eventually the function updates the base file and sends it along with the new basket data to the server through AJAX.
- **Operations on baskets:**
  - `bloadlist:` given a user, it sends an XMLHttpRequest to the server asking for the base file. Eventually, it decrypts the base file and displays the user basket list.

  - `openbasket`: given a user and a basket name, this function sends an XMLHttpRequest to the server asking for the basket description file. Once got it, it decrypts it and displays it to the user.
- **Operations on files:**
  - `Upload`: this function assumes that a file (to upload) and a basket have both been selected. It locally loads the file from an HTML form, saves its related information into a JSON object, encrypts the file, pushes the new JSON into the basket description array and encrypts the description as well. The file and the updated description are sent to the server through the remote interface UpdateBasket.
  - `Download`: this function assumes that a file (to download) and a basket have both been selected. It retrieves file information from the basket description and then requests the selected file to the server through the file id. Once a response from the server has arrived, the client decrypts it and generates a downloadable file on the fly.
  - `Delete`: this function assumes that a file (to delete) and a basket have both been selected. It updates the current basket description by deleting the selected file (identified through the provided file id). The updated basket description is sent to the server along with the query needed to remove the file from the storage server as well.

4 Dynamic Views

Some of the actions described in the previous section are herein reported through sequence diagrams. This section aims to give the reader a clearer vision of the whole project by pointing out how client and server work together.

Figure 7 shows how user registration is based on the validation of an activation code that is generated at subscription time on the server's side.

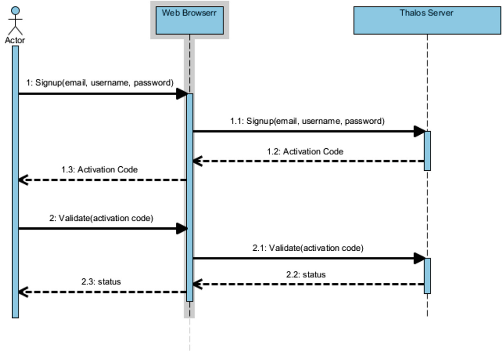

Actor
Web Browserr
Thalos Server
1: Signup(email, username, password)
1.1: Signup(email, username, password)
1.2: Activation Code
1.3: Activation Code
2: Validate(activation code)
2.1: Validate(activation code)
2.2: status
2.3: status

Fig. 7. User registration sequence diagram

Similarly, Fig. 8 illustrates that a request for the creation of a key pair (through a user-provided pass-phrase) is served directly within the browser. Of the pair in question, the private key is provided back to the user, while the public key is delivered to the Thalos server, where it gets stored for all future uses.

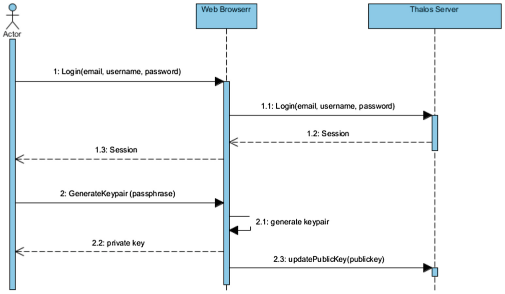

Actor
Web Browserr
Thalos Server
1: Login(email, username, password)
1.1: Login(email, username, password)
1.2: Session
1.3: Session
2: GenerateKeypair(passphrase)
2.1: generate keypair
2.2: private key
2.3: updatePublicKey(publickey)

Fig. 8. Master key pair generation sequence diagram

Figure 9 sketches what happens when creating a Thalos basket. Basically, the original base file is first retrieved from Thalos and locally decrypted with the crypto material provided by the user (pass-phrase and private key). Then, the *addBasket* method is triggered, which translates into the creation of a brand new basket key pair, the update of the downloaded base file and eventually the upload of the updated (and encrypted) base file to the Thalos server, together with the newly generated basket public key.

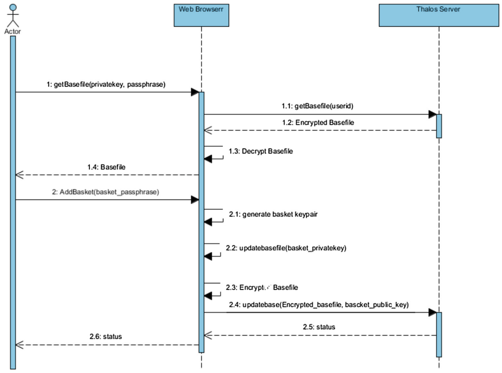

Actor
Web Browserr
Thalos Server
1: getBasefile(privatekey, passphrase)
1.1: getBasefile(userid)
1.2: Encrypted Basefile
1.3: Decrypt Basefile
1.4: Basefile
2: AddBasket(basket_passphrase)
2.1: generate basket keypair
2.2: updatebasefile(basket_privatekey)
2.3: Encrypt Basefile
2.4: updatebase(Encrypted_basefile, bascket_public_key)
2.5: status
2.6: status

Fig. 9. Basket creation sequence diagram.

Figure 10 describes the process of retrieving the list of files contained inside a basket. As already discussed for the previous case, we first download the encrypted base file, which is locally decrypted with user-provided information.

Then, we call the *getBasketList* method on the client-side Thalos JavaScript library. With the basket list readily available, we can eventually call the client-side *openbasket* method, which in turn downloads from the Thalos server the encrypted basket description file. As usual, the encrypted description is locally decrypted and the resulting file list is provided back to the requesting user.

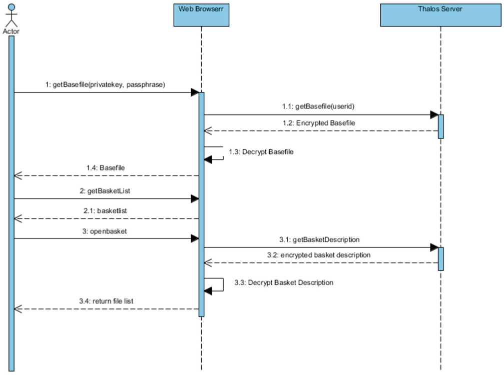


Fig. 10. Basket list retrieval sequence diagram

Figure 11 focuses on file upload. Steps 1 through 3.4 are exactly the same as those described when commenting Fig. 10. Starting from there (i.e., assuming the file list has been made available to the end-user), we can call the client-side *UploadFile* method, which: (i) updates and encrypts the basket description; (ii) encrypts the file to be uploaded; (iii) uploads to the Thalos server both the encrypted file and the encrypted (as well as updated) basket description.

Finally, Fig. 12 focuses on file download. Once again, we start from step 4 in the diagram, which shows how a call to the client-side *GetFile* method gets translated into an analogous *getFile* call to the Thalos server. Such a call allows the browser to download an encrypted copy of the requested file, which is decrypted on-the-fly and provided to the end-user in the clear.

