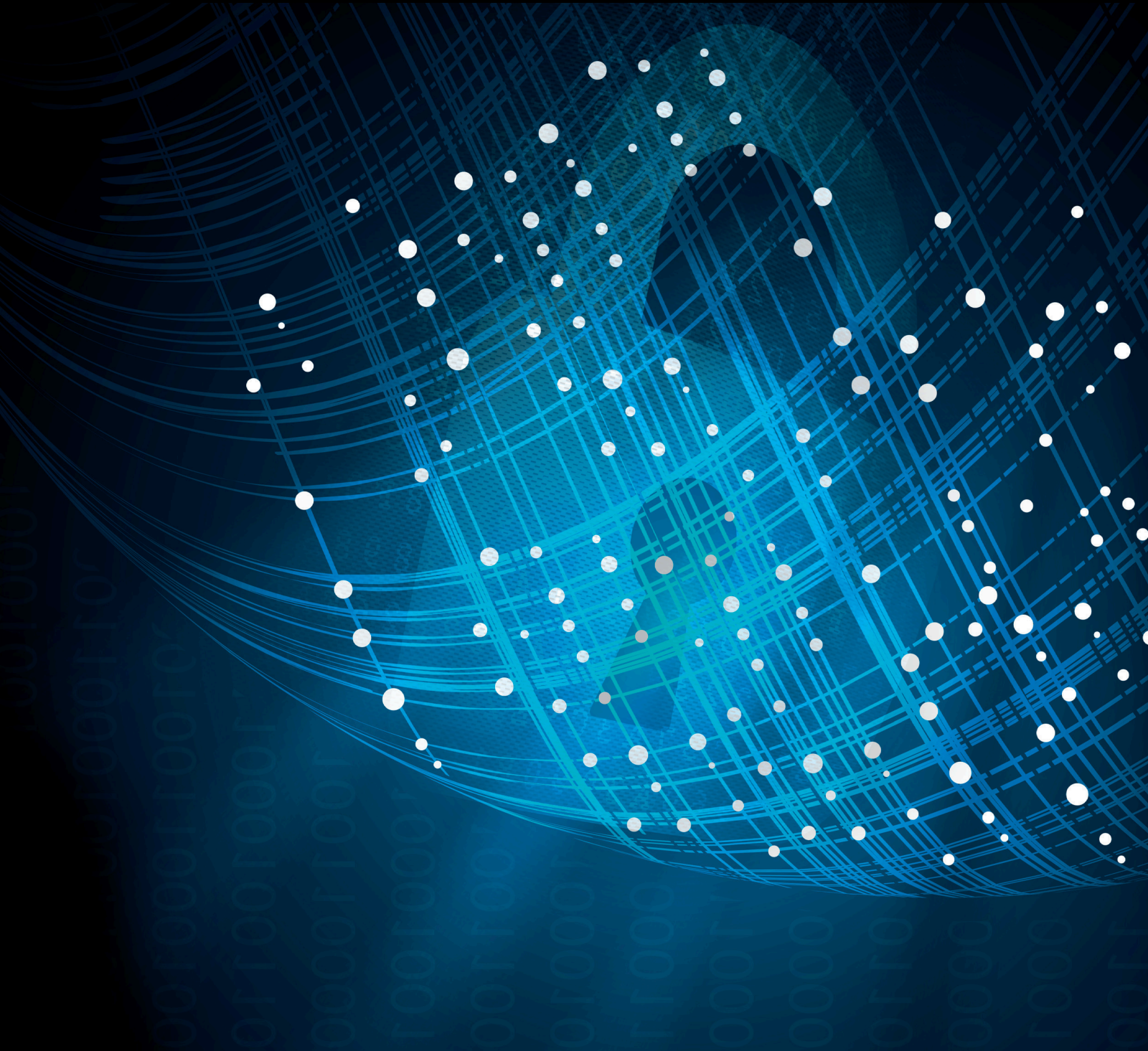


Security and Privacy in Smart Cities

Lead Guest Editor: Chalee Vorakulpipat

Guest Editors: Ryan Ko, Qi Li, and Ahmed Meddahi



Security and Privacy in Smart Cities

Security and Communication Networks

Security and Privacy in Smart Cities

Lead Guest Editor: Chalee Vorakulpipat

Guest Editors: Ryan Ko, Qi Li, and Ahmed
Meddahi



Copyright © 2021 Hindawi Limited. All rights reserved.

This is a special issue published in "Security and Communication Networks." All articles are open access articles distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Chief Editor

Roberto Di Pietro, Qatar

Editorial Board





Ahmed A. Abd El-Latif, Egypt
Mamoun Alazab, Australia
Cristina Alcaraz, Spain
Saud Althunibat, Jordan
Ruhul Amin, India
Maria Azees, India
Benjamin Aziz, United Kingdom
Shahram Babaie, Iran
Spiridon Bakiras, Qatar
Pablo Garcia Bringas, Spain
William Buchanan, United Kingdom
Michele Bugliesi, Italy
Jin Wook Byun, Republic of Korea
Pino Caballero-Gil, Spain
Bruno Carpentieri, Italy
Luigi Catuogno, Italy
Shehzad Ashraf Chaudhry, Turkey
Ricardo Chaves, Portugal
Chien-Ming Chen, China
Rongmao Chen, China
Tom Chen, United Kingdom
Kim-Kwang Raymond Choo, USA
Stelvio Cimato, Italy
Vincenzo Conti, Italy
Luigi Coppolino, Italy
Salvatore D'Antonio, Italy
Paolo D'Arco, Italy
Alfredo De Santis, Italy
Angel M. Del Rey, Spain
Roberto Di Pietro, France
Jesús Díaz-Verdejo, Spain
Wenxiu Ding, China
Nicola Dragoni, Denmark
Wei Feng, China
Carmen Fernandez-Gago, Spain
Mohamed Amine Ferrag, Algeria
AnMin Fu, China
Clemente Galdi, Italy
Dimitrios Geneiatakis, Italy
Bela Genge, Romania
Anwar Ghani, Pakistan
Debasis Giri, India
Muhammad A. Gondal, Oman
Prosanta Gope, United Kingdom

Francesco Gringoli, Italy
Biao Han, China
Jinguang Han, United Kingdom
Weili Han, China
Khizar Hayat, Oman
Jiankun Hu, Australia
Ray Huang, Taiwan
Iqtadar Hussain, Qatar
Azeem Irshad, Pakistan
M.A. Jabbar, India
Mian Ahmad Jan, Pakistan
Rutvij Jhaveri, India
Tao Jiang, China
Xuyang Jing, China
Minho Jo, Republic of Korea
Bruce M. Kapron, Canada
Marimuthu Karuppiah, India
ASM Kayes, Australia
Habib Ullah Khan, Qatar
Majid Khan, Pakistan
Fazlullah Khan, Pakistan
Kiseon Kim, Republic of Korea
Sanjeev Kumar, USA
Maryline Laurent, France
Wenjuan Li, Hong Kong
Huaizhi Li, USA
Kaitai Liang, United Kingdom
Xueqin Liang, Finland
Zhe Liu, Canada
Guangchi Liu, USA
Flavio Lombardi, Italy
Pascal Lorenz, France
Yang Lu, China
Leandros Maglaras, United Kingdom
Emanuele Maiorana, Italy
Vincente Martin, Spain
Barbara Masucci, Italy
Jimson Mathew, United Kingdom
David Megias, Spain
Weizhi Meng, Denmark
Laura Mongioi, Italy
Raul Monroy, Mexico
Rebecca Montanari, Italy
Leonardo Mostarda, Italy




Mohamed Nassar, Lebanon
Shah Nazir, Pakistan
Qiang Ni, United Kingdom
Mahmood Niazi, Saudi Arabia
Petros Nicopolitidis, Greece
Vijayakumar Pandi, India
A. Peinado, Spain
Gerardo Pelosi, Italy
Gregorio Martinez Perez, Spain
Pedro Peris-Lopez, Spain
Carla Ràfols, Germany
Francesco Regazzoni, Switzerland
Abdaloussein Rezai, Iran
Helena Rifà-Pous, Spain
Arun Kumar Sangaiah, India
Neetesh Saxena, United Kingdom
Savio Sciancalepore, The Netherlands
Young-Ho Seo, Republic of Korea
De Rosal Ignatius Moses Setiadi, Indonesia
Daniel Slamanig, Austria
Salvatore Sorce, Italy
Abdulhamit Subasi, Saudi Arabia
Zhiyuan Tan, United Kingdom
Farhan Ullah, China
Fulvio Valenza, Italy
Sitalakshmi Venkatraman, Australia
Jinwei Wang, China
Qichun Wang, China
Guojun Wang, China
Hu Xiong, China
Xuehu Yan, China
Zheng Yan, China
Anjia Yang, China
Qing Yang, USA
Yu Yao, China
Kuo-Hui Yeh, Taiwan
Yong Yu, China
Xiaohui Yuan, USA
Sherali Zeadally, USA
Leo Y. Zhang, Australia
Zhili Zhou, China
Youwen Zhu, China

Contents

Security and Privacy in Smart Cities

Chalee Vorakulpipat , Ryan K. L. Ko , Qi Li , and Ahmed Meddahi 
Editorial (2 pages), Article ID 9830547, Volume 2021 (2021)

Migrating to Zero Trust Architecture: Reviews and Challenges

Songpon Teerakanok , Tetsutaro Uehara , and Atsuo Inomata 
Review Article (10 pages), Article ID 9947347, Volume 2021 (2021)




Efficient Ciphertext-Policy Attribute-Based Encryption Constructions with Outsourced Encryption and Decryption

Hassan El Gafif  and Ahmed Toumanari 
Research Article (17 pages), Article ID 8834616, Volume 2021 (2021)




PurExt: Automated Extraction of the Purpose-Aware Rule from the Natural Language Privacy Policy in IoT

Lu Yang , Xingshu Chen , Yonggang Luo , Xiao Lan , and Li Chen
Research Article (11 pages), Article ID 5552501, Volume 2021 (2021)

An Adaptive Protection of Flooding Attacks Model for Complex Network Environments

Bashar Ahmad Khalaf, Salama A. Mostafa , Aida Mustapha, Mazin Abed Mohammed , Moamin A. Mahmoud, Bander Ali Saleh Al-Rimy, Shukor Abd Razak , Mohamed Elhoseny, and Adam Marks
Research Article (17 pages), Article ID 5542919, Volume 2021 (2021)






Privacy-Preserving Publication of Time-Series Data in Smart Grid

Franklin Leukam Lako , Paul Lajoie-Mazenc , and Maryline Laurent 
Research Article (21 pages), Article ID 6643566, Volume 2021 (2021)

ABSAC: Attribute-Based Access Control Model Supporting Anonymous Access for Smart Cities

Runnan Zhang , Gang Liu , Shancang Li , Yongheng Wei , and Quan Wang 
Research Article (11 pages), Article ID 5531369, Volume 2021 (2021)

Enhancing Digital Certificate Usability in Long Lifespan IoT Devices by Utilizing Private CA

Daiki Yamakawa , Takashi Okimoto , Songpon Teerakanok , Atsuo Inomata , and Tetsutaro Uehara 
Research Article (14 pages), Article ID 6610863, Volume 2021 (2021)

Editorial

Security and Privacy in Smart Cities

Chalee Vorakulpipat ¹, **Ryan K. L. Ko** ², **Qi Li** ³, and **Ahmed Meddahi** ⁴

¹Information Security Research Team, National Electronics and Computer Technology Center, Pathumthani 12120, Thailand

²School of Information Technology and Electrical Engineering, University of Queensland, St Lucia, Queensland 4072, Australia

³Institute for Network Sciences and Cyberspace, Beijing National Research Center for Information Science and Technology (BNRist), Tsinghua University, Beijing 100084, China

⁴IMT Lille Douai, Institut Mines-Télécom, Lille 59500, France

Correspondence should be addressed to Chalee Vorakulpipat; chalee.vorakulpipat@nectec.or.th

Received 28 July 2021; Accepted 28 July 2021; Published 15 August 2021

Copyright © 2021 Chalee Vorakulpipat et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Smart cities have a unique characteristic: integration of deployment of information and communication technology (ICT) services and innovations to handle complex data in storage devices and during citywide transmission. Technologies adopted in smart cities mostly are state of the art and may not be found outside smart cities. Moreover, smart cities today have been used as testbeds or showcases for new technologies for which security and privacy are still uncertain. Consequently, valuable data supported by those technologies can be at risk from various attacks. Security with privacy is an essential key to driving a smart city. Security and privacy issues affect not only a smart city as a whole but also its smart elements including buildings, factories, health, education, and transportation.

Particularly in the smart industry, Internet of Things (IoT), industrial IoT (IIoT), and cyber-physical systems currently lack adequate access control and antimalware and operate 24/7. They are vulnerable to attacks such as DDoS. Regarding personal data protection in a smart context, data confidentiality is vital for privacy law compliance, such as General Data Protection Regulation (GDPR) in the EU and personal data protection acts in many other countries. Cryptography, access control models, and new security architectures for specific contexts can support privacy and law compliance. Consequently, security with privacy is an essential topic for ICT audiences with an emphasis on smart cities.

This special issue aimed to collate original research and review articles emphasizing security and privacy in smart cities. After peer review, we selected seven papers, including six research articles and one review article, which all highlight current issues and trends in the related topics.

As mentioned above, IoT plays a critical role in smart cities; thus, new requirements and risks in IoT security are never ending and can always be discovered. D. Yamakawa et al. investigated the risk associated with using a public certificate authority (CA) in long-lifespan IoT devices. The study addresses the possibility that IoT devices can be disconnected from the network for a very long time, leading to the problem of certificate expiration. The paper proposes a mechanism using certificates issued and signed by a private CA in conjunction with an embedded key used for verifying firmware updates.

There is still room for improvement in terms of access control models in IoT. R. Zhang et al. enhanced the privacy of IoT devices (referred to as subjects) requesting access to other entities (referred to as objects) in the context of smart cities by proposing, implementing, and evaluating a new ABAC-based access control solution called the AB_SAC framework. The AB_SAC framework inherits the features of ABAC such as fine-grained access control, hides the identity of a subject from an entity (referred to as an authorization authority) authorizing access requests, and provides accountability in terms of tracing back the subject identity.

Natural language processing (NLP) can link to an implementation of data privacy policy and IoT. L. Yang et al. presented an information extraction system for purpose-aware rules of privacy policies in IoT to facilitate data privacy compliance and reduce privacy risks due to unfriendly policies. The purpose-aware rules written in natural language are analyzed using semantic role labeling (SRL), whereby meaningful arguments of the main verb are extracted with sequence tagging. In this paper, the actors, actions, manipulated data, and purpose are extracted from

these rules. The authors also propose a method to improve the accuracy of SRL by domain adaptation on a supplementary dataset. The results show that their approach improves the accuracy of extracting the purpose-aware rules.

Privacy preservation for smart grid systems is one of the concerns in smart cities. F. L. Lako et al. addressed the differential privacy (DP) problem in smart grid-based energy delivery networks for publishing aggregate data (e.g., energy consumption of users) while guaranteeing individual privacy. The paper proposes a mechanism called clamping fourier perturbation algorithm (CFPA) that extends or “revisits” the existing FPA mechanism for better privacy protection, improving data aggregation (utility and sensitivity) without disclosing individual data.

Regarding a security intelligence mechanism combatting cyberattacks, B. A. Khalaf et al. propose an adaptive agent-based model called Adaptive Protection of Flooding Attacks (APFA) for protection against Distributed Denial of Service (DDoS) attacks and Flash Crowd (FC) flooding traffics targeting a Network Application Layer (NAL). This model aims to protect the NAL against DDoS and FC flooding by differentiating between DDoS and FC abnormal traffic and then separating DDoS botnets into Demons and Zombies to apply a suitable attack-handling methodology.

The cryptography paper by H. El Gafif and A. Toumanari presents an interesting technique for applying Ciphertext-Policy Attribute-Based Encryption (CP-ABE) as a service. Similar to many CP-ABE-based approaches, the proposed method relies on a Trusted Authority (TA) to issue keys to users. However, in the proposed scheme, the generated keys are separated and kept between a user and the ABE service provider, which allows the user to encrypt the data partially and let the ABE service provider perform the rest. This results in improvements in terms of both computational and communication costs at the user side.

Zero trust architecture (ZTA) has been increasingly mentioned these days, while very few research studies have been done. The last paper, a review article by S. Teerakanok et al., presents challenges and concerns in migrating from perimeter-based security to ZTA. Unlike the legacy network, in which everything inside the internal network is considered trustable, ZTA raises the security level of the entire system by assuming that breaches have been happening everywhere, including inside the corporate network. Based on NIST SP800-207, the authors discuss new threats and challenges in ZTA, including new attack surfaces and vendor lock-in problems. Furthermore, steps and other aspects to consider during the migration process from a legacy network to ZTA are discussed.

Conflicts of Interest

The Guest Editors declare that they have no conflicts of interest regarding the publication of this special issue.

Acknowledgments

We would like to express our appreciation and gratitude to all authors who submitted their works to our special issue. We also cordially thank all reviewers who dedicated their

valuable time to consider the papers and provide constructive comments. Additionally, we would like to thank the Editorial Board of this journal for giving us this excellent opportunity, resulting in success in the publication of this special issue.

*Chalee Vorakulpipat
Ryan K. L. Ko
Qi Li
Ahmed Meddahi*

Review Article

Migrating to Zero Trust Architecture: Reviews and Challenges

Songpon Teerakanok ^{1,2,3} Tetsutaro Uehara ^{1,4} and Atsuo Inomata ^{1,5}

¹Cyber Security Laboratory, Ritsumeikan University, Kusatsu, Shiga 525-8577, Japan

²Faculty of Information and Communication Technology, Mahidol University, Nakhon Pathom 73170, Thailand

³Research Organization of Science and Technology, Ritsumeikan University, Kusatsu, Shiga 525-8577, Japan

⁴College of Information Science and Engineering, Ritsumeikan University, Kusatsu, Shiga 525-8577, Japan

⁵Osaka University, Suita, Japan

Correspondence should be addressed to Songpon Teerakanok; songpon.te@cysec.cs.ritsumei.ac.jp

Received 10 March 2021; Accepted 7 May 2021; Published 25 May 2021

Guest Editor: Qi Li

Copyright © 2021 Songpon Teerakanok et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Zero trust (ZT) is a new concept involving the provisioning of enterprise/organization resources to the subjects without relying on any implicit trust. Unlike the perimeter-based architecture in which any subject behind the wall (i.e., inside the predefined perimeter) is considered trusted, zero trust architecture (ZTA) processes any request and provides a resource to the subject without relying on implicit trust. In this paper, based on NIST Special Publication SP800-207, the concept of ZT and ZTA is introduced. Also, challenges, steps, and things to consider when migrating from the legacy architecture to ZTA are presented and discussed.

1. Introduction

The advancement of technologies brings forth new and more convenient ways of living through the invention of smarter tools and services. The proliferation of cloud technologies and the Internet of things (IoT) bring revolutionary changes to today's IT systems. These changes, however, also come with great challenges. As the IT systems grow, also hackers and malicious actors' skills are adapted and honed rapidly to an astonishing degree. An example of a recent sophisticated attack in December 2020 on SolarWind Orion is given [1], which is an IT monitoring and management software, affecting global victims. This global intrusion campaign was carried out by using a supply chain attack via a trojan (so-called "SUNBURST") backdoor. Using several obfuscating and evasive techniques, the campaign is believed to be the work of a highly skilled threat actor.

1.1. Perimeter Security. "Everything inside the internal corporate/organization network is considered trusted." Until now, this is the concept that we believed and built our

network/system upon. The traditional idea of border protection divides networks into two main areas: internal and external networks. The internal network covers every subject (e.g., devices) within the predefined border based on the devices' physical location while the external network covers the rest. Firewalls, IDS, IPS, and other security controls are usually deployed at the edges (borders) of the corporate/organization network to draw a secure boundary (or "network perimeter") that separates its internal network from the rest of the Internet. This idea forms the basis of perimeter-based network security.

Generally, the legacy method of perimeter-based security allows the use of an implicit trust in which once the subject is authenticated and allowed to enter the internal network, it is considered trusted. This allows a malicious (compromised) subject to perform further lateral movement and roam freely once it infiltrated the internal network [2].

1.2. Zero Trust Concept. As technologies continue to advance, the demands and lifestyles of users are also changing rapidly. Cloud technologies offer us new ways to access

services and resources anywhere and anytime with high cost performance. Nowadays, people are no longer required to work from their office/workplace; instead, they can work remotely from anywhere as long as all resources needed to perform their job are available. With the emerging of a Bring Your Own Device (BYOD) policy and the current COVID-19 pandemic situation, remote working and working from home (WFH) have become a common thing (so-called “new normal”) for many organizations. For example, in November 2020, Square Enix, a large Japanese video game company, offered their employees an option to permanently work from home [3, 4] which greatly demonstrates a paradigm shift in this regard.

Now, the question lies in how this change in paradigm affects an organization in terms of security. The legacy perimeter-based network security is considered insufficient since the users are currently allowed to work remotely from any place which may no longer be located inside the secured perimeter. Therefore, it has become very difficult to define or draw the exact borders/perimeters, let alone securing them.

These problems brought about the concept of “Zero Trust” (ZT) in which an enterprise must assume that there is no implicit trust in every subject. In the ZT security model, the enterprise-owned environments are considered no more trustworthy than any nonenterprise-owned environment [2]. More details regarding ZT and zero trust architecture (ZTA) are provided in Section 2.

In this paper, we discuss the adoption of the ZT concept and ZTA to the enterprise/organization. Since ZT is an evolving concept which cannot and will not be completed by just buying and replacing all the network equipment, there are many factors an organization needs to consider while migrating from the legacy perimeter-based model to ZTA. In this paper, details of threats and challenges in transitioning from traditional network to ZTA are introduced. Furthermore, some key factors and basic guidelines for ZTA migration are presented and discussed.

The rest of this paper is organized as follows. Section 2 presents a brief introduction to the concept of zero trust (ZT) and zero trust architecture (ZTA). In Section 3, we discuss security requirements needed for ZTA deployment and new threats against ZT-based systems. Section 4 presents and discusses processes and factors needed to be considered to successfully migrate to ZTA. Next, the remaining challenges, details, and steps for ZTA implementation are presented and discussed in Sections 5 and 6, respectively. Finally, we summarize this paper in Section 7.

2. Zero Trust Architecture

Zero trust architecture (ZTA) adopts an idea in which all subjects are implicitly considered untrusted no matter where they are located (either internal or external), which is the opposite of how perimeter security works. However, it does not mean that there is no trust at all in ZTA. In this new way of thinking, typically, a subject earns trust from the system on a particular request/transaction by proving itself through authentication and authorization.

Enforcing the authentication and authorization process on every request/transaction gives the system the ability to granularly control and adjusts the security level required to access a particular resource.

In the following subsections, components, common models of ZTA, and a brief introduction to the trust algorithm are presented and discussed, respectively.

2.1. Components. There are 5 major logical components in ZTA as displayed in Figure 1: subject, resource, policy decision point (PDP), policy enforcement point (PEP), and supplement [2]. Subject refers to a user or any device requesting access to the enterprise resources. As the name suggested, resource refers to the corporate/enterprise resource being requested by a subject. The resource can be either single or multiple pieces of resources depending on the content of the request.

A policy decision point (PDP) is responsible for deciding to allow or deny access to the enterprise resource and establish or terminate the communication between a subject and a resource being requested. PDP can be broken down into two components: policy engine (PE) and policy administrator (PA) which are responsible for decision making and communication management, respectively.

A subject sends a request to access an enterprise resource which ends up sending to the policy enforcement point (PEP). The PEP forwards the request to PDP. After the PDP decides what to do with the request, it then issues a command to PEP to enable or terminate the communication between the subject with the resource. As we will see, PEP acts as a gate between the subject and resources. Not only controlling the flow of the communication but PEP is also responsible for monitoring the network traffic going between the subject and the requested resource.

Lastly, supplement helps to provide useful information (e.g., threat intelligence information and network/system logs) to the PE. This allows PE to make more accurate and correct decisions (less false positive and false negative) which also ends up enhancing the overall security of the system.

2.2. ZTA Models. NIST SP 800–207 [2] classifies ZTA deployment into 4 types, i.e., device agent-based, enclave-based, resource portal-based deployments, and lastly the ZTA deployment using device application sandboxing, depending on how resources are managed and safeguarded. There are both agent and agent-less approaches with PEP acting as a gateway to the enterprise resources. The PEP can be attached to a single or multiple resources or act as a portal to all enterprise resources.

Also, there is a special variant of an agent-based model utilizing sandboxing, which could be a virtual machine (VM) or containers such as Dockers. The goal of utilizing a sandbox in this model is to prevent the trusted application from any malicious activities that may originate from a possibly compromised subject (host). Please refer to Section 3.2 of [2] for further information regarding the ZTA deployment model.

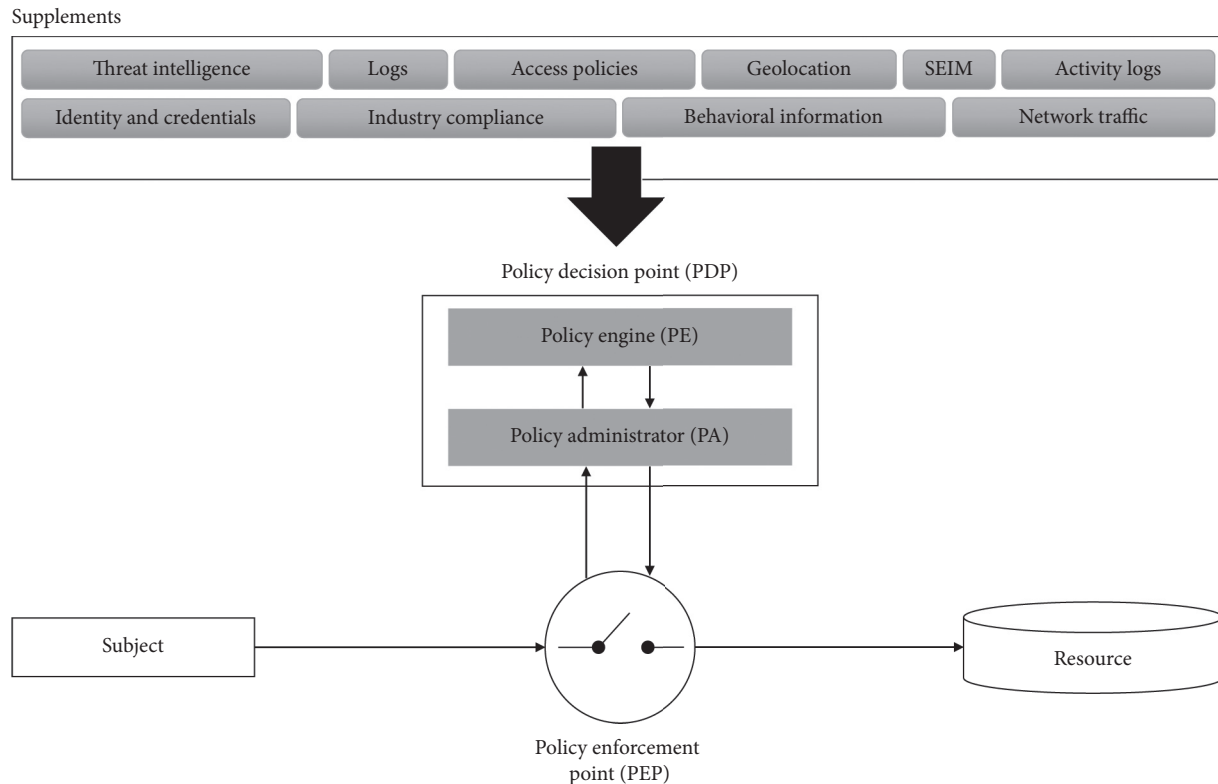


FIGURE 1: Logical components of the zero trust architecture (ZTA).