5 The External Perspective

In this last section we try to follow the breadcrumbs left by a user who uses Thalos to securely store his/her files. The goal is to show to the reader how the system works in terms of files and data stored on the server. As it can be seen in Fig. 13, the actor in question first of all creates an account in the webapp and validates it.

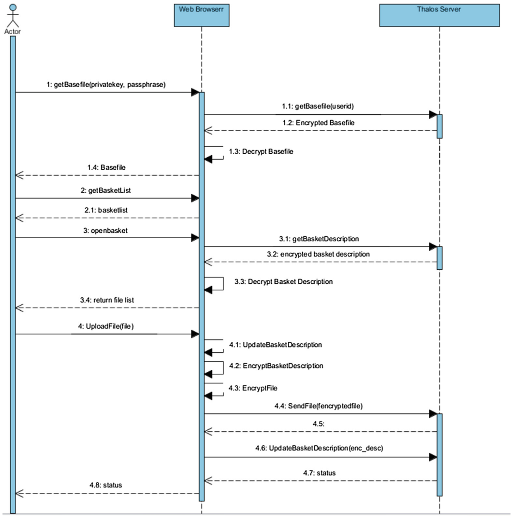

Actor
Web Browserr
Thalos Server
1: getBasefile(privatekey, passphrase)
1.1: getBasefile(userid)
1.2: Encrypted Basefile
1.3: Decrypt Basefile
1.4: Basefile
2: getBasketList
2.1: basketlist
3: openbasket
3.1: getBasketDescription
3.2: encrypted basket description
3.3: Decrypt Basket Description
3.4: return file list
4: UploadFile(file)
4.1: UpdateBasketDescription
4.2: EncryptBasketDescription
4.3: EncryptFile
4.4: SendFile(fencryptedfile)
4.5:
4.6: UpdateBasketDescription(enc_desc)
4.7: status
4.8: status

Fig. 11. File upload sequence diagram

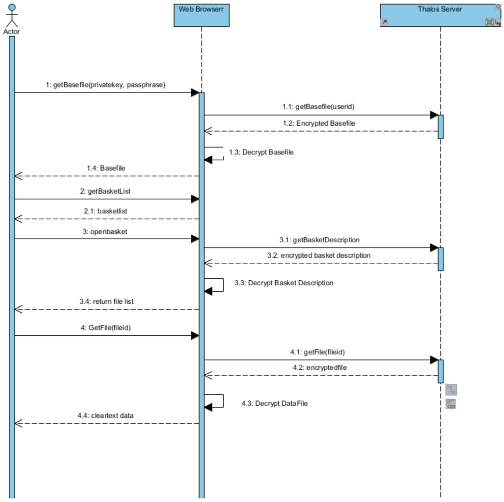
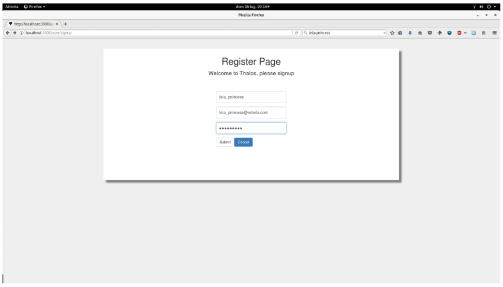


Fig. 12. File download sequence diagram

Once done with the previous step, our special guest signs in through the dashboard and generates his/her master key pair by clicking on the *Generate Master Key* button (Fig. 14).



Register Page

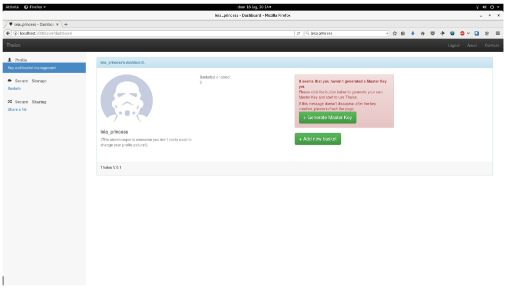
Welcome to Thalos, please signup

leia_princess

leia_princess@rebels.com

••••••••••

Fig. 13. Account registration



leia_princess's dashboard.

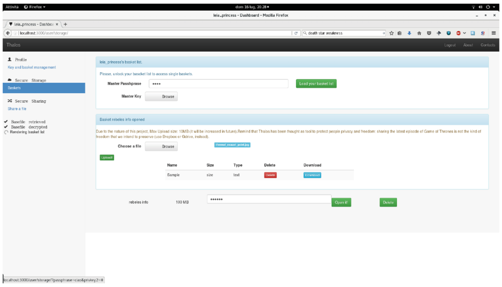
Basket(s) enabled:
0

It seems that you haven't generated a Master Key yet.
Please click the button below to generate your own Master Key and start to use Thalos.
If this message doesn't disappear after the key creation, please refresh the page.

leia_princess
(This stormtrooper is awesome you don't really need to change your profile picture!).

Thalos V 0.1

Fig. 14. Dashboard



✓ Basefile retrieved
✓ Basefile decrypted
○ Rendering basket list

leia_princess's basket list.

Please, unlock your basket list to access single baskets.

Master Passphrase ••••

Master Key

Basket rebeles info opened

Due to the nature of this project, Max Upload size: 10MB (it will be increased in future).Remind that Thalos has been thought as tool to protect people privacy and freedom: sharing the latest episode of Game of Thrones is not the kind of freedom that we intend to preserve (use Dropbox or Gdrive, instead).

Choose a file thermal_exaust_point.jpg

Name	Size	Type	Delete	Download
Sample	size	text	Delete	Download

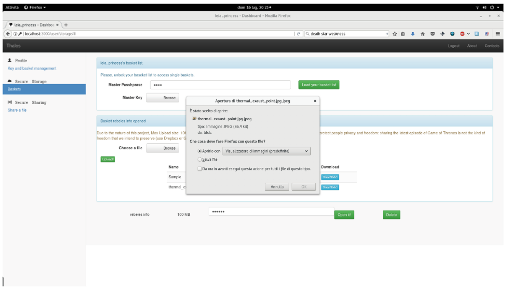
rebeles info 100 MB ••••••

localhost:3000/user/storage/?passphrase=ciao&privkey2=#

Fig. 15. Information uploading

Using the newly created key pair, the user adds a basket to his/her basket list by following the instructions displayed in the web consolle. Eventually, he/she uploads confidential information (Fig. 15) to the remote Thalos server.

From now on, the file is in the secure storage and it is ready to be downloaded by its owner whenever the need arises, as illustrated in Fig. 16.



leia_princess's basket list.

Please, unlock your bascket list to access single baskets.

Master Passhprase ••••

Master Key

Basket rebeles info opened

Due to the nature of this project, Max Upload size: 10M[...] protect people privacy and freedom: sharing the latest episode of Game of Thrones is not the kind of freedom that we intend to preserve (use Dropbox or G[...]

Choose a file

rebeles info 100 MB ••••••

Apertura di thermal_exaust_point.jpg.jpeg

È stato scelto di aprire:

thermal_exaust_point.jpg.jpeg

tipo: Immagine JPEG (36,4 kB)

da: blob:

Che cosa deve fare Firefox con questo file?