2.3. Trust Algorithm. While the policy engine (PE) can be considered a brain of the ZTA, inside this brain holds a crucial thought process known as trust algorithm. The trust algorithm (TA) is used by PE to decide whether to grant or deny access to the requested resource. To draw such a decision, PE generally incorporates several pieces of information from various sources which may include access request (i.e., request content), subject information (corresponding to the subject database), asset database (known enterprise assets including BYOD devices), resource requirements, security and network traffic logs, and threat intelligence information.

There are some trust algorithms proposed in the literature. Chen et al. [5], in 2020, introduced the use of behavior-based anomaly discovery technique incorporated with user and devices identities, user behavior, terminal security status, and system behavior information to perform the trust assessment process. The proposed method utilizes the hierarchical trust level-based access control, in which the resource access is granted only when the trust level of a subject reaches a certain level. This approach offers a dynamic access control with fine-grain tuning capability using the required trust level/threshold.

Similarly, in October 2020, Yao et al. [6] proposed a dynamic access control and authorization system for ZTA. The authors proposed a trust-based access control (TBAC) which is based on the calculating of behavior trust (BT) score for each user. In case that BT exceeds the minimum trust score required to access the resource, access permission is granted. In this method, user behaviors affecting the BT score

calculation are login behavior (e.g., login method and time), network behavior (e.g., amount of TCP traffic), and operational behavior (e.g., resource name and access history).

Lastly, Vanickis et al. [7], in 2018, proposed a policy enforcement framework for the zero trust network (ZTN). In this paper, FURZE, a framework for risk adaptive access control (RAdAC) is proposed. Using the RAdAC-based approach allows the system to utilize both operational need and security risk to accurately and dynamically grant or deny access to the enterprise resource. Furthermore, the authors introduce two domain-specific policy languages PAROLE (similar to XACML [8]) for configuring access control to network resources and FACL (firewall access control list) which is designed for expressing firewall-specific filtering rules and configurations.

3. Security in ZTA Environment

New technology comes at a security cost. Cyberattacks are continuously being refined and becoming more sophisticated. To adopt the zero trust concept to the organization, there are things to look out for and to be aware of. In this section, threats and challenges that might come with ZTA are introduced and discussed. Furthermore, we discuss some requirements needed to successfully deploy ZTA in practice.

3.1. Threats. Most of today's information system has its soft spots whether they are technical (e.g., system designs and configurations) or nontechnical (e.g., human). Taking

advantage of these weaknesses, adversaries prey on an enterprise by exploiting these vulnerabilities. To defend against such attacks, the organization should first get a clear grasp of what the attack surface looks like. Understanding the attack surface allows us to have insights on where (in which way and how) the attack might be carried out. In the following subsections, we discuss the new attack surface and some threats associated with ZTA.

3.1.1. New Attack Surface. As shown in Figure 1, the policy decision point (PDP) and policy enforcement point (PEP) are the core parts of ZTA. It is responsible for making any decision whether to grant or deny access to the enterprise resource and also managing the connection between the subject and the resource. In ZTA, these core parts (i.e., PDP and PEP) can be new targets for adversaries. Unlike the traditional perimeter-based architecture, ZTA may suffer from attacks on these core parts such as DDoS, route hijacking, or supply chain attacks on PDP. By disrupting PDP and PEP operations, the operations of the enterprise network may come to a halt. Furthermore, in case of PE being compromised, it may cause serious harm (e.g., data tampering, or leakage) to the organization.

Furthermore, since there is no longer a wall to protect enterprise assets, all assets have the possibility of being targeted. Compromised assets or accounts with a high level of access privilege, especially ones with access permission to the resources interested by an attacker (e.g., commercial, and financial information), are likely to be primary targets of attacks.

3.1.2. Denial-of-Service (DoS) and Network Disruption against PDP. As mentioned earlier, the PDP, consisting of policy engine (PE) and policy administrator (PA), can be a new target of attack since every decision to grant or deny access to the resource is determined by PDP. Corrupting or disrupting PDP will also greatly affect such decision process which can result in a halt in operations. One way to mitigate such network disruption attacks is to put PDP (i.e., PE and PA) on the secured cloud environment which is more resilient in the face of such an attack.

3.1.3. Unauthorized/Unapproved Changes in PDP. According to NIST SP800-207, the system administrator may perform unauthorized changes or accidentally creates misconfigurations which may disrupt or create vulnerabilities in the system. A misconfigured or compromised PE might grant access to some restricted resources that would otherwise not be allowed. Also, a compromising PA may allow adversaries to bypass PE's decision process and access the enterprise resource directly.

Mitigating such risks, as suggested in NIST SP800-207 [2], involves the logging and monitoring of PDP activities. Moreover, PE and PA should be properly configured with all changes being documented. Lastly, both PE and PA should be subjected to audit.

3.1.4. Credential Theft. In ZTA, all assets are not implicitly trusted and have the possibility of being attacked based on their importance and the importance of the information they hold. Some assets or accounts are more likely to be targeted. Compromising an account or asset is not something new and is not unique to the zero trust architecture. With the new BYOD policy, it may be even easier for an attacker to successfully compromise a BYOD asset. Since BYOD devices are not controlled by the enterprise, they may not receive the latest security patch or may not have any antimalware mechanism installed. Since there is no such wall of protection as in the perimeter-based architecture, a well-developed ZTA should prevent or hinder compromised assets or accounts from accessing enterprise resources.

One way to mitigate such a problem is to monitor the subject's behavior which might include login history and pattern, duration, and resource access pattern. A subject accessing any resource outside its normal access pattern may raise a flag which can lead to a more thorough investigation. A good example of deploying a behavior-based approach in ZTA is presented by Yao et al. [6] in 2020. In [6], the subject's behavior is continuously observed and calculated into a behavior trust (BT). Access to a resource is granted only if BT exceeds the trust threshold (TT) which may change dynamically depending on the environment.

3.1.5. Network Traffic Monitoring and Inspection. ZTA relies on end-to-end communication which usually contains encrypted information. Some third-party software/services are fully encrypted making it very difficult or impossible to perform full packet inspection. This leaves the enterprise no choice but to perform packet analysis based on the metadata of the packet. However, it is suggested in [2] to incorporate machine learning techniques to help to analyze the encrypted traffic for better efficiency.

3.2. Requirements. There are three primary requirements needed to be fulfilled to successfully deploy and implement ZTA.

3.2.1. Granular Data Visibility, Access Control, and Data Protection. One key point of ZTA that differentiates it from the legacy perimeter-based model is how it protects the resources. ZTA protects individual resources while perimeter-based architecture protects all enterprise resource as a whole. In the case of breaches, perimeter-based architecture tends to suffer greater damage comparing to ZTA. To protect individual resources, enterprises are required to adopt the data-centric approach [9] including data/resource discovery, tracking, and analysis. Enterprises are expected to know what kind of data and resources they are holding, how they are protected, and who and when accessed these resources.

Enterprises should exercise the least privilege policy [10] so that there is only the least amount of data and resources available to a subject. The least privilege policy can be applied with dynamic trust-based access control (see [5, 6], for example) to provide granular visibility of data. Also, an

individual resource should be protected at the border of that resource (using resource gateway, for example).

3.2.2. No Implicit Trust. “Guilty until proven innocent” [10, 11] is the term that best describes ZTA. A subject is trusted conditionally (i.e., only when some conditions are met). To be trusted, a subject is required to perform some action (e.g., authentication) to earn it. Nothing in ZTA has implicit trust; the only way to gain trust is to earn through verification.

The verification comes not only in the form of authentication but also in policies and requirements. All subjects are required to meet the minimum security requirements and access policies. For example, the agent, in the agent-based model, may drop the request automatically should the device performing a request is not patched or not meet the minimum security requirements set by the enterprise.

Not only trust can be earned but can also be lost. For example, the PDP may decide to reduce the level of trust given to a subject if it performs some suspicious activities such as requesting a resource outside the subject’s scope of permission or performing multiple login attempts in a very short period. Lastly, trust is not a constant value, and its value should be decreased over time. It means that even though a subject can access a certain resource at the moment, the same right to access this particular resource is not guaranteed in the future. The re-evaluation of a subject’s trust is needed. An enterprise is recommended to apply the decaying property to the trust algorithm in its zero trust system.

3.2.3. Continuous Authentication and Monitoring. One important requirement for building a zero trust system is continuous monitoring and evaluation. In ZTA, a subject earns trust separately for each of its requests. In ZTA, requesting the same resources at different times should require verification and re-evaluation of trust. As mentioned earlier in the previous subsection, trust can be earned and also can be lost. Therefore, continuous monitoring of the subject activities is required. Continuous monitoring helps maintain the current level of trust a subject has while detecting inconsistencies or anomalies in a subject’s behavior (e.g., changes in access pattern and failed login attempts) which are crucial information in re-evaluating trust.

Continuous authentication and monitoring do not mean a user has to type in his/her password for every single request. The process of continuous authentication should be done in a nonobtrusive manner [9] to increase practicability. A subject may perform multifactor authentication (MFA) at first and maintain its trust level by allowing the system to observe and analyze its behavior (e.g., request pattern, keystroke, geolocation, or network traffic).

An early example of continuous monitoring is the work of Brosso et al. [12] in 2010. In [12], a continuous authentication system based on user behavior is proposed. The proposed method utilizes user behavior in determining the level of trust for each user. By incorporating the neuro-fuzzy

logic technique to continuously update the user behavioral database, the author introduces the new method to calculate trust with better accuracy.

4. Migrating to ZTA

Deploying and implementing ZTA is a multiple-step and continuous process that cannot be achieved by replacing all tools and equipment with new ones. To introduce the zero trust concept to existing perimeter-based architecture, there are several steps and factors to concern. Figure 2 shows the ZTA deployment cycle which is originally based on the NIST risk management framework (RMF) [2, 13].

In this paper, we divide the migration process into three major steps. The following three subsections explain the details of each step.

4.1. Assessment. Assessment involves several things including identifying the actors and assets of the enterprise. The enterprise should have a clear grasp of its subjects including both human and nonperson entities (NPEs). It should be able to identify and monitor both enterprise-owned and nonenterprise-owned assets, including both hardware and digital artifacts (e.g., software and digital certificates). Furthermore, the enterprise is expected to have the ability to configure, manage, and observe the current state of the asset.

Dealing with “shadow IT” and nonenterprise-owned assets such as BYOD devices is certainly more challenging compared with managing enterprise-owned assets. However, the enterprise should try its best to discover and observe these assets [2]. Lastly, depending on the critical mission of the enterprise, it may also need to list and categorize its high-value assets (HVAs) and all relevant processes.

4.2. Risk Assessment and Prioritization. An enterprise should identify and rank its business process based on the criticality and importance of its mission. By studying the risk impact and performing some prioritization, the enterprise may decide to start migrating its very first business process that has relatively low-risk to ZTA and then continues later with a more critical business workflow/process after experiencing and learning from the first transition.

Once the candidate business process has been selected, now, it is time to create policies for this candidate process. In this step, all resources used or affected by the candidate process should be identified. This allows the enterprise to know precisely what resources are involved with the migration process. Lastly, in the case of a criteria-based trust algorithm, a set of criteria is determined. On the other hand, regarding the score-based trust algorithm, trust/confidence level weights of each resource used in the candidate process are initially defined. Both the mentioned set of criteria and the weights are expected to change during the initial tuning period and may also change over time to ensure its effectiveness and practicality.

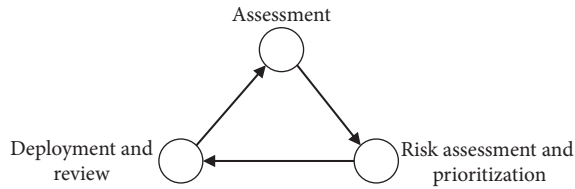


FIGURE 2: Development cycle of ZTA (based on NIST SP800-207 [2]).

4.3. Deployment and Review. After the enterprise determines the candidate business process to migrate to ZTA, the actual deployment begins. Implementation of a new ZTA-based business workflow must follow the security policies developed in the earlier phases. Generally, the deployment always associates with the logging and monitoring process. Analyzing logged and monitored information (e.g., request pattern, login attempts, and communication patterns) allows the enterprise to ensure that the new ZTA-based business process works effectively as intended.

In this step, results including mistakes during earlier phases are collected. The organization analyzes and learns from mistakes and makes changes (if necessary). The results and changes should be properly documented. With each migration, the enterprise/organization gains more confidence and can choose a more challenging workflow for its next migration cycle.

4.4. Things to Consider during ZTA Migration Process. There are few things that an enterprise needs to pay close attention to during migration to ZTA.

4.4.1. Changes Procedure. When making changes during the migration process, all changes are needed to be properly documented. Since migrating to ZTA is the change from the lowest level of the architecture, i.e., the fundamental concept, therefore, there is a possibility that the enterprise needs to perform several changes (both hardware and software) or even redesigns some of its workflows from scratch. Therefore, a well-designed change management policy and plan are required. Well-documented change procedures help the enterprise keep track of what is changed and help provide useful information once something goes wrong.

4.4.2. Risk Management. Risk management is not something new or unique only to ZTA. Generally, the enterprise is required to identify and assess the risk associated with its missions or business workflows. Deployment and implementation of ZTA may cause the enterprise to reidentify and re-evaluate risks associated with any business process involved. NIST risk management framework, in the Special Publication SP800-37 [13], provides a guideline in this regard which is also applicable with ZTA.

4.4.3. Identity Management. ZTA involves and relies heavily on the subject identity to perform subject provisioning. To grant a resource to a subject, the policy engine needs several

pieces of information including identity and credential information to perform a decision. Therefore, the enterprise should also make changes to its ICAM [14] policy to include and support new ZTA-based workflows. NIST provides recommendations regarding digital identity management through its special publication SP800-63-3 (i.e., Digital Identity Guidelines [15]) which are also applicable to ZTA.

4.4.4. Laws and Regulations. To migrate to ZTA, the enterprise should make efforts to ensure that new ZTA workflows comply with laws and regulations, for example, Health Insurance Portability and Accountability Act (HIPAA) [16], Payment Card Industry Data Security Standard (PCI-DSS) [17], and General Data Protection Regulation (GDPR) [18].

5. Challenges in ZTA

To successfully implement a good zero trust system, there are currently many difficulties an enterprise needs to overcome. In this section, we discuss some current problems and some remaining challenges that hinder the deployment of ZTA in practice.

5.1. Vendors Lock-In and Interoperability. The vendor lock-in problem is not something new. We experienced this problem before in both IoT and cloud platforms. In an IoT system, we usually find some of our devices unable to communicate and interoperate with devices from different vendors. For example, some of Google's IoT devices (e.g., Nest Thermostat) are not compatible with Apple's IoT ecosystem (i.e., HomeKit).

Regarding cloud platforms, the vendor lock-in problem refers to the restriction enforced by the cloud provider to prevent or discourage users from switching a service. Generally, cloud providers encourage new users to easily sign up and offer their services at a low initial price. However, once the user decided to scale the service, the price starts to grow exponentially. At this point, the user may consider switching to other cloud providers with better offers. This is also when the vendor lock-in problem happens. A user switching to the other providers means the company losing its revenue; therefore, the cloud provider may try to impose some technical difficulties, legal restrictions, or some additional fees to discourage the user from leaving.

Some providers are utilizing proprietary technologies which also make the migration even harder. Opara-Martins et al. [19] provided a comprehensive analysis highlighting the impact of vendor lock-in problem in business perspective. Surprisingly, the study shows that many customers including decision-makers of many companies lack sufficient knowledge and awareness regarding proprietary technologies that might prevent or restrict interoperability and portability when procuring cloud services from providers.

Concerning vendor lock-in problems in ZTA, zero trust is a new evolving concept which is not fully matured at the

moment. Therefore, there is no single-vendor solution available from any vendor at the moment. For many existing enterprises and organizations, migrating to ZTA is a continuous process that might take a long time. Therefore, some organizations may prefer purchasing components from different vendors according to their needs instead of purchasing from any single vendor.

However, to avoid vendor lock-in problems, some standards to support interoperability between devices are needed. As we already know, many vendors usually rely on proprietary APIs rather than the standard. This makes it very difficult for two devices from different vendors to work together smoothly since they follow different protocols and API. Also, in the case of partner companies changing their API behavior, vendors are required to make changes to support them as well.

There is no silver bullet to this problem. The problem of vendor lock-in is simple but hard to solve and will continue to exist. To select suitable technologies, platforms, or infrastructure to support ZTA, an enterprise needs to first identify dependencies in their IT system. If its current IT systems are designed to operate or rely on legacy technologies, there might be left with only limited options to choose from. Furthermore, if the current IT systems are compatible with only limited technologies and platforms, the enterprise may consider upgrading the legacy systems before migrating to ensure the compatibility and interoperability of these IT systems. These upgrades will help to prevent future loss should the company decide to switch from one vendor to another.

5.2. Proprietary Data Formats and Need for Standardization. The decision-making process of ZTA is done by the policy decision point (PDP). This process requires information from various sources to draw the final decision on whether to grant or deny access to the enterprise resource. Although some of them already have standards for exchanging information, for instance, STIX [20] and TAXII [21] for sharing threat intelligence information, there is still no common standard for some of them. In the case of a provider has encountered some security or technical issues and the enterprise needs to find a quick replacement, it is very difficult to do so without paying a high price [2].

5.3. Avoiding User Disruptions. One of the biggest challenges in implementing ZTA is to avoid disrupting users during deployment. At the start of each migration cycle, an enterprise may start the migration process by introducing a new ZTA-based approach, which is designed to replace the legacy system, inside the perimeterized zone and encourage its users to utilize this new ZTA-based system. The enterprise may gradually impose technical restrictions to the old method at the same time to encourage users to switch to the new one. As more users utilize the new system for an extended period without any problem and exception, the enterprise may decide to move the new workflow from the perimeterized zone to the unprivileged network. Finally, the enterprise needs to perform a final clean-up to remove any

decommissioned or deprecated service to complete the migration process.

5.4. Trust Level and Resource Classification. Requesting resources in ZTA involves calculating a trust level from credentials and information provided by the subject at the time of request and comparing it to the predetermined trust level required to access the particular resource. Determining an appropriate level of trust for each resource is a challenging task. The enterprise needs to determine an appropriate level of trust which is not too high or too low for each resource. Too high makes the resource too hard to access which may also end up hindering the workflow, while too low means the resource is too easy to access and less secure.

5.5. Dealing with Unmanaged Devices. Some companies, such as Google (see [22–27] and Section 6.4 for further details), decide to implement ZTA by allowing only corporate-owned or managed devices to access corporate resources. However, some companies may exercise a BYOD policy that might not allow them to freely monitor, control, or install any agent software on employee's personal devices due to privacy issues. Therefore, allowing unmanaged devices to securely access corporate resources without imposing too many restrictions is also one of the remaining challenges in deploying ZTA.

5.6. Improving Trust Algorithm. Technically, the trust algorithm (TA) is considered the thought process inside the brain (i.e., PDP) of ZTA providing PE an ability to accurately decide whether to grant or deny access to all incoming requests. TA incorporates information from various sources including threat intelligence, SIEM logs, network traffic, subject's geolocation, user's identities, and credentials.

Each piece of information is not equally important; some information, such as user credentials, are more important and are weighted more, comparing to other factors such as network traffic, in calculating a trust level of a subject. Currently, there is no optimal solution, guideline, or reliable approach in weighting such factors; the enterprise implementing ZTA needs to continuously observe and adjust these parameters over time to ensure it functions accurately as intended.

TA may consist of both static rules (e.g., deny all access from a known compromised or malicious device) and a dynamic decision mechanism that calculates the possibility of a subject being malicious based on information available at the time of the request. This dynamic decision-making approach usually involves the use of machine learning (ML) allowing TA to heuristically improve its decision-making capability over time. As mentioned earlier, TA is considered a crucial part of ZTA. Inaccurate or tampered results from TA can greatly affect PDP's final decision which might end up allowing a malicious subject to access corporate resources (i.e., false positive) or deny legitimate users from accessing the resources they need (i.e., false negative). Therefore, an

enterprise is required to put efforts into fine-tuning TA from time to time to maintain and improve its functionality.

Finally, TA incorporates information from many sources which usually come in different formats. Some information is redundant, while some are poor in quality or irrelevant. Therefore, this information is needed to be normalized, filtered, and then correlated to improve the overall efficiency of TA.

6. Implementing ZTA

In this section, we discuss generic details, steps, and information regarding the implementation of ZTA. Every ZTA implementation is a unique and completely different journey for each organization; therefore, we will only discuss generic details and common issues in putting ZTA into practice. There are 3 major steps in implementing ZTA. Sections 6.1–6.3 explain these steps in detail. Lastly, in Section 6.4, we briefly discuss a real case study of ZTA implemented by Google, called “BeyondCorp.”

6.1. Identify Devices and Users. ZTA involves heavily in managing access control to the corporate resources. Thus, the first step to implement ZTA is to identify what resources and assets the company owns, including corporate-owned and possibly BYOD devices. Making device inventory and user database helps the company keep track of this information. Then, the company needs to implement two mechanisms to identify devices requesting resources and to identify the user sending the request.

6.2. Removing Implicit Trust. Next, we remove implicit trust from all related subjects. All devices located both inside and outside the corporate network are treated the same as devices connected from the outside (external network). An enterprise may also utilize segmentation techniques, such as VLAN, to temporarily separate devices into safe and quarantine VLAN for better management.

6.3. Externalizing Workflows. In this step, candidate applications and workflows are externalized via internet-facing policy enforcing point (PEP) (see Figure 1). All requests passing the authentication and authorization processes are delegated to the backend server. In this step, policy engine (PE) performs a service-level authorization to grant or deny access to corporate resources. All communications in this step are encrypted.

6.4. A Case Study of Google’s BeyondCorp. An excellent example of ZTA implementation is Google’s BeyondCorp. In this section, we discuss how Google achieved its idea and implementation of the zero trust model according to steps explained earlier in Sections 6.1–6.3. Figure 3 shows the overview of all components in Google’s BeyondCorp.

First, in BeyondCorp, Google allows only managed devices to communicate and send requests for corporate resources. A device inventory database responsible for storing device-related information was created to keep track of all managed device states and all relevant information. Also, digital certificates and some encryption keys are stored in TPM or qualified applications to help in securing and identifying these devices.

To identify a user, a multifactor authentication (MFA) is required. MFA can be performed using a managed device either via a single sign-on (SSO) platform against the pre-determined user/group database or via RADIUS using 802.1x protocol.

Next, to remove implicit trust from the legacy network, some managed devices located in the private privilege network inside the Google building are then moved to an unprivileged network. These devices are now treated the same as devices connected from the external (outside) network. Utilizing VLAN, managed devices authenticated via RADIUS server using 802.1x protocol are assigned to a different virtual network. On the other hand, all unrecognized and unmanaged devices are automatically assigned to a guest or quarantine network for further remediation.

To externalize a workflow making it accessible from anywhere, Google deploys an internet-facing access proxy which is similar to policy enforcement point (PEP) in NIST SP 800–207 [2]. Applications in BeyondCorp are registered with public DNS having their CNAME pointing to Google’s access proxy server. Hence, all requests coming from both public (external) and private networks are sent to the access proxy. After passing the authentication, the access proxy then asks the access control engine to verify and authorize/deny the request.

The access control engine (ACE, for short), equivalent to policy enforcement (PE) in [2], is considered the heart and the brain of Google’s BeyondCorp. ACE incorporates various sources of information to decide if the requesting user and device are trustworthy and have the right to access the resources being requested or not. Every decision is made on a per-request basis.

Trust inference, similar to the trust algorithm in [2], is a logical component that continuously computes and updates the trust level of all subjects in real-time. An access request is authorized by the ACE only when all predefined rules/conditions are met, and the trust level of the requesting user is higher than the minimum required trust level defined for the resource being requested.

Regarding migration to ZTA, Google first identifies candidate workflow by performing workflow qualification, job function analysis, identifying candidate population, and traffic analysis via both privileged network traffic sampling and unprivileged network simulation. The migration starts with low-risk migration and moves to more critical workflows with higher risk once the company accumulates enough confidence in their migration process. More details regarding Google’s BeyondCorp are provided in [19, 22–25].

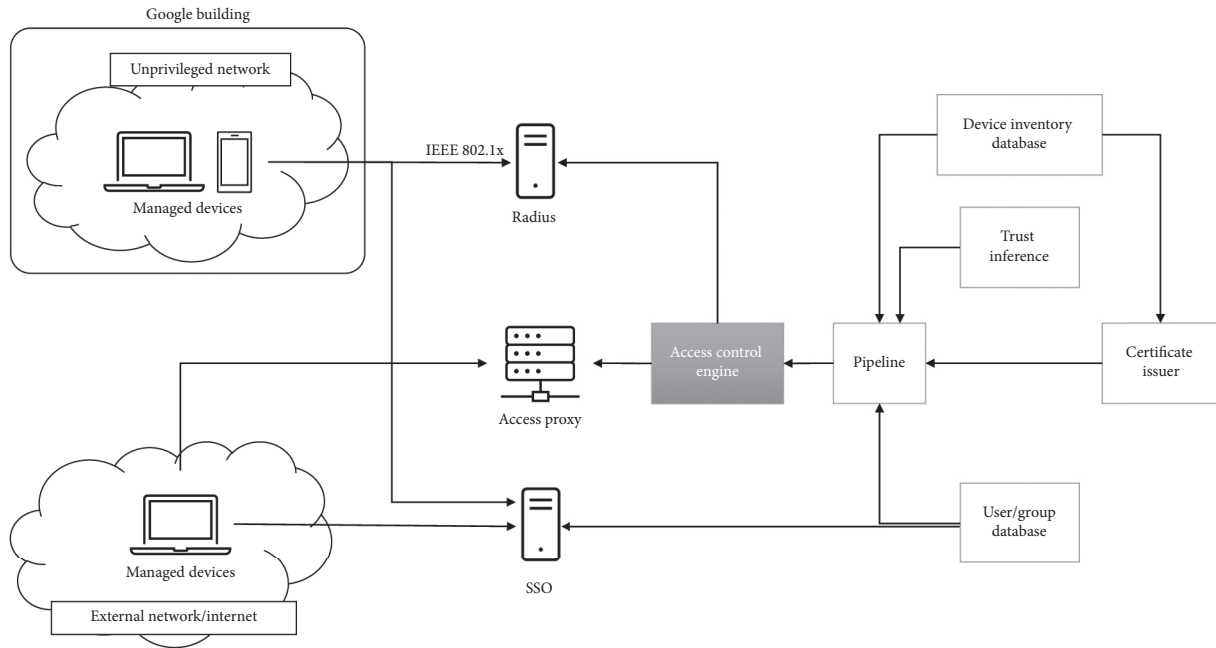


FIGURE 3: An overview of Google's BeyondCorp (original image from [22]).