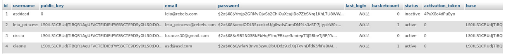
● Aprirlo con Visualizzatore di immagini (predefinita)

○ Salva file

□ Da ora in avanti esegui questa azione per tutti i file di questo tipo.

Fig. 16. Information downloading

Let's now assume that a bad guy has gotten, in some way, access to the database. What can he actually learn about our user? In Fig. 17 a view of the database is shown.

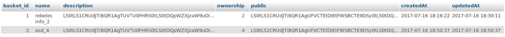


id	username	public_key	email	password	last_login	basketcount	status	activation_token	base
1	asddasd	0	leia@rebels.com	$2a$08$Ymjp2GRMvOjvSb2ChiOuXeajiBo7ZbSNrg1KhL7U8WW...	*NULL*	0	inactive	4PuX8c4dPu8yo	
2	leia_princess	LS0tLS1CRUdJTiBQR1AgUFVCTElDIEtFWSBCTE9DSy0tLS0tDQ...	leia_princess@rebels.com	$2a$08$omdODL5SxcnknU/g6wdsCumDM9Ls3zSTr7JIypIrW0c...	*NULL*	1	active	0	LS0tLS1CRUdJTiBQI
3	ciccio	LS0tLS1CRUdJTiBQR1AgUFVCTElDIEtFWSBCTE9DSy0tLS0tDQ...	lucacas30@gmail.com	$2a$08$c9B5N0SRkEbHqEYm/ERkqe/kmiepT3J5RbeTJ/lP/Yk...	*NULL*	0	active	0	LS0tLS1CRUdJTiBQI
4	ciaone	LS0tLS1CRUdJTiBQR1AgUFVCTElDIEtFWSBCTE9DSy0tLS0tDQ...	asd@asd.com	$2a$08$0/wlaNBnvo3zwuDbUDdz9.clXqTxvnDfd63/hPaj0AI...	*NULL*	1	active	0	LS0tLS1CRUdJTiBQI

Fig. 17. Database view: users

From the attached snapshot, we can derive that the attacker is now sure that *princess Leia* uses Thalos and that she owns one basket named *rebels_info*, as reported in the further snapshot in Fig. 18. Thus, the attacker tries to decrypt the basket in question by reading the *princess* basefile where keys are stored. What he gets is just a meaningless sequence of PGP-encrypted bytes.

The same thing happens if he tries to read any of the basket descriptions. The only information that is 'leaked' is that Leia created one container. Though, nothing is leaked with respect to sensitive data like, e.g., the total number of files that have been uploaded.



basket_id	name	description	ownership	public	createdAt	updatedAt
1	rebeles info_2	LS0tLS1CRUdJTiBQR1AgTUVTU0FHRS0tLS0tDQpWZXJzaW9uOi...	2	LS0tLS1CRUdJTiBQR1AgUFVCTElDIEtFWSBCTE9DSy0tLS0tDQ...	2017-07-16 18:16:22	2017-07-16 18:30:11
2	asd_4	LS0tLS1CRUdJTiBQR1AgTUVTU0FHRS0tLS0tDQpWZXJzaW9uOi...	4	LS0tLS1CRUdJTiBQR1AgUFVCTElDIEtFWSBCTE9DSy0tLS0tDQ...	2017-07-16 18:50:37	2017-07-16 18:50:37

Fig. 18. Database view: baskets

Assume that, at this point, the attacker accesses the hard drive as well as the database. When he tries to list the storage directory, the only thing that he realizes is that a lot of files are saved on disk with a random generated 199 characters length and a name that is encrypted with some key.

Again, the match between each file and its owner is recorded into the basket description that is encrypted with the basket key, which, itself, needs to be decoded with the master key. In conclusion the attacker will never discover sensitive user's information as long as the user in question keeps his/her Master Key in a safe place.

6 Related Work

In this section we provide an overview of common services that focus on *untrusted computing* and its application to the field of secure storage techniques. We will try and highlight their differences with respect to our solution, for better or worse.

MegaNZ is probably the most famous service on the market offering secure storage for free on the Internet with a file level granularity. Mega developers have written from scratch their own implementation of encryption algorithms using Javascript asm. Files encryption works locally to the user machine and their work has been open-sourced[1]. Moreover, the service is offered through a friendly interface and the asymmetric encryption process is completely transparent to the end user. On the other hand, the infrastructure does not provide any form of anonymization. In fact, as written within the privacy policies, information on files such as metadata, ownership and upload date is clearly stored by remote servers[2]. Also, the asymmetric encryption breaks when the needs arises to share a file. In such a case, the key needed for file decryption has to be explicitly provided along with the file. Finally, the entire server side architecture is kept hidden by the company; in this sense, the service cannot be deployed within a private network.

Storj [8] is an interesting service that uses a pure P2P configuration in order to keep user files secret. Every file is split in hundreds of shards that are stored, encrypted, all over the nodes. The service is completely free and the code, written in C, has been open-sourced. Unlike our solution, this service comes with the strengths and weaknesses of peer-to-peer and neither reliability nor availability of files can be ensured under all circumstances.

Clear storage with an encrypted security layer is another approach that can be considered capable to reach a good privacy level in remote storage. It consists in adding a double key encryption layer to a common storage service such as Dropbox, Google Drive and Microsoft One Drive. Many applications have been developed with this purpose but, unlike Thalos, they cannot provide the user with file anonymization. This approach indeed requires a user to be aware of the common privacy issues, as well as of the existence of countermeasures such as advanced encryption.

[1] https://github.com/meganz/webclient.

[2] https://mega.nz/privacy.

7 Conclusions

In this paper we have presented Thalos, an architecture for the storage of content within third-party storage facilities, with both security and privacy guarantees. We have discussed how Thalos has been designed and implemented as a remotely accessible, web-enabled service. We have also briefly compared Thalos functionality with wide-spread cloud-based storage facilities.

Thalos has been already presented to the international security community as an open-source tool for the secure storage of contents in the presence of untrusted third-party storage providers. Namely, the project has been presented at the recent BlackHat Europe 2017 conference that has taken place in London between the 4^{th} ad the 7^{th} of December 2017. BlackHat is a renowned venue for security researchers and practitioners, providing attendees with the very latest advances in research, development, and trends in Information Security. Thalos has been part of the so-called *BlackHat Arsenal*[3], that is a session entirely devoted to the presentation of cutting-edge tools in all fields of security.

Source code, documentation and installation information for Thalos are all publicly available on gitlab at the following address: http://gitlab.comics.unina.it/NS-Projects/Thalos.

Acknowledgments. This work was partially funded by the European Space Agency, within the framework of project SHINE, ESA Contract No.: 4000118273/16/NL/CLP.