7. Conclusions

In this paper, we discuss the concept and application of the zero trust architecture. Some challenges in ZTA, including lacking standardization and vendor lock-in problems, are introduced and discussed. Lastly, brief information focusing on steps and things to consider regarding migration from perimeter-based architecture to ZTA is presented.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

- [1] FireEye, "Highly evasive attacker leverages solarwinds supply chain to compromise multiple global victims with sunburst backdoor," 2020, <https://www.fireeye.com/blog/threat-research/2020/12/evasive-attacker-leverages-solarwinds-supply-chain-compromises-with-sunburst-backdoor.html>.
- [2] S. Rose, O. Borchert, S. Mitchell, and S. Connelly, *Special Publication 800-207: Zero Trust Architecture*, National Institute of Standards and Technology, Gaithersburg, MD, USA, 2020, <https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-207.pdf>.
- [3] Square Enix, "Square enix to make work from home permanent as of December 1 -mostly home-based hybrid model to strike balance between flexibility and manageability," 2020, <https://www.jp.square-enix.com/company/en/news/2020/html/df9995782da2d516db9ebac425d02d4019665f70.html>.
- [4] C. Page, "Square enix expects 80% of employees to work from home permanently," 2020, <https://www.forbes.com/sites/carlypage/2020/11/25/square-enix-expects-80-of-employees-to-work-from-home-permanently/?sh=60d3ed42294c>.
- [5] B. Chen, S. Qiao, J. Zhao et al., "A security awareness and protection system for 5G smart healthcare based on zero-trust architecture," *IEEE Internet of Things Journal*, 2020.
- [6] Q. Yao, Q. Wang, X. Zhang, and J. Fei, "Dynamic access control and authorization system based on zero-trust architecture," in *Proceedings of the 2020 International Conference on Control, Robotics and Intelligent System*, Xiamen, China, October 2020.
- [7] R. Vanickis, P. Jacob, S. Dehghanzadeh, and B. Lee, "Access control policy enforcement for zero-trust-networking," in *Proceedings of the 2018 29th Irish Signals and Systems Conference (ISSC)*, Belfast, UK, June 2018.
- [8] OASIS, *eXtensible Access Control Markup Language (XACML)*, OASIS, Burlington, MA, USA, 2013, <http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.html>.
- [9] B. Embrey, "The top three factors driving zero trust adoption," *Computer Fraud & Security*, vol. 2020, no. 9, pp. 13–15, 2020.
- [10] T. Pandit, "Cloud desktops further the shift to zero trust," 2019, <https://www.workspot.com/blog/cloud-desktops-zero-trust/>.
- [11] Information Security Media Group, "Zero trust and the role of internet isolation," Information Security Media Group, Boston, UK, 2013, <https://www.bankinfosecurity.com/zero-trust-role-internet-isolation-a-15652>.
- [12] I. Brosso, A. L. Neve, G. Bressan, and W. V. Ruggiero, "A continuous authentication system based on user behavior analysis," in *Proceedings of the 2010 International Conference on Availability, Reliability and Security*, Krakow, Poland, February 2010.
- [13] Joint Task Force, *Special Publication 800-37 Rev. 2: Risk Management Framework for Information Systems and Organizations: A System Life Cycle Approach for Security and Privacy*, National Institute of Standards and Technology, Gaithersburg, MD, USA, 2018, <https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-37r2.pdf>.

- [14] Cybersecurity & Infrastructure Security Agency, “Identity, credential, and access management (ICAM),” Cybersecurity & Infrastructure Security Agency, Arlington, VA, USA, 2021, <https://www.cisa.gov/safecom/icam-resources>.
- [15] P. A. Grassi, M. E. Garcia, and J. L. Fenton, *Special Publication 800-63-3: Digital Identity Guidelines*, National Institute of Standards and Technology, Gaithersburg, MD, USA, 2017, <https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-63-3.pdf>.
- [16] U.S. Department of Health & Human Services, *Health Information Privacy*, U.S. Department of Health & Human Services, Washington, DC, USA, 2021, <https://www.hhs.gov/hipaa/index.html>.
- [17] L. Goodspeed, *Request for Comments: PCI DSS Version 4.0 Draft Standard*, PCI Security Standards Council, Wakefield, MA, USA, 2020, <https://blog.pcisecuritystandards.org/request-for-comments-pci-dss-version-4.0-draft-standard>.
- [18] The European Parliament and The Council of the European Union, *Regulation (EU) 2016/679 OF The European Parliament and of The Council of 27 April 2016 on the protection of Natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation)*, European Union, Brussels, Belgium, 2016, <https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016R0679>.
- [19] J. Opara-Martins, R. Sahandi, and F. Tian, “Critical analysis of vendor lock-in and its impact on cloud computing migration: a business perspective,” *Journal of Cloud Computing*, vol. 5, no. 4, 2016.
- [20] OASIS Cyber Threat Intelligence (CTI), *STIX™ version 2.1*, OASIS, Burlington, MA, USA, 2021, <https://www.oasis-open.org/standard/6426/>.
- [21] OASIS, *TAXII™ Version 2.1*, OASIS, Burlington, MA, USA, 2021, <https://docs.oasis-open.org/cti/taxii/v2.1/cs01/taxii-v2.1-cs01.pdf>.
- [22] R. Ward and B. Beyer, “BeyondCorp: a new approach to enterprise security,” *Usenix*, vol. 39, no. 6, pp. 6–11, 2014, <https://www.usenix.org/publications/login/dec14/ward>.
- [23] B. Osborn, J. McWilliams, B. Beyer, and M. Saltonstall, “BeyondCorp: design to deployment at Google,” *Usenix*, vol. 41, pp. 28–34, 2016, <https://www.usenix.org/publications/login/spring2016/osborn>.
- [24] L. Cittadini, B. Spear, B. Beyer, and M. Saltonstall, “BeyondCorp: the access proxy,” *Usenix*, vol. 41, no. 4, pp. 28–33, 2016, <https://www.usenix.org/publications/login/winter2016/cittadini>.
- [25] J. Peck, B. Beyer, C. Beske, and M. Saltonstall, “Migrating to BeyondCorp: maintaining productivity while improving security,” *Usenix*, vol. 42, no. 2, pp. 49–55, 2017, <https://www.usenix.org/publications/login/summer2017/peck>.
- [26] V. Escobedo, B. Beyer, M. Saltonstall, and F. Żyżniewski, “BeyondCorp: the user experience,” *Usenix*, vol. 42, no. 3, pp. 38–43, 2017, <https://www.usenix.org/publications/login/fall2017/escobedo>.
- [27] H. King, M. Janosko, B. Beyer, and M. Saltonstall, “BeyondCorp 6: building a healthy fleet,” *Usenix*, vol. 43, no. 3, pp. 24–30, 2018, https://www.usenix.org/system/files/login/articles/login_fall18_05_king.pdf.

Research Article

Efficient Ciphertext-Policy Attribute-Based Encryption Constructions with Outsourced Encryption and Decryption

Hassan El Gafif  and **Ahmed Toumanari** 

Laboratory of Applied Mathematics and Intelligent Systems Engineering (MAISI), National School of Applied Sciences (ENSA), Agadir 80999, Morocco

Correspondence should be addressed to Hassan El Gafif; hassan.elgafif@edu.uiz.ac.ma

Received 5 September 2020; Revised 14 November 2020; Accepted 27 April 2021; Published 18 May 2021

Academic Editor: Chalee Vorakulpipat

Copyright © 2021 Hassan El Gafif and Ahmed Toumanari. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The invention of the Ciphertext-Policy Attribute-Based Encryption scheme opened a new perspective for realizing attribute-based access control systems without being forced to trust the storage service provider, which is the case in traditional systems where data are sent to the storage service provider in clear and the storage service provider is the party that controls the access to these data. In the Ciphertext-Policy Attribute-Based Encryption model, the data owner encrypts data using an attribute-based access structure before sending them to the storage service, and only users with authorized sets of attributes can successfully decrypt the generated ciphertext. However, Ciphertext-Policy Attribute-Based Encryption schemes employ expensive operations (i.e., bilinear pairings and modular exponentiations) and generate long ciphertexts and secret keys, which makes them hard to implement in real-life applications especially for resource-constrained devices. In this paper, we propose two Ciphertext-Policy Attribute-Based Encryption Key Encapsulation Mechanisms that can be provided as services in the cloud, minimizing the user's encryption and decryption costs without exposing any sensitive information to the public cloud provider. In the first scheme, the ABE Service Provider is considered fully untrusted. On the other hand, the second scheme requires the ABE Service Provider to be semi-trusted (Honest-but-Curious) and does not collude with illegitimate users. Both schemes are proved to be selectively CPA-secure in the random oracle. The theoretical and experimental performance results show that both our first and second schemes are more efficient than the reviewed outsourced CP-ABE schemes in terms of user-side computation, communication, and storage costs.

1. Introduction

In the past, businesses were suffering from the overheads of dealing with their IT infrastructure installation and management. Nowadays, they can easily minimize these costs by externalizing their activities to one of the existing cloud solutions and paying only the amount of resources they consumed. This new paradigm is beneficial for both users and cloud providers, and this is what makes cloud services continue to attract more enterprises and individual users, helping them to start or improve their businesses easily. Cloud Storage is one of the services offered by cloud providers to help companies and individuals store, manage, and share data efficiently. Nevertheless, when outsourcing data,

data owners are also outsourcing the control over their data. Therefore, this creates data security and confidentiality challenges against a third party who comprised the cloud server to steal data or even against a curious cloud provider [1]. Hence, data owners should encrypt data before outsourcing them to make sure that only authorized users can decrypt and gain access to the data.

Cryptosystems in traditional public cryptography are one-to-one ciphers, meaning that the data owner should retrieve the public key of all the authorized users and encrypt a copy of his data for each user with the corresponding public key. For example, if a data owner wants to share a document with 100 users, he must create 100 copies of the document, retrieve 100 public keys, and encrypt each copy

with the public key of the corresponding user. Thus, this solution is not practical since it produces huge computation, storage, and communication overheads.

In 2005, Sahai and Waters [2] proposed a Fuzzy Identity-Based Encryption (FIBE) scheme with a new model that is not based on users' public keys or identities (as in Identity-Based Encryption schemes), but instead, their model is using attributes to encrypt data and to generate secret keys. In this model, a Trusted Authority (TA) generates users' secret keys based on their sets of attributes, and data owners specify a set of attributes and a threshold (which is the minimum number of attributes in the encryption set of attributes that should exist in the user's set of attributes) and encrypt data using this set of attributes and threshold. Only users with a number of attributes existing in the encryption set of attributes that is greater than the threshold will be able to decrypt the ciphertext using their secret keys.

Later on, two main variants of FIBE were proposed. Goyal et al. proposed the Key-Policy Attribute-Based Encryption (KP-ABE) scheme [3] where the data owner encrypts data with a set of attributes and users' secret keys are generated based on an access policy that is associated with them. Bethencourt et al. presented the Ciphertext-Policy Attribute-Based Encryption (CP-ABE) scheme [4] where the data owner encrypts data with an access policy and users' secret keys are generated based on their sets of attributes.

The computation overhead in ABE schemes is the most challenging part that makes them hard to be adopted in real-life applications. This is due to the number of expensive modular exponentiations and pairing operations that increases linearly with the size of the access policy.

Many contributions were proposed to optimize this computation overhead. Some of these contributions used different techniques to minimize the number of these operations in the encryption and decryption phases [5, 6] or to split them into two phases: in the first phase, most of the expensive operations are performed offline before knowing the message, and the second phase rapidly assembles the ciphertext [7, 8]. Others replaced the expensive modular exponentiations and pairing operations with the lightweight elliptic curve additions and point-scalar multiplications [9–11]. However, these solutions are still hard to be implemented in the applications where devices are resource-constrained such as the Internet of Things (IoT) and Wireless Sensor Networks (WSN). Computation outsourcing is another direction that achieved better results. In this solution, a big part of the encryption and decryption computation is outsourced to the cloud without revealing any sensitive information to the cloud providers that can help them reveal the plaintexts.

1.1. Our Contribution. Based on [4], we propose two CP-ABE with Outsourced Encryption and Decryption (CP-ABE-OED) Key Encapsulation Mechanisms (KEM) where the public ABE Service Provider performs all the encryption and decryption expensive operations leaving only one modular exponentiation and simple multiplications to be executed by the user when encrypting or decrypting data.

The first scheme is suitable for the applications that consider the cloud service provider untrusted. On the other hand, the second scheme requires that the ABE Service Provider is a semi-trusted party that cannot collude with unauthorized users. Both schemes achieve provable CPA-security selectively in the random oracle.

1.2. Organization. The rest of our paper will be organized as follows. In Section 2, we will discuss the related work. Next, we define the preliminaries in Section 3. Later on, we present our 1st and 2nd CP-ABE-OED KEMs and their security analysis in Section 4. Section 5 is dedicated to showing and analysing the performance results. Finally, we conclude our paper in Section 6.

2. Related Work

In 2011, Green et al. [12] proposed the first outsourced CP-ABE scheme, which is selectively CPA-secure. They outsourced a big part of the expensive decryption operations in Waters's large universe construction [13] to a decryption proxy (e.g., Cloud Server), leaving only one modular exponentiation to be executed by the user. In the registration phase, the Trusted Authority (TA) generates a public transformation key TK and a secret decryption key z for each user. To decrypt a ciphertext CT, the user sends his TK to the decryption proxy which transforms CT to a short ElGamal-style [14] ciphertext. Then, using the decryption key z , the user decrypts the transformed ciphertext. To decrypt an ABE ciphertext containing 100 attributes, it takes nearly 30 seconds of sustained computation on a 412 MHz ARM-based iPhone 3G with 128 MB of RAM using the original CP-ABE scheme [13], while it requires only 60 milliseconds using Green et al.'s scheme [12]. Besides, thousands of lines of code, dedicated to determining how a key satisfies the access policy, were removed from the user's side. For instance, in libfenc [15], about 3000 lines are dedicated to access policy handling, excluding dependencies. An improved scheme is also provided in [12] that is selectively secure in the Replayable Chosen-Ciphertext Attack (RCCA) security model using Fujisaki and Okamoto techniques [16].