References

1. Rong, C., Nguyen, S.T., Jaatun, M.G.: Beyond lightning: a survey on security challenges in cloud computing. Comput. Electr. Eng. **39**(1), 47–54 (2013). ISSN 0045–7906
2. OpenPGPjs.org, OpenPGP.js. https://openpgpjs.org/, https://github.com/openpgpjs/openpgpjs
3. NodeJS Foundation: NodeJS. https://nodejs.org/en/
4. PUGjs. https://pugjs.org
5. Expressjs. https://expressjs.com/
6. Sequelizejs. http://docs.sequelizejs.com/
7. FlashJS. http://flashjs.org/
8. Storj. https://storj.io/

[3] http://www.blackhat.com/eu-17/arsenal/schedule/index.html.

Ubiquitous Individual Information Systems

Claris Chung, Khushbu Tilvawala, Shohil Kishore, Gabrielle Peko,
Asfahaan Mirza, and David Sundaram(✉)

Department of Information Systems and Operations Management,
University of Auckland, Auckland, New Zealand
{claris.chung, k.tilvawala, s.kishore, g.peko,
a.mirza, d.sundaram}@auckland.ac.nz

1 Introduction

There is more computing power in your smart phone now than all the computers used by NASA in 1969 to place man on the moon. People spend more time on mobile apps than on their desktops. These mobile apps are proliferating and weaving themselves into every facet of our life (physical, emotional, financial, relational, and spiritual). After the first 500 apps in the Apple App Store when it made its debut in July 2008 [1], mobile application development has been dramatically growing and millions of apps for personal use have been created and deployed. The growth of Ubiquitous Individual Information Systems [UIIS] potentially poses many questions, problems, issues, and requirements for the design of UIIS.

However, the focus of seminal IS research has predominantly been on the design of traditional organizational information systems to support professional/supporting organizational processes in the context of the office [2]. IS research has largely ignored the design of ubiquitous information systems for personal and/or professional purposes in the context of the home, office, and/or other contexts. Alter [3] reviewed more than 20 different definitions of information systems from published papers. While most include references to computers or technology, and most also refer to organizations in some way, individually owned IS was excluded of these conceptualizations.

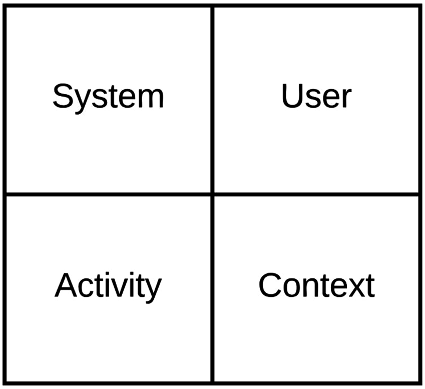
In IS research, often newly emerged system concepts and phenomenon lead many large, pragmatic investigations. Baskerville [4] has spotted this research gap and elaborated the individual information systems as a research arena. He suggested that an individual information system is "an activity system in which individual persons, according to idiosyncratic needs and preferences, perform processes and activities using information, technology, and other resources to produce informational products and/or services for use by themselves or others". However, no seminal study was followed nor developed by any IS researchers since his elaboration.

One of the challenges in designing and developing a ubiquitous individual information system is a tension between individuals' distinctive needs and system components availability. Each individual has unique situations and roles within different life dimensions even in a single day. In addition, life dimensions are highly interrelated therefore the boundary between each life dimension is generally interdependent and inseparable. On the other hand, common features and functionalities need to be

R. Doss et al. (Eds.): FNSS 2018, CCIS 878, pp. 193–196, 2018.
https://doi.org/10.1007/978-3-319-94421-0_15

identified for developing the systems. Resting atop this tension, a design process is premised to be experiential in UIIS design [4].

Another approach of designing Ubiquitous Individual Information Systems is considering design dimensions of the systems. The major design dimensions are namely the system, the activity, the user, and the context that the system is designed to support (Fig. 1). Focusing on one or more dimension(s), many design research questions for UIIS can be raised.



System	User
Activity	Context

Fig. 1. Experiential design of ubiquitous individual information systems

2 Research Questions

A candidate list of questions that can be explored in this research are:

1. **What are UIIS?**
   a. **What are the key reference disciplines, concepts, and processes that could be used to underpin UIIS?**

The key reference disciplines would definitely include Computer Science [CS] and Information Systems [IS] but in addition could include domain relevant disciplines. For instance, it could include medicine if we were designing health apps or it could include finance if we were designing financial apps.

Concepts that could underpin UIIS would range from core IS and CS concepts to do with aesthetics, design, human behavior, psychology, visualization, User Interface [UI], User Experience [UX], and the domain(s) under consideration.

Obviously, design, implementation, and usage processes would form a key component of the processes that would support UIIS. In addition, domain specific processes as well as configuration and integration of apps processes would be significant.

   b. **What are key models and frameworks that could help us understand, structure, and communicate UIIS?**

While many of the IS and CS models and frameworks would inform us to some extent regarding the UIIS we believe that there is a need to propose new models and frameworks that reflect the particular aspects, issues, features, and challenges of the nexus between ubiquitous, individual, integrated, information systems.

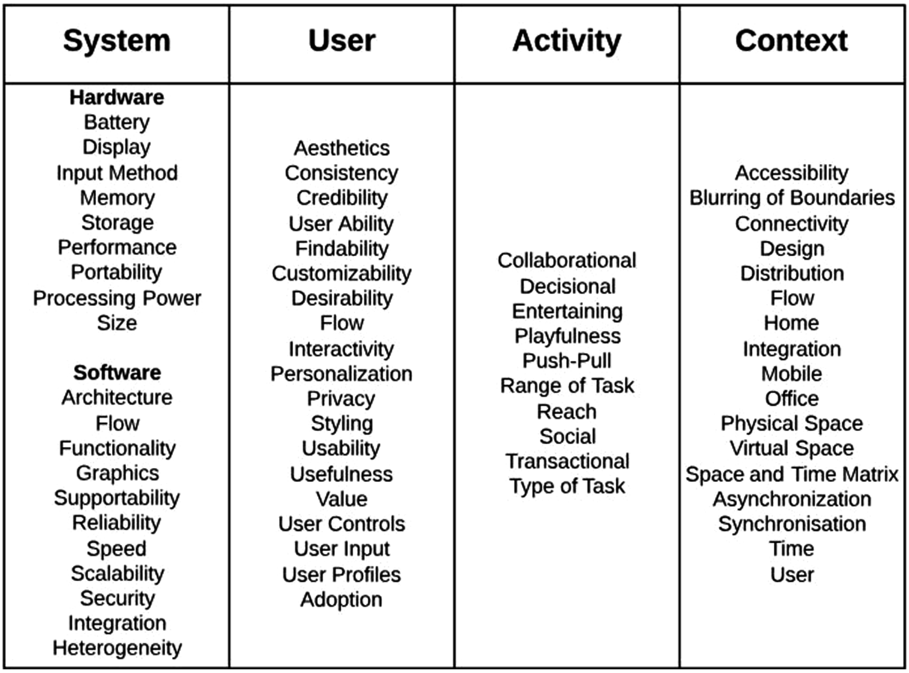
2. **How do we design UIIS?**
   a. **How do we enable these UIIS to integrate and collaborate with each other in a holistic manner?**

The integration and active collaboration of diverse applications together as well as with cloud services and other associated databases and servers can be technically challenging when there are few standards around.

   b. **How do we increase the density of the intelligence provided by these UIIS from data to information to knowledge to wisdom?**

The processes involved in extracting data from various devices, sensors, and systems and integrating them and transforming them into highly dense information that could be used by individual decision makers can be numerous and complex. The processes encompass the transactional to the decisional.

3. **How do we implement UIIS?**
   a. **How do we implement consumption as well as creation-oriented applications for UIIS?**
   b. **How do we implement personal and professional applications that allow seamless interweaving between different contexts (home, office, and other spaces) in a secure and well governed manner?**



System	User	Activity	Context
Hardware Battery Display Input Method Memory Storage Performance Portability Processing Power Size **Software** Architecture Flow Functionality Graphics Supportability Reliability Speed Scalability Security Integration Heterogeneity	Aesthetics Consistency Credibility User Ability Findability Customizability Desirability Flow Interactivity Personalization Privacy Styling Usability Usefulness Value User Controls User Input User Profiles Adoption	Collaborational Decisional Entertaining Playfulness Push-Pull Range of Task Reach Social Transactional Type of Task	Accessibility Blurring of Boundaries Connectivity Design Distribution Flow Home Integration Mobile Office Physical Space Virtual Space Space and Time Matrix Asynchronization Synchronisation Time User

Fig. 2. Dimensions of ubiquitous individual information systems design

c. **How do we support the customization and configuration of such complex and unique UIIS?**
d. **How do we create a portfolio of UIISs that are integrated towards serving a particular purpose?**
e. **How do we implement UIIS that span work, home, societal, and cultural boundaries?**
f. **How do we implement UIIS.**

We explore these questions in light of four major dimensions (system, user, activity and context) and their sub-dimensions (Fig. 2). These dimensions represent an applied version of the MIT90s framework, where system represents technological aspects, users represent individuals and roles, activities represent process, and context represents the external environment.

3 Research Approach

The aim of this research is to define, design and implement UIIS. The word “Design” means “to create, fashion, execute, or construct according to plan” [5]. Therefore, it is best to discover through design [4] and adapt a multi-methodological approach to conduct this design science research. For this study, Nunamaker’s et al. [6] multi-methodological approach for information systems research will be adapted to propose and develop various artifacts. Moreover, the criteria for the design science artifacts proposed by both Nunamaker et al. [6] and Hevner et al. [7] will be followed throughout the study. The adapted multi-methodological approach is a practical way of designing and implementing a system. It consists of five research strategies/phases - observation, theory building, systems development, evaluation, and generalization. The phases are all mutually connected to support creation and validation of a system with multiple iterations.

References

1. Strain, M.: 1983 to today: a history of mobile apps (2015). http://www.theguardian.com/media-network/2015/feb/13/history-mobile-apps-future-interactive-timeline
2. Crowston, K., Fitzgerald, B., Gloor, P., Schultze, U., Yoo, Y.: Shifting boundaries: how should IS researchers study non-organizational uses of ICT? In: Proceedings of the 2010 International Conference on Information Systems - ICIS 2010, pp. 1–5 (2010)
3. Alter, S.: Defining information systems as work systems: implications for the IS field. Eur. J. Inf. Syst. **17**, 448–469 (2008). https://doi.org/10.1057/ejis.2008.37
4. Baskerville, R.: Individual information systems as a research arena. Eur. J. Inf. Syst. **20**, 251–254 (2011). https://doi.org/10.1057/ejis.2011.8
5. Merriam-Webster: Design. https://www.merriam-webster.com/dictionary/design
6. Nunamaker, J., Chen, M., Purdin, T.: Systems development in information systems research. J. Manag. Inf. Syst. **7**, 89–106 (1991). https://doi.org/10.1109/ISIE.1992.279627
7. Hevner, A.R., March, S.T., Park, J., Ram, S.: Design science in information systems research. MIS Q. **28**, 75–105 (2004). https://doi.org/10.2307/25148625

Security and Privacy Protection for eHealth Data

Sharmin Jahan[1], Mozammel Chowdhury[2](✉), Rafiqul Islam[2], and Junbin Gao[3]

[1] Department of Biochemistry and Molecular Biology, Jahangirnagar University, Dhaka, Bangladesh
sharmin.biochemist@yahoo.com

[2] School of Computing and Mathematics, Charles Sturt University, Bathurst, NSW 2795, Australia
{mochowdhury,mislam}@csu.edu.au

[3] Discipline of Business Analytics, The University of Sydney Business School, The University of Sydney, Sydney, NSW 2006, Australia
junbin.gao@sydney.edu.au

Abstract. Recently, both security and privacy are the growing concerns in eHealth platforms that deal with sensitive clinical data stored in electronic health records (EHR). Breaches or damage of sensitive data of an individual's health record can be occurred due to attacks by hackers or malicious insiders. Therefore, it is very crucial to enforce privacy and security of clinical data in eHealth applications by technological means. Understanding and finding the issues related to the security and privacy of eHealth systems are important in designing and developing an effective eHealth system. In this paper, we therefore aim to investigate and analyze the recent security issues in eHealth applications and explore their solutions to preserve privacy and security of sensitive health data.

Keywords: Security · Privacy · Clinical data · eHealth · EHR

1 Introduction

eHealth or electronic health is the delivery of health services and resources by electronic means through extensive information sharing and collaboration [1]. The European Commission (EC) has defined eHealth as the use of modern information and communication technology (ICT) to improve the access, efficiency, effectiveness, and quality of clinical and business processes utilized by healthcare organizations, practitioners, patients, and consumers in an effort to improve the healthcare status [2].

Over the recent years, many eHealth applications have been proposed worldwide. Due to the sensitive nature of medical information and healthcare records, issues of integrity, security, privacy, and confidentiality are very significant. Hence, privacy and security must be effectively addressed in developing eHealth schemes to protect patient's health data. There are several laws around the world designed to protect the electronic health data that the healthcare institutions maintain for their patients [3–5]. Furthermore, many sophisticated security mechanisms, such as access control

R. Doss et al. (Eds.): FNSS 2018, CCIS 878, pp. 197–205, 2018.
https://doi.org/10.1007/978-3-319-94421-0_16

mechanisms [32], encryption techniques [21] and auditing tools [33, 34] are developed for secure eHealth systems.

Owing to several key challenges, the widespread practice of eHealth systems is still at premature stage. Hence, it is significant to understand how far eHealth data are protected and what factors can lead to enhance a successful eHealth system. We therefore investigate the issues, drivers, and initiatives for security, compliance, and interoperability in healthcare activities carried out by means of information and communication technologies. The possible solutions against these challenges are also evaluated that will help professional to develop secure and effective eHealth systems.