Afterward, many contributions added the notion of verifiability (i.e., the ability to verify the correctness of the transformation performed by a proxy) to the mechanism of decryption outsourcing [17–21]. In 2016, Mao et al. [21] proposed a generic construction that transforms any (selectively) CPA-secure ABE scheme with outsourced decryption (e.g., Green et al. [12]) into a (selectively) CPA-secure ABE scheme with verifiable outsourced decryption. In contrast with [17] that separately encrypts an extra random message (which is used to commit to the true message), [21] is encrypting the true message and a random message together. It then commits the random value to the message using a commitment scheme that satisfies the hiding and binding properties (at least computationally). In the decryption phase, the user receives the partially decrypted ciphertext from the decryption proxy and the

commitment from the storage server and runs the revealing algorithm of the commitment scheme to verify the correctness of the transformation. The authors showed that the instantiation of this construction in the standard model using Green et al.'s small-universe, backward-compatible, and selectively CPA-secure CP-ABE scheme with outsourced decryption [12] and Pedersen Commitment [22] as the underlying commitment scheme is more efficient than Lai et al.'s scheme [17]. They also proposed a second generic construction to transform any (selectively) CPA-secure ABE scheme with outsourced decryption, that has ciphertext verifiability (i.e., the possibility to verify whether a normal ciphertext will be recovered into the same plaintext under two different decryption keys with two specific attributes) or delegatability (i.e., the capability to use a key to derive another inferior key), into a (selectively) RCCA-secure ABE scheme with verifiable outsourced decryption. They claimed that this is the first RCCA-secure construction that does not rely on a random oracle. In this construction, they combined a secure encapsulation scheme, a strong one-time message authentication code, and a secure commitment scheme.

Obviously, the previous schemes are not suitable for IoT applications where lightweight devices encrypt data and not only decrypt them (e.g., Wireless Sensor Networks) because the encryption cost produced in these schemes is still high. Accordingly, outsourcing the encryption operations in addition to the decryption operations became a new direction [23–27].

Based on [4], Zhou et al. proposed a CP-ABE scheme with outsourced encryption and decryption [23]. They outsourced a big part of the encryption operations by subdividing the access policy T into two parts: T_{DO} (data owner's access tree) and T_{ESP} (Encryption Service Provider's access tree) such that $T = T_{ESP} \text{ AND } T_{DO}$. The data owner generates a random number $s \in Z_p$ and a random 1-degree polynomial $q_R(x)$, where $q_R(0) = s$, $q_R(1) = s_1$ and $q_R(2) = s_2$ and computes $C = M \cdot e(g, g)^{\alpha \cdot s}$ and $C' = g^{\beta \cdot s}$. Then, he generates the ciphertext components C_y and C'_y for his subtree T_{DO} in the same way as CP-ABE [4] using s_2 as the shared key and sends $\{C, C, \{C_y, C'_y\}_{y \in Y_{DO}}, T_{DO}, T_{ESP}, s_1\}$ to the Encryption Service Provider (ESP). Similarly, ESP computes the ciphertext components C_y and C'_y for T_{ESP} using s_1 as the shared key. The final ciphertext is $CT = \{T = T_{ESP} \wedge T_{DO}, C, C, \{C_y, C'_y\}_{y \in Y_{ESP} \cup Y_{DO}}\}$. The decryption outsourcing is achieved using almost the same key-blinding technique of Green et al. [12]. However, an untrusted ESP can reveal the encrypted data by colluding with unauthorized users with sets of attributes that satisfy T_{DO} . Therefore, this solution is suitable only for applications where the ESP is at least semi-trusted.

In 2014, Asim et al. [25] proposed a new CP-ABE scheme where they outsourced a part of the encryption operations to a semi-trusted proxy A and they outsourced the decryption phase to a semi-trusted proxy B following the same technique employed in [12]. Using an encryption secret key generated by the Trusted Authority, the proxy A computes g^s and uses it as the access policy root's secret to generate the

access policy's leaf nodes' components \widehat{g}^{s_j} . Afterward, it multiplies each leaf node's component with the corresponding attribute component $\widetilde{C}_j = H_1(a_j)^{-s}$ in the partially encrypted ciphertext received from the host. The authors claim that their construction is secure in the generic group model under the assumption that proxy A and proxy B will not collude with unauthorized users and will not collude with each other. However, unauthorized users with at least one attribute (a_x) that exists in the access policy can reveal the plaintext using the partially encrypted ciphertext \widetilde{CT} and the ciphertext generated by proxy A (CT). For each leaf node j of the access policy, the attacker retrieves $\widetilde{C}_j = H_1(a_j)^{-s}$ from \widetilde{CT} and $C_j = g^{s_j} \cdot H_1(a_j)^{-s}$ from CT and computes $g^{s_j} = C_j / \widetilde{C}_j$. Then, he executes PolicyGeneration function backward to retrieve g^s and computes $A^s = (e(C, D^1) / e(g^s \cdot H_1(a_x)^{-s}, D^2) \cdot e(C, D_x^3))^z$. Finally, the attacker reveals the plaintext $M = C \oplus H_2(A^s)$. In addition, their scheme is not correct (i.e., given an SK of a set of attributes S that satisfies the access policy τ , $\text{Dec}(\text{Enc}(M, \tau), \text{SK}) \neq M$). In the PolicyGeneration phase, they used g^s (instead of \widehat{s}) as the shared key to get the shares g^{s_j} . However, in the decryption phase they used the polynomial interpolation on \widehat{s}_j , which will result in a value that is different than g^s and, as a result, the decryption output will be different than M .

Subsequently, Zhang et al. [26] presented a fully outsourced CP-ABE scheme that, for the first time, achieves outsourced key generation, encryption, and decryption simultaneously. In their system, two Key Generation Service Providers (KGSP1, KGSP2) help TA to generate Intermediate Secret Keys (ISKs), and two Encryption Service Providers (ESP1, ESP2) help users to generate Intermediate Ciphertexts (ITs). Decryption outsourcing is achieved using the same key blinding used in Green et al.'s scheme [12]. The extra communication costs that had arisen from outsourced key generation and encryption are offline, meaning that TA and users can communicate with the cloud servers in their spare time. The system is proved to be secure under the assumption that two KGSPs (ESPs) do not collude with each other, so the final combined ISK (IT) should be information-theoretically hidden from two servers. It is selectively CPA-secure against corrupt users colluding with KGSP1, ESP1, and SSP and corrupt users colluding with KGSP2, ESP2, and SSP who can obtain the conversion key at Decryption Service Provider.

Other contributions proposed outsourced CP-ABE schemes using trusted parties such as fog nodes [28] or a trusted private cloud provider [29].

3. Preliminaries

3.1. Bilinear Maps [4]. Let G and G_T be two multiplicative cyclic groups of prime order p . Let g be a generator of G and e be a bilinear map, $e : G \times G \rightarrow G_T$, that has the following properties:

- (i) Bilinearity: for all $u, v \in G$ and $a, b \in Z_p$, we have $e(u^a, v^b) = e(u, v)^{a \cdot b}$
- (ii) Non-degeneracy: $e(g, g) \neq 1$

We say G is a bilinear group if the group operation in G and the bilinear map $e : G \times G \rightarrow G_T$ are both efficiently computable.

3.2. Access Structure [30]. Let $\{P_1, P_2, \dots, P_n\}$ be a set of parties. A collection $\mathbb{A} \subseteq 2^{\{P_1, P_2, \dots, P_n\}}$ is monotone if $\forall B, C : \text{if } B \in \mathbb{A} \text{ and } B \subseteq C \text{ then } C \in \mathbb{A}$. An access structure (respectively, monotone access structure) is a collection (respectively, monotone collection) \mathbb{A} of non-empty subsets of $\{P_1, P_2, \dots, P_n\}$; i.e., $\mathbb{A} \subseteq 2^{\{P_1, P_2, \dots, P_n\}} \setminus \{\emptyset\}$. The sets in \mathbb{A} are called the authorized sets, and the sets not in \mathbb{A} are called the unauthorized sets.

In our context, we will use a monotone access structure where the attributes play the role of the parties, which means that the access structure \mathbb{A} will contain the authorized sets of attributes.

3.3. Linear Secret Sharing Scheme (LSSS) [13]. A secret-sharing scheme Π over a set of parties P is called linear (over Z_p) if the following is satisfied:

- (i) The shares for each party form a vector over Z_p .
- (ii) There exists a matrix M with l rows and n columns called the share-generating matrix for Π . For all $i = 1, \dots, l$, the i 'th row of M , we let the function ρ define the party labeling row i as $\rho(i)$. When we consider the column vector $v = (s, r_2, \dots, r_n)$, where $s \in Z_p$ is the secret to be shared, and $r_2, \dots, r_n \in Z_p$ are randomly chosen; then $M \cdot v$ is the vector of l shares of the secret s according to Π . The share $(M \cdot v)_i$ belongs to party $\rho(i)$.

It is shown in [30] that every linear secret sharing-scheme according to the above definition also enjoys the linear reconstruction property, defined as follows: suppose that Π is an LSSS for the access structure A . Let $S \in A$ be any authorized set, and let $I \subset \{1, \dots, l\}$ be defined as $I = \{i : \rho(i) \in S\}$. Then, there exist constants $\{w_i \in Z_p\}_{i \in I}$ such that, if $\{\lambda_i\}$ are valid shares of any secret s according to Π , then $\sum_{i \in I} w_i \cdot \lambda_i = s$.

Furthermore, it is shown in [30] that these constants $\{w_i\}$ can be found in time polynomial in the size of the share-generating matrix M .

We note that we use the convention that vector $(1, 0, 0, \dots, 0)$ is the ‘‘target’’ vector for any linear secret sharing scheme. For any satisfying set of rows I in M , we will have that the target vector is in the span of I . For any unauthorized set of rows I , the target vector is not in the span of the rows of the set I . Moreover, there will exist a column vector w such that $(1, 0, 0, \dots, 0) \cdot w = -1$ and $M_i \cdot w = 0$ for all $i \in I$.

Using standard techniques [30], one can convert any monotonic Boolean formula into an LSSS representation. An access tree of l nodes will result in an LSSS matrix of l rows. We refer the reader to the appendix of [31] for a discussion of how to perform this conversion.

3.4. CPA-Security Game

- (i) *Setup.* The challenger runs the Setup algorithm and gives the public parameters PK to the adversary.
- (ii) *Phase 1.* When the adversary \mathcal{A} queries the decryption key and the transformation key on S , the challenger passes S on to the key generation oracle to get the corresponding decryption key and transformation key and then returns the result to \mathcal{A} .
- (iii) *Challenge.* The adversary \mathcal{A} submits the access structure (M^*, ρ^*) (which is not satisfied by any of the sets of attributes S passed in phase 1) to be challenged on and requests the challenge Key^* . The challenger flips a random coin $b \in \{0, 1\}$. If $b = 0$, it returns CT^* to \mathcal{A} , where the first element in CT^* is a random Key^* . If $b = 1$, it returns CT^* to \mathcal{A} , where the first element in CT^* is a well-constructed Key^* .
- (iv) *Phase 2.* Phase 1 is repeated with the restriction that the adversary cannot obtain a decryption key for a set of attributes that satisfies (M^*, ρ^*) .
- (v) *Guess.* The adversary outputs 0 if Key^* is random and 1 if Key^* is a well-constructed key.

4. Our Proposed Constructions

4.1. The 1st Proposed CP-ABE-OED Key Encapsulation Mechanism. In this scheme, the ABE Service Provider is considered to be an untrusted party.

4.1.1. The Construction. (1) *Setup Phase.* In this phase, we execute the function $\text{setup}(\lambda)$ that takes as input a security parameter λ , which determines the size of the groups. $\text{setup}(\lambda)$ chooses a bilinear group G of prime order p with a generator g and a bilinear map $e : G \times G \rightarrow G_T$.

It also defines a hash function $H_1 : \{0, 1\}^* \rightarrow G$ mapping each attribute (described as a binary string) to a random group element, and a hash function $H_2 : \{0, 1\}^* \rightarrow Z_p$.

Afterward, it generates two random numbers $\alpha, \beta \in Z_p$. Then, it secretly stores the master key $\text{MK} = \{g^\alpha, \beta\}$ and publishes the public parameters:

$$\text{PK} = \{G, g, e(\cdot, \cdot), H_1(\cdot), H_2(\cdot), h = g^\beta, e(g, g)^\alpha\}. \quad (1)$$

(2) *Registration and Key Generation Phase.* In Figure 1, Alice and Bob represent two users. Bob plays the role of the data owner and Alice plays the role of the data receiver. In the registration phase, both Alice and Bob behave in the same way.

First, the Trusted Authority (TA) registers Alice and Bob and associates a set of attributes to each of them (S_A for Alice and S_B for Bob) and executes $\text{KeyGen}(U, \text{MK}, S_i)$.

$\text{KeyGen}(U, \text{MK}, S_i)$ is defined as follows:

- (i) First, it generates the encryption key $\text{EK}_i = s_i$ where s_i is picked randomly in Z_p , and the decryption key $\text{DK}_i = z_i$ where z_i is a random number in Z_p .

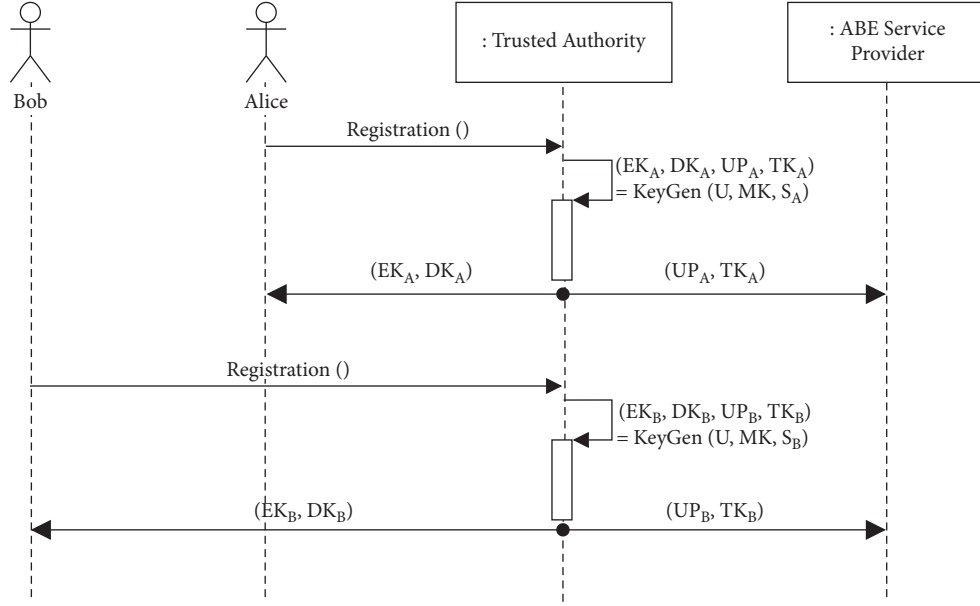


FIGURE 1: Registration and key generation's sequence diagram.