2 Methodology

This paper employs a systematic review to find relevant sources and identify the issues and challenges in eHealth implementation. A systematic review is a research technique that attempts to collect all empirical evidence regarding a research question, to assess it critically and to obtain conclusions to support the development of guidelines to solve the problem. This systematic review has followed the quality reporting guidelines set by the Preferred Reporting Items for Systematic reviews and Meta-Analysis (PRISMA) [6].

The selection of sources is based on the security and issues of eHealth. We include only the articles written in English language, as it is the international language widely employed in research studies. The selection process is performed by searching the articles with a search engine using the strings such as: ‘eHealth’, ‘Electronic Health Record’, ‘Electronic Medical Record’, AND (‘Privacy’ OR ‘Security’). The searching process is applied to MEDLINE, ACM Digital Library, Wiley InterScience, IEEE Digital Library, ScienceDirect, Scopus, MetaPress, and ERIC database (Fig. 1). We also scanned the reference lists included in articles to ensure that this review would be more comprehensive. All articles from 2002 till 2017 are considered for searching in different databases. We find that many appeared articles from the search are duplicated. Therefore, Endnote software helped to avoid downloading duplicate articles. To identify more relevant articles, we have considered the abstracts as well.

3 Findings

This section demonstrates the major findings of the literature study. We have identified a number of recent issues regarding privacy and security in eHealth applications and their solutions for effective implementation.

3.1 Issues in Accessing Electronic Health Record (EHR)

The primary concern regarding the privacy and security of eHealth system is the ‘data privacy’ which refers to the ability of an individual to control over their clinical data and disclosure of personal information. The Electronic Health Record (EHR) of an eHealth system contains huge amount of clinical data that are electronically recorded and available. These clinical data consist of medical and scientific documents and of

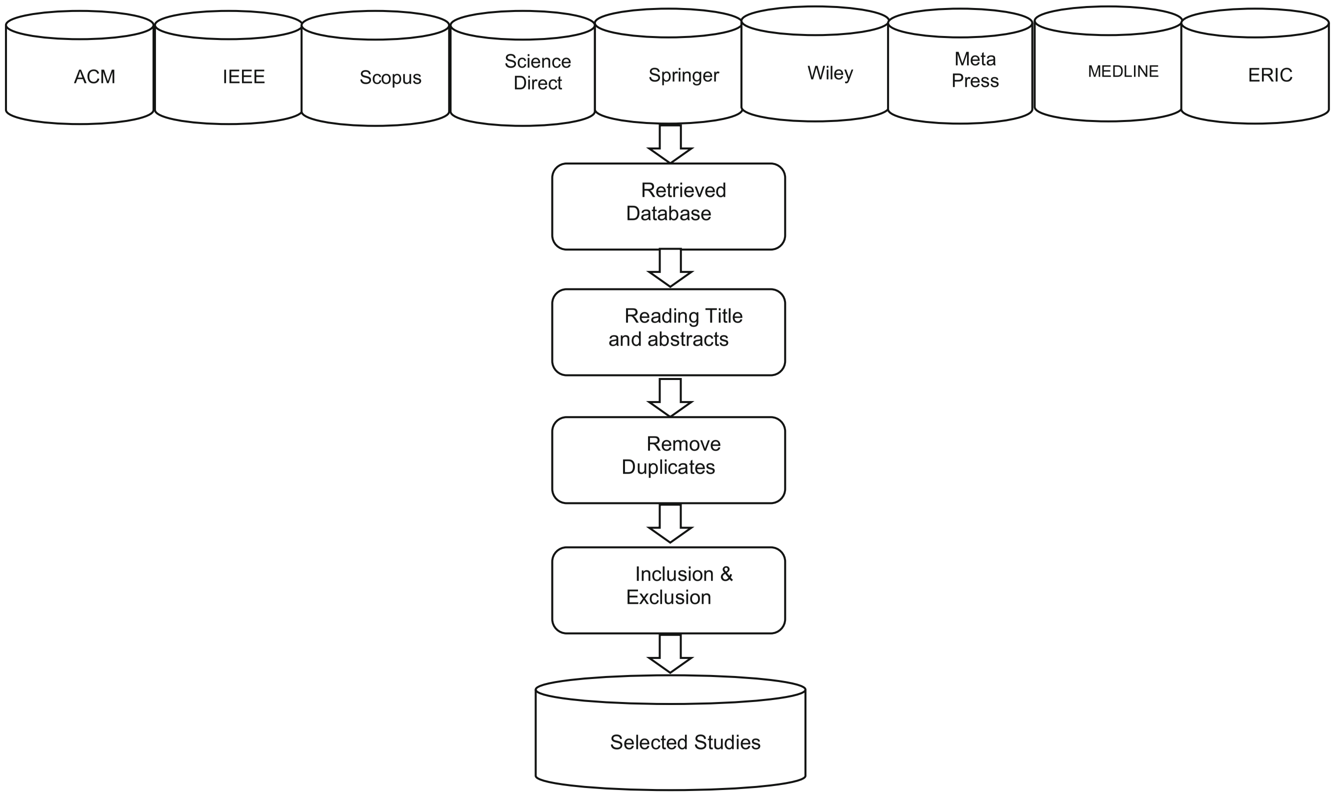


Fig. 1. Selection process of study sources.

patient health records. An electronic health record is a digital version of a patient's paper chart. EHR contains a patient's medical history, diagnoses, medications, treatment plans, immunization dates, allergies, radiology images, and laboratory and test results [7].

Since medical information is usually associated with individuals, privacy must be ensured when data is made available for secondary use. Nowadays, EHRs are collected and maintained by public and private institutions that made them available for health practitioners and researchers. Those institutions should guarantee that health information associated with the patients are only made public with their authorization [8]. Moreover, the US Health Insurance Portability and Accountability Act (HIPAA) privacy rule permits publishing personal health information for public-health purposes without patient consent, if individual's privacy is sufficiently guaranteed [4].

The EHR is the fundamental tool which every doctor needs to access for providing the best care to the patients. Personal records of hospitals that contain information such as, date and numbers of birth, immunization and death, for instance are important for state control agencies. Healthcare suffers when patient's data are not at the right place or are lost or damaged. It must be assured that everyone, who has the right, can access the record, in an efficient manner, where no waiting time or a deadlock situation can occur.

A patient-centric eHealth system must provide the patients with control over the utilization and dissemination of their own private information [9]. Unfortunately, traditional security mechanisms are insufficient to meet the requirements of patient-centric eHealth services in the open cloud environment. A study [10] has identified a set of security requirements for eHealth services hosted by a Cloud computing environment, including authentication, authorization, ownership of information, and integrity, confidentiality and availability of data. A model is proposed in that study to address the security and privacy issues relating to access to and management of Electronic Health Records (EHRs).