(ii) Afterward, it computes the user's parameters as follows:

$$\text{UP}_i = \left\{ \begin{array}{l} \text{UP}_{i,u,1} = g^{1/H_2(u\|s_i)} \\ \text{UP}_{i,u,2} = H_1(u)^{1/H_2(u\|s_i)} \end{array} \right\}_{\forall u \in U}.$$

(iii) It also computes the Transformation Key (TK_i). First, it chooses a random number $r_i \in Z_p$ and for each $j \in S_i$ it picks $r_{i,j} \in Z_p$ randomly. Then, it computes

$$\text{TK}_i = \left\{ \begin{array}{l} S_i \\ D_i = g^{(\alpha+r_i)/\beta \cdot z_i} \\ \forall j \in S_i: \left\{ \begin{array}{l} D_{i,j} = g^{r_{i,j}/z_i} \cdot H_1(j)^{r_{i,j}/z_i} \\ D'_{i,j} = g^{r_{i,j}/z_i} \end{array} \right\} \end{array} \right\}.$$

(iv) Finally, it outputs $(\text{EK}_i, \text{DK}_i, \text{UP}_i, \text{TK}_i)$.

TA sends $(\text{EK}_i, \text{DK}_i)$ securely to the user i and sends $(\text{UP}_i, \text{TK}_i)$ publicly to the ABE Service Provider.

(3) *Encryption Phase.* As shown in Figure 2, the encryption phase consists of two steps. In the first step, Bob uses its encryption key EK_B and an $l \times n$ LSSS access structure (M, ρ) and calls the function $\text{Encrypt}(\text{PK}, \text{EK}_B, (M, \rho))$.

$\text{Encrypt}(\text{PK}, \text{EK}_B, (M, \rho))$ is defined as follows:

(i) First, it generates a random column vector $v = (s, y_2, y_3, \dots, y_n) \in Z_p^n$ to share the encryption exponent s .

(ii) For each $i \in \{1, 2, \dots, l\}$, it computes $\lambda_i = M_i \cdot v$ where M_i is the i th row of M .

(iii) Then, it generates

$$\text{preCT} = \left\{ \begin{array}{l} (M, \rho) \\ C = h^s \\ \forall i \in \{1, 2, \dots, l\}: C_i^{\text{pre}} = H_2(\rho(i) \parallel s_B) \cdot \lambda_i \end{array} \right\}. \quad (2)$$

(iv) Finally, it outputs preCT .

Afterward, Bob sends preCT to the ABE Service Provider.

In the second step, the ABE Service Provider executes the function $\text{OutEncrypt}(\text{PK}, \text{UP}_B, \text{preCT})$ after receiving preCT .

$\text{OutEncrypt}(\text{PK}, \text{UP}_B, \text{preCT})$ performs the following instructions:

(i) For each $i \in \{1, 2, \dots, l\}$, it computes

$$\left\{ \begin{array}{l} C_i = \text{UP}_{B,\rho(i),1}^{C_i^{\text{pre}}} = \left(g^{1/H_2(\rho(i)\|s_B)} \right)^{H_2(\rho(i)\|s_B) \cdot \lambda_i} = g^{\lambda_i}, \\ C'_i = \text{UP}_{B,\rho(i),2}^{C_i^{\text{pre}}} = \left(H_1(\rho(i))^{1/H_2(\rho(i)\|s_B)} \right)^{H_2(\rho(i)\|s_B) \cdot \lambda_i} = H_1(\rho(i))^{\lambda_i}. \end{array} \right. \quad (3)$$

(ii) Then, it outputs $\text{CT} = \{(M, \rho), C, \{C_i, C'_i\}_{i \in \{1, 2, \dots, l\}}\}$.

Finally, the ABE Service Provider sends CT to the Cloud Storage Provider (CSP).

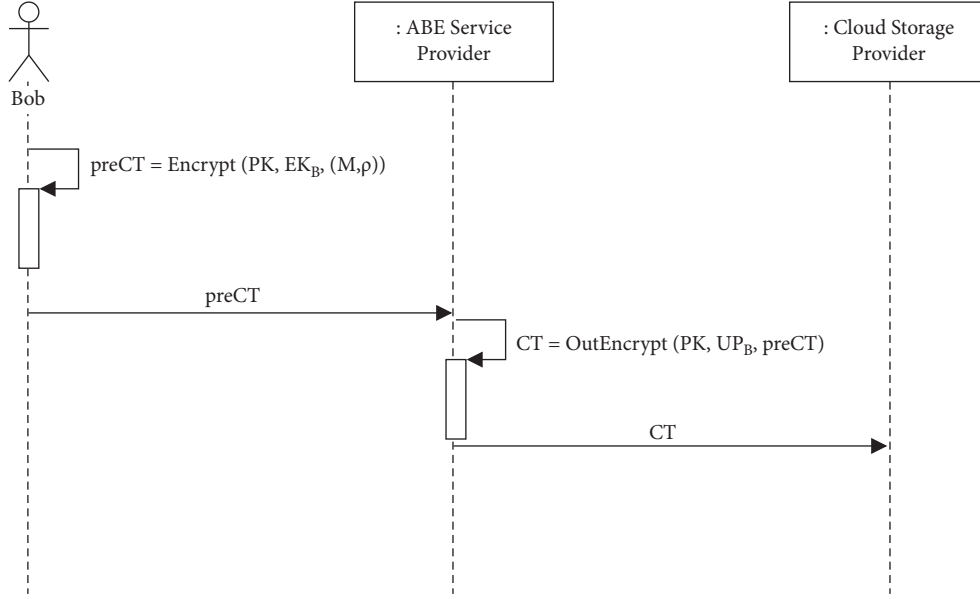


FIGURE 2: Encryption phase's sequence diagram.

(4) *Decryption Phase.* First, as shown in Figure 3, Alice requests the transformed ciphertext transCT from the ABE Service Provider. The ABE Service Provider receives CT from the CSP and executes the function $\text{OutDecrypt}(\text{PK}, \text{CT}, \text{TK}_A)$.

$\text{OutDecrypt}(\text{PK}, \text{CT}, \text{TK}_A)$ is defined as follows:

- (i) If Alice's set of attributes S_A does not satisfy the access structure, then it outputs \perp . Otherwise, let $I = \{i : \rho(i) \in S_A\}$ and $\{w_i \in Z_p\}_{i \in I}$ such that $\sum_{i \in I} w_i \cdot M_i = (1, 0, 0, \dots, 0)$.
- (ii) Then, it computes

$$\begin{aligned}
 A &= \prod_{i \in I} \left(\frac{e(D_{A,i}, C_i)}{e(D'_{A,i}, C'_i)} \right)^{w_i} = \prod_{i \in I} \left(\frac{e(g^{r_A/z_A} \cdot H_1(\rho(i))^{r_{A,i}/z_A}, g^{\lambda_i})}{e(g^{r_{A,i}/z_A}, H_1(\rho(i))^{\lambda_i})} \right)^{w_i} = \prod_{i \in I} \left(\frac{e(g^{r_A} \cdot H_1(\rho(i))^{r_{A,i}}, g)}{e(g^{r_{A,i}}, H_1(\rho(i)))} \right)^{w_i \cdot \lambda_i / z_A} \\
 &= \prod_{i \in I} e(g, g)^{r_A \cdot w_i \cdot \lambda_i / z_A} = e(g, g)^{r_A / z_A \cdot \sum_{i \in I} w_i \cdot \lambda_i} = e(g, g)^{r_A \cdot s / z_A}.
 \end{aligned} \tag{4}$$

(iii) Finally, it outputs transCT generated as follows:

$$\text{transCT} = \frac{e(C, D_A)}{A} = \frac{e(g^{\beta \cdot s}, g^{(\alpha + r_A) / \beta \cdot z_A})}{e(g, g)^{r_A \cdot s / z_A}} = \left(\frac{e(g, g)^\alpha \cdot e(g, g)^{r_A}}{e(g, g)^{r_A}} \right)^{s / z_A} = e(g, g)^{\alpha \cdot s / z_A}. \tag{5}$$

The ABE Service Provider sends transCT to Alice.

After receiving transCT , Alice decrypts it using its decryption key DK_A by calling the function $\text{Decrypt}(\text{PK}, \text{transCT}, \text{DK}_A)$.

$\text{Decrypt}(\text{PK}, \text{transCT}, \text{DK}_A)$ executes the following instructions:

- (i) It computes

$$\text{Key} = \text{transCT}^{\text{DK}_A} = \left(e(g, g)^{\alpha \cdot s / z_A} \right)^{z_A} = e(g, g)^{\alpha \cdot s}. \tag{6}$$

- (ii) Then, it outputs the Key.

4.1.2. Security Analysis. Before starting our security proof, we will create a modified version of Bethencourt et al.'s CP-

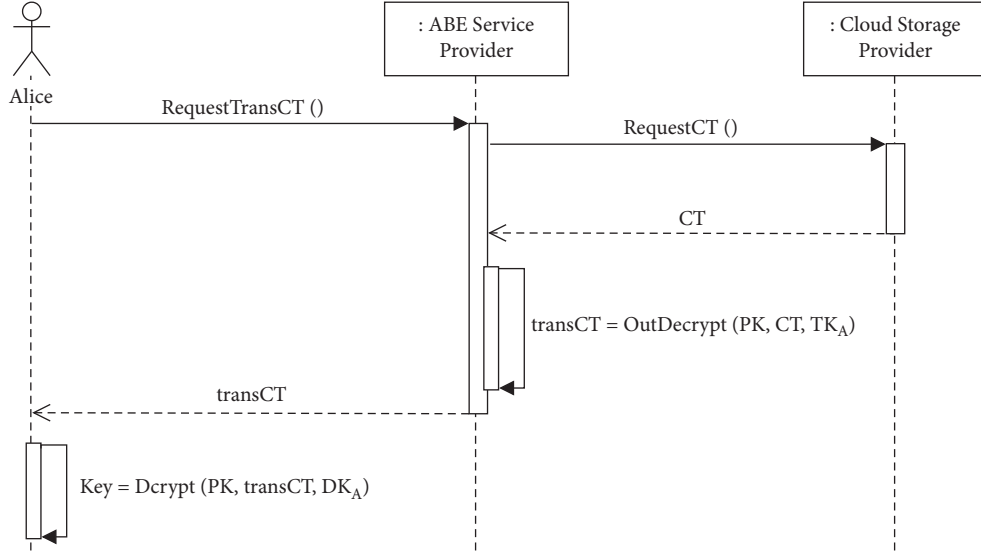


FIGURE 3: Decryption phase's sequence diagram.

ABE scheme [4] and prove that it achieves the same security level of the original scheme in the random oracle. Then, we will prove that our 1st CP-ABE-OED KEM is selectively CPA-secure in the random oracle if the modified version of Bethencourt et al.'s scheme is selectively CPA-secure in the random oracle. Thus, our 1st CP-ABE-OED KEM is selectively CPA-secure in the random oracle if Bethencourt et al.'s scheme is selectively CPA-secure in the random oracle.

We create the modified Bethencourt et al.'s scheme by modifying the encryption phase as follows:

- (i) Instead of using Shamir's Secret Sharing Scheme to build the access policy, we use the LSSS access structure (M, ρ) in the same way as in our scheme and generate the shares λ_i .
- (ii) We choose a random number $s_B \in Z_p$ and compute for each row i of M : $\mathcal{R}_i = H_2(\rho(i)||s_B) \cdot \lambda_i$ (the hash function $H_2: \{0, 1\}^* \rightarrow Z_p$ must be defined in the public parameters).
- (iii) The generated ciphertext will be defined as follows:

$$\left\{ \begin{array}{l} (M, \rho), \\ \tilde{C} = M \cdot e(g, g)^{\alpha \cdot s} \\ h^s, \\ \forall i = 1, \dots, l: \left\{ \begin{array}{l} \mathcal{R}_i = H_2(\rho(i)||s_B) \cdot \lambda_i \\ C_i = g^{\lambda_i}, \\ C'_i = H_1(\rho(i))^{\lambda_i}. \end{array} \right. \end{array} \right. \quad (7)$$

It is obvious that the modified Bethencourt et al.'s scheme achieves the same security level as the original scheme in the random oracle. That is because if we consider $H_2(\rho(i)||s_B)$ random, then \mathcal{R}_i is random and the attacker cannot compute λ_i from \mathcal{R}_i without knowing $H_2(\rho(i)||s_B)$.

Therefore, an attacker cannot distinguish between the distributions $(H_2(\rho(i)||s_B) \cdot \lambda_i, g^{\lambda_i}, H_1(\rho(i))^{\lambda_i})$ and $(r, g^{\lambda_i}, H_1(\rho(i))^{\lambda_i})$, where $r \in Z_p$ is a random number.

Now, we prove the following theorem:

Theorem 1. *Our 1st CP-ABE-OED KEM is selectively CPA-secure in the random oracle if the modified Bethencourt et al. scheme is selectively CPA-secure in the random oracle.*

Suppose that we have an adversary \mathcal{A} with non-negligible advantage ϵ in the selective CPA-security game against our construction. We show how to build a simulator \mathcal{B} that can attack the modified Bethencourt et al. scheme in the selective CPA-security model with advantage ϵ .