More use of EHRs can also cause less security to the eHealth system. For example, if a patient allows his or her spouse or friend to access his/her personal records, risk of data leakage can be raised if the user is not expert or honest. In addition, the incompatibility between different systems and database remain a threat to integrate records. Although this will diminish with the adoption of consistent technology and data standard. This requires more effort to ensure the effective interactivity between the patients and the health provider record.

To mitigate privacy attacks, various privacy preserving approaches such as profile matching [11], pseudonyms [12], and attribute-based secure mechanisms [13–15] have been proposed recently. Based on the bilinear pairings, Lu et al. [11] proposed a secure handshake scheme that enables patients with similar symptoms to share their personal health information within a mobile healthcare social network. With this approach, a pseudo-ID is generated for each patient along with a private key corresponding to his/her symptom. If two patients meet and know they have the same symptom, they can use their own private keys for the specific symptom to achieve mutual authentication. However, if the symptoms of both patients are different, the proposed scheme does not disclose each other's symptom information. This approach allows patients to reap the benefits of a mobile healthcare social network while collaborating on their personal health information.

3.2 Attacks on the Host Environment

Attack on host environment can be established by three ways: hardware concession, software concession, and user concession. Faulty hardware could be a serious concern for data integrity. Software concession could arise from software updates. Attacks by malicious software can be established when patients install applications on their smart phones to share clinical information with health practitioners. In user concession, the assailant could gain unauthorized access to a patient network resources and devices masquerading as the patient [8]. When there is a threat from the host network, integrity of eHealth data cannot be guaranteed. According to [16], hardware and software infrastructures needs to be encrypted and dully tested and certified before installations. For an update to be performed, a secured distribution platform is required to guild against malicious and unauthorized updates.

3.3 Internet Security Issues

The internet has become the most popular source of information that connects individuals with health experts and supports. A lot of services could be accessed via the internet which is a medium without borders. Many websites are not trusted and thus when users use these mediums may face system vulnerabilities. Most patient's data are transmitted via the internet. Doctors and health professionals help to feed the system with all necessary information to be accessed online. Hence, transmissions of health data over the internet are vulnerable to several threats.

The essence of eHealth system is to make health care services readily available to people as well as provide safety means to exchange and share medical information, improve the quality of service offered to patients, secure message transmission, processing and storage of patient's data with regards to record keeping, outcome of expectation, confidentiality and billing [17, 18]. Despite consumer's satisfaction, incorrect information could be life threatening and so also is wrong prescription of drugs and health maintenance while maintaining the security, integrity, and confidentiality of patient data. However, most times, patient's data is faced with variety of attacks ranging from unauthorized access to theft and alterations of patient records. To control this threat, Fan et al. propose the Data Capture and Auto Identification Reference (DACAR). This system aims to solve the problem by providing a cloud based secure eHealth that the core component is Single Point of Contact (SPOC). This addresses the most fundamental security requirements such as secure data transmission, authentication, authorization and persistence as can be seen in the third layer of the DACAR conceptual platform. In addition, the lack of public education on the value of internet hygiene and password secrecy stands as a threat to personal health records (PHRs) [19]. The architecture can resolve the security and privacy issues related to PHRs that is dependent on the kind of access level. That's why authorized persons should only have access to personal health records.

3.4 System Security Issues

From the stage in which a message is generated by a user and transmitted by the application software until the message is received by the authorized recipient, the data is exposed to numerous security risks [20]. These risks involve the standard security protection relating to the hardware, secure storage, secure processing, software, human interference, logical problems network issues and natural disasters. Medical information must be transmitted securely; it must have integrity, confidentiality, identification, authentication, authorization and nonrepudiation. Patient's data in transit is subject to security threats as required for other sensitive data. The patient's data is subject to damage, late delivery, attack or loss [21]. The determination of data security is inherent in three possible states, which are:

- Secure processing - Determining if the data are in process.
- Transmission - data move from one location to another.
- Storage - data stored in secure location in each of the above possible states, it is important to determine if the confidentiality, integrity, and availability of the data are compromised [22].

Since always the eHealth system is being accessed either by the doctors, pharmacists, patients or even the eHealth system providers themselves, it is paramount to determine who is authorized to have access to the system and who access what: since attackers can take undue advantage of the system. To this effect, biometrics can be put in place to solve this issue [23, 24]. Biometrics is using either physiological (fingerprints, facial recognition, retina geometry and iris scan) or behavioral (pattering/keystroke, voice scan/speaker and signature/handwriting) traits to identify an individual. Database administrators should determine the rights and privileges of users in other to limit access to sensitive information that could enable patients' activities being tacked [22].

3.5 Traffic Analysis Issues

Attacks based on traffic analysis involve intercepting and examining messages (including encrypted ones), as well as analyzing traffic statistics (such as the number of packets in unit time or the length of data packets) to deduce information from patterns of communication. In an E-health monitoring cloud-based system, users often need to send their health data to remote servers in the cloud through long-distance network connections that are vulnerable to traffic analysis attacks. For instance, in the case of remote patient-monitoring applications, analysis of traffic patterns could reveal the type of sensors mounted on the patient's body, enabling the attacker to infer about the possible health problems of the patient or derive his/her real identity [25, 26].

To mitigate traffic analysis attacks on E-health monitoring cloud-based systems, the authors [26] proposed an E-health monitoring system that ensures minimum service delay and preserves the privacy of users' health data by exploiting geo-distributed clouds. The proposed eHealth monitoring system has two major components: a resource allocation scheme and a traffic-shaping algorithm.

3.6 Operations Security Issues

Operations security includes monitoring, audit, archiving, and back-up in EHR systems. Audit refers to record logs of users' activities. Archiving means to store information in an offline site to be able to restore them when necessary [27]. Monitoring is significant in order to provide security of data transmission through communication channels, identify any suspicious activity and respond to any malicious events. Intrusion Detection and Prevention Systems (IDPS) is one such system [28, 29]. The EHR system should offer mechanisms to back-up patient data for authorized users to ensure patient privacy [30, 31].

4 Conclusion

eHealth systems share clinical data between authorized health stakeholders in order to improve the quality of healthcare delivery and to achieve massive savings. In eHealth systems, privacy and security concerns are tremendously important, since the patient may encounter serious threats if sensitive information is disclosed, stolen or damaged. In this article, we have identified and analyzed critical privacy and security aspects of the EHRs systems, based on a systematic review of a number of research articles. In this review, we encounter and analyze only five key security areas related to eHealth implementation. In the future, we hope to carry out an in depth systematic review concerning the privacy and security in wireless devices connected to EHR systems. The privacy and security issues and their solutions explored by this research could help professionals to implement a sustainable eHealth framework capable of facing cyber threats.

References

1. Jahan, S., Chowdhury, M.M.H.: Assessment of present health status in Bangladesh and the applicability of e-Health in healthcare services: a survey of patients' expectation toward e-Health. World J. Comput. Appl. Technol. **2**(6), 121–124 (2014)
2. Jennifer, M.: E-Health: navigating the internet for health information healthcare. Advocacy White Paper. Healthcare Information and Management Systems Society, May 2002
3. The Department of Health, Australian Government. PCEHR: Personally Controlled Electronic Health Record System Operator: Annual Report 2012–2013
4. HIPPA 1996: US Department of Health & Human Services. https://www.hhs.gov/sites/default/files/privacysummary.pdf
5. European Union (EU) Directive 95. http://www.dataprotection.ie/docs/EU_Directive_95/46/EC_Chapter_1/92.htm
6. Liberati, A., Altman, D.G., Tetzlaff, J., Mulrow, C., Gøtzsche, P.C., Ioannidis, J.P., et al.: The PRISMA statement for reporting systematic reviews and meta-analyses of studies that evaluate health care interventions: explanation and elaboration. Ann. Intern. Med. **151**, W65–W94 (2009)
7. Fernández-Alemán, J.L., et al.: Security and privacy in electronic health records: a systematic literature review. J. Biomed. Inf. **46**, 541–562 (2013)

8. Kashif, H., Wolfgang, L.: Threats identification for the smart internet of things in eHealth ad adaptive security counter measures. IEEE (2015)
9. Benzschawel, S., Silveira, M.D.: Protecting patient privacy when sharing medical data. In: Proceedings of eTELEMED 2011. IEEE (2011)
10. Zhang, R., Liu, L.: Security models and requirements for healthcare application clouds. In: Proceedings of CLOUD 2010, pp. 268–275. IEEE (2010)
11. Lu, R., Lin, X., Liang, X., Shen, X.: A secure handshake scheme with symptoms-matching for mhealthcare social network. J. Mob. Netw. Appl. **16**(6), 683–694 (2011)
12. Liang, X., Li, X., Zhang, K., Lu, R., Lin, X., Shen, X.S.: Fully anonymous profile matching in mobile social networks. IEEE J. Sel. Areas Commun. **31**(9), 641–655 (2013)
13. Guo, L., Zhang, C., Sun, J., and Fang Y.: PAAS: a privacy preserving attribute-based authentication system for eHealth networks. In: IEEE 32nd International Conference on Distributed Computing Systems (ICDCS 2012), pp. 223–233 (2012)
14. Liang, X., Li, X., Shen, Q., Lu, R., Lin, X., Shen, X.S., Zhuang, W.: Exploiting prediction to enable secure and reliable routing in wireless body area networks. In: Proceedings IEEE INFOCOM, pp. 388–396 (2012)
15. Lu, R., Lin, X., Shen, X.: SPOC: a secure and privacy-preserving opportunistic computing framework for mobile-healthcare emergency. IEEE Trans. Parallel Distrib. Syst. **24**(3), 614–624 (2013)
16. Weber-Janke, J.H., Williams, J.B.: Beyond privacy policies-assessing inherent privacy risks of consumer health services. In 2011 Ninth Annual International Conference on Privacy, Security and Trust (PST), pp. 229–237. IEEE, July 2011
17. Idoga, P.E., Agoyi, M., Coker-Farrell, E.Y., Ekeoma, O.L.: Review of security issues in e-Healthcare and solutions. In: 2016 HONET-ICT, Nicosia, pp. 118–121 (2016)
18. Fan, L., Buchanan, W., Thuemmler, C., Lo, O., Khedim, A., Uthmani, O., Bell, D.: DACAR platform for eHealth services cloud. In: 2011 IEEE International Conference on Cloud Computing (CLOUD), pp. 219–226. IEEE, July 2011
19. Habib, K., Torjusen, A., Leister, W.: Security analysis of a patient monitoring system for the internet of things in eHealth. In: Proceedings of the Interational Conference on eHealth, Telemedicine, and Social Medicine (eTELEMED 2015) (2015)
20. Lohr, H., Sadeghi, A.R., Winandy, M.: Securing the e-Health cloud. In: Proceedings of the 1st ACM Interational Health Informatics Symposium, pp. 220–229. ACM, November 2010
21. Frontoni, E., Baldi, M., Zingareti, P., Landro, V., Misericordia, P.: Security issues for data sharing ad service interoperability in eHealth systems: the Nu. Sa. test bed. In: IEEE Interational Carnahan Conference on Security Technology (ICCST 2014), pp. 1–6 (2014)
22. Frank, K., Elaine, L., Martin, F., Yen, Y.Y.: Security, privacy and legal issues in pervasive eHealth monitoring systems. In: IEEE 2008 7th International Conference on Mobile Business (2008)
23. Okoh, E., Awad, A.I.: Biometrics Applications in e-Health Security: A Preliminary Survey. In: Yin, X., Ho, K., Zeng, D., Aickelin, U., Zhou, R., Wang, H. (eds.) HIS 2015. LNCS, vol. 9085, pp. 92–103. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-19156-0_10
24. Chowdhury, M., Islam, R., Gao, J.: Robust ear biometric recognition using neural network. In: IEEE Conference on Industrial Electronics & Applications (ICIEA 2017), Siem Reap, Cambodia (2017)
25. Buttyan, L. Holczer, T.: Traffic analysis attacks and countermeasures in wireless body area sensor networks. In: IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM 2012), pp. 1–6 (2012)
26. Shen, Q., Liang, X., Shen, X.S., Lin, X.: Exploiting geodistributed clouds for a e-health monitoring system with minimum service delay and privacy preservation. IEEE J. Biomed. Health Inf. **18**(2), 430–439 (2014)

27. Chen, Y.Y., Lu, J.C., Jan, J.K.: A secure EHR system based on hybrid clouds. J. Med. Syst. **36**(5), 3375–3384 (2012)
28. Acharya, S., Coats, B., Saluja, A., Fuller, D.: Secure electronic health record exchange: achieving the meaningful use objectives. In: 46th Hawaii International Conference on System Sciences, Wailea, Hawaii, USA, pp. 2555–2564 (2013)
29. Sun, J., Zhu, X., Zhang, C., Fang, Y.: HCPP: cryptography based secure EHR system for patient privacy and emergency healthcare. In: 31st International Conference on Distributed Computing Systems, Minneapolis, MN (2011)
30. Mackenzie, I.S., Mantay, B.J., McDonnell, P.G., Wei, L., Macdonald, T.M.: Managing security and privacy concerns over data storage in healthcare research. Pharmacoepidemiol. Drug Saf. **20**(8), 885–893 (2011)
31. Stingl, C., Slamanig, D.: Health records and the cloud computing paradigm from a privacy perspective. J. Healthc. Eng. **2**(4), 487–508 (2011)
32. Gajanayake, R., Iannella, R., Sahama, T.: Privacy oriented access control for electronic health records. Electron. J. Health Inf. **8**(2), e15 (2014)
33. Clinical Audit Tools: NOCR, UK. http://www.ncor.org.uk/practitioners/audit/clinical-audit-tools/ (visited on 10th July 2017)
34. PEN Clinical Audit Tool: Australia (2008). http://www.pencs.com.au/files/HealthClix_March_2008.pdf

Author Index