(1) *Init.* The adversary gives the challenge access structure (M^*, ρ^*) to the simulator \mathcal{B} . \mathcal{B} sends (M^*, ρ^*) to the challenger.

(2) *Setup.* The simulator \mathcal{B} obtains the public parameters from the challenger:

$$PK' = \left\{ G, g, e(\cdot, \cdot), H_1(\cdot), H_2(\cdot), h = g^{\beta'}, f = g^{1/\beta'}, e(g, g)^{\alpha'} \right\}. \quad (8)$$

The random oracles $H_1(\cdot)$ and $H_2(\cdot)$ are programmed by the challenger.

Then, \mathcal{B} sends the public parameters

$$PK = \left\{ G, g, e(\cdot, \cdot), H_1(\cdot), H_2(\cdot), h = g^{\beta'}, e(g, g)^{\alpha'} \right\}, \quad (9)$$

to the adversary \mathcal{A}

(3) *Phase I.* The adversary sends request queries of sets of attributes S that do not satisfy the challenge access structure (M^*, ρ^*) to \mathcal{B} . The simulator \mathcal{B} calls the challenger's key generation oracle on S to obtain the key

$$SK' = \left\{ \begin{array}{c} S \\ D' = g^{(\alpha'+r)/\beta'} \\ \forall j \in S_i: \left\{ \begin{array}{c} D'_j = g^r \cdot H_1(j)^{r_j} \\ D'_j = g^{r_j} \end{array} \right\} \end{array} \right\}. \quad (10)$$

The simulator chooses a random value $z \in Z_p$ and sets the decryption key as $DK = z$ and the transformation key as

$$TK = \left\{ \begin{array}{c} S \\ D = D'^{1/z} \\ \forall j \in S_i: \left\{ \begin{array}{c} D_j = D_j^{1/z} \\ D_j = D_j^{1/z} \end{array} \right\} \end{array} \right\}. \quad (11)$$

(4) *Challenge*. The simulator sends two distinct random messages m_0 and m_1 to the challenger. The challenger flips a coin $\pi \in \{0, 1\}$ and creates

$$CT' = \left\{ \begin{array}{c} (M^*, \rho^*), \\ C' = m_{\pi} \cdot e(g, g)^{\alpha' s'}, \\ C' = h^{s'}, \\ \forall i = 1, \dots, l: \left\{ \begin{array}{c} \mathcal{R}_i = H_2(\rho(i) \| s_B) \cdot \lambda'_i, \\ C'_i = g^{\lambda'_i}, \\ C'_i = H_1(\rho(i))^{\lambda'_i}. \end{array} \right\} \end{array} \right\}. \quad (12)$$

Then, the challenger sends CT' to the simulator. Later on, the simulator computes

$$UP^* = \left\{ \begin{array}{c} \left\{ \begin{array}{c} UP_{u,1} = (C'_i)^{1/\mathcal{R}_i} \\ UP_{u,2} = (C'_i)^{1/\mathcal{R}_i} \end{array} \right\}_{\forall u \in U \cap Y} \\ \left\{ \begin{array}{c} UP_{u,1} = g^{1/t_u} \\ UP_{u,2} = x_u^{1/t_u} \end{array} \right\}_{\forall u \in U \sim Y} \end{array} \right\}. \quad (13)$$

where $t_u \in Z_p$ and $x_u \in G$ are random numbers.

Then, \mathcal{B} creates CT^* as follows:

$$CT^* = \left\{ \begin{array}{c} (M^*, \rho^*), \\ C = C', \\ C_i^{\text{pre}} = \mathcal{R}_i, \end{array} \right\}. \quad (14)$$

Finally, the simulator flips a coin $b \in \{0, 1\}$ and computes $Key^* = C/m_b$ and then sends CT^* , UP^* , and Key^* to the adversary.

(5) *Phase 2*. The simulator continues to answer queries as in Phase 1.

(6) *Guess*. The adversary will eventually output a guess b' of b . The adversary outputs 0 to guess that Key^* is random, and outputs 1 to guess that $Key^* = e(g, g)^{\alpha' s'}$. The simulator outputs b if $b' = 1$; otherwise it outputs \bar{b} . Thus, if the adversary wins the selective CPA-security game with a non-negligible advantage, then \mathcal{B} can break the security of the modified Bethencourt et al.'s scheme with the same advantage.

4.2. The 2nd Proposed CP-ABE-OED Key Encapsulation Mechanism

4.2.1. *The Construction*. In this scheme, we consider the ABE Service Provider semi-trusted, which means that it cannot collude with illegitimate users to reveal the plaintext.

We will only describe the modified methods that are different from the previous scheme.

(1) *KeyGen* (U, MK, S_i)

(i) First, it generates the encryption key $EK_i = s_i$ where s_i is picked randomly in Z_p , and the decryption key $DK_i = z_i$ where z_i is a random number in Z_p .

(ii) Afterward, it computes the user's parameters as follows: $UP_i = \left\{ \begin{array}{c} UP_{i,1} = g^{1/s_i}, \\ \forall u \in U: UP_{i,u,2} = H_1(u)^{1/s_i}. \end{array} \right\}$

(iii) It also computes the Transformation Key (TK_i). First, it chooses a random number $r_i \in Z_p$ and for each $j \in S_i$ it picks $r_{i,j} \in Z_p$ randomly. Then, it computes $TK_i =$

$$\left\{ \begin{array}{c} S_i \\ D_i = g^{(\alpha+r_i)/\beta \cdot z_i} \\ \forall j \in S_i: \left\{ \begin{array}{c} D_{i,j} = g^{r_{i,j}/z_i} \cdot H_1(j)^{r_{i,j}/z_i} \\ D_{i,j} = g^{r_{i,j}/z_i} \end{array} \right\} \end{array} \right\}.$$

(iv) Finally, it outputs (EK_i, DK_i, UP_i, TK_i) .

(2) *Encrypt* ($PK, EK_B, (M, \rho)$)

(i) It picks a random number $s \in Z_p$ and computes

$$\text{preCT} = \left\{ \begin{array}{c} (M, \rho), \\ C = h^s, \\ C^{\text{pre}} = s_B \cdot s \end{array} \right\}. \quad (15)$$

(ii) Then, it outputs preCT .

(3) *OutEncrypt* (PK, UP_B, preCT)

(i) First, it chooses a random column vector $v = (C^{\text{pre}}, y_2, y_3, \dots, y_n) \in Z_p^n$.

(ii) For each $i \in \{1, 2, \dots, l\}$, it computes $\lambda_i = M_i \cdot v$ where M_i is the i th row of M .

(iii) Then, for each $i \in \{1, 2, \dots, l\}$, it computes

$$\left\{ \begin{array}{c} C_i = UP_{B,1}^{\lambda_i} = g^{\lambda_i/s_B} \\ C'_i = UP_{B,\rho(i),2}^{\lambda_i} = H_1(\rho(i))^{\lambda_i/s_B}, \end{array} \right\}. \quad (16)$$

(iv) Then, it outputs $CT = \{(M, \rho), C, \{C_i, C'_i\}_{i \in \{1, 2, \dots, l\}}\}$.

The decryption phase will perform in the same way as in the previous scheme; however, we will describe it here to show the correctness of our scheme.

(4) *OutDecrypt* (PK, CT, TK_A).

(i) If Alice's set of attributes S_A does not satisfy the access structure, then it outputs \perp . Otherwise,

let $I = \{i : \rho(i) \in S_A\}$ and $\{w_i \in Z_p\}_{i \in I}$ such that $\sum_{i \in I} w_i M_i = (1, 0, 0, \dots, 0)$.

(ii) Then, it computes

$$\begin{aligned} A &= \prod_{i \in I} \left(\frac{e(D_{A,i}, C_i)}{e(D'_{A,i}, C_i)} \right)^{w_i} = \prod_{i \in I} \left(\frac{e(g^{r_A/z_A} H_1(\rho(i))^{r_{A,i}/z_A} g^{\lambda_i/s_B})}{e(g^{r_{A,i}/z_A} H_1(\rho(i))^{\lambda_i/s_B})} \right)^{w_i} = \prod_{i \in I} \left(\frac{e(g^{r_A} H_1(\rho(i))^{r_{A,i}} g)}{e(g^{r_{A,i}} H_1(\rho(i)))} \right)^{w_i \lambda_i / z_A s_B} \\ &= \prod_{i \in I} e(g, g)^{r_A w_i \lambda_i / z_A s_B} = e(g, g)^{r_A / z_A s_B \sum_{i \in I} w_i \lambda_i} = e(g, g)^{r_A s_B s / z_A s_B} = e(g, g)^{r_A s / z_A}. \end{aligned} \quad (17)$$

(iii) Finally, it outputs transCT generated as follows:

$$\text{transCT} = \frac{e(C, D_A)}{A} = \frac{e(g^{\beta s}, g^{(\alpha + r_A)/\beta z_A})}{e(g, g)^{r_A s / z_A}} = \left(\frac{e(g, g)^\alpha \cdot e(g, g)^{r_A}}{e(g, g)^{r_A}} \right)^{s/z_A} = e(g, g)^{\alpha s / z_A}. \quad (18)$$

(5) Decrypt(PK, transCT, DK_A)

(i) It computes

$$\text{Key} = \text{transCT}^{DK_A} = (e(g, g)^{\alpha s / z_A})^{z_A} = e(g, g)^{\alpha s} \quad (19)$$

(ii) Then, it outputs the Key.

$$\text{PK}' = \left\{ G, g, e(\cdot, \cdot), H_1(\cdot), h = g^{\beta'}, f = g^{1/\beta'}, e(g, g)^{\alpha'} \right\}. \quad (20)$$

and forwards them to the adversary \mathcal{A} .

(3) *Phase I.* The adversary sends request queries of sets of attributes S that do not satisfy the challenge access structure (M^*, ρ^*) to \mathcal{B} . The simulator \mathcal{B} calls Bethencourt et al.'s [4] key generation oracle on S to obtain the key

$$\text{SK}' = \left\{ \begin{array}{c} S \\ D' = g^{(\alpha' + r)/\beta'} \\ \forall j \in S_i: \left\{ \begin{array}{l} D'_j = g^r \cdot H_1(j)^{r_j} \\ D_j = g^{r_j} \end{array} \right\} \end{array} \right\}. \quad (21)$$

The simulator chooses a random value $z \in Z_p$ and sets the decryption key as $DK = z$ and the transformation key as

$$\text{TK} = \left\{ \begin{array}{c} S \\ D = D'^{1/z} \\ \forall j \in S_i: \left\{ \begin{array}{l} D_j = D_j'^{1/z} \\ D'_j = D_j'^{1/z} \end{array} \right\} \end{array} \right\}. \quad (22)$$

(4) *Challenge.* The simulator sends two distinct random messages m_0 and m_1 to the challenger. The challenger flips a coin $\pi \in \{0, 1\}$ and creates

$$\text{CT}' = \left\{ \begin{array}{l} (M^*, \rho^*), \\ C' = m_{\pi}, \text{Key}', \\ C' = h^{s'}, \\ \forall i = 1, \dots, l: \left\{ \begin{array}{l} C'_i = g^{\lambda'_i}, \\ C'_i = H_1(\rho(i))^{\lambda'_i} \end{array} \right\} \end{array} \right. \quad (23)$$

4.2.2. *Security Analysis.* In this security proof, we will consider two types of adversaries:

- (i) Type-1 adversary: which refers to illegitimate users trying to break our scheme
- (ii) Type-2 adversary: which refers to a curious ABE cloud provider trying to reveal sensitive information

For the Type-1 adversary, our scheme is viewed as Bethencourt et al.'s scheme [4] with outsourced decryption.

Now, we prove the following theorem:

Theorem 2. *Our 2nd CP-ABE-OED KEM is selectively CPA-secure in the random oracle against Type-1 adversaries if Bethencourt et al.'s scheme [4] is selectively CPA-secure in the random oracle.*

Suppose we have an adversary \mathcal{A} with non-negligible advantage ε in the selective CPA-security game against our construction. We show how to build a simulator \mathcal{B} that can attack Bethencourt et al.'s scheme [4] in the selective CPA-security model with advantage ε .

(1) *Init.* The adversary gives the challenge access structure (M^*, ρ^*) to the simulator \mathcal{B} . \mathcal{B} sends the challenge access structure to the challenger.

(2) *Setup.* The simulator \mathcal{B} obtains Bethencourt et al.'s [4] public parameters

TABLE 1: Notations used in the performance results and analysis section.

Notation	Definition
$ x $	Number of elements in x
l	Number of rows in the LSSS access structure or number of leaf nodes in the access tree
N_l	An upper bound greater than l of all the access policies
S	User's set of attributes
U	Universe of attributes
M_{eG}	Modular exponentiation in G
M_{eG_T}	Modular exponentiation in G_T
M_G	Multiplication in G
M_{G_T}	Multiplication in G_T
M_Z	Multiplication in Z_p
H_G	Hashing in G
H_Z	Hashing in Z_p
G	Element in G
G_T	Element in G_T
Z_p	Element in Z_p

Then, the challenger sends CT^* to the simulator.

Later on, the simulator computes

$$UP^* = \left\{ \left\{ \begin{array}{l} UP_{u,1} = g^{1/t_u} \\ UP_{u,2} = H_1(u)^{1/t_u} \end{array} \right\}_{\forall u \in U} \right\}. \quad (24)$$

where $t_u \in Z_p$ are random numbers.

Later on, the simulator constructs CT^* as follows:

$$CT^* = \left\{ \begin{array}{l} (M^*, \rho^*), \\ C^* = C', \\ \forall i = 1, \dots, l: \begin{cases} C_i^* = C'_i, \\ C_{i'}^* = C_{i'}^b. \end{cases} \end{array} \right. \quad (25)$$

Finally, the simulator flips a coin $b \in \{0, 1\}$ and computes $Key^* = C^*/m_b$, then sends CT^* , UP^* , and Key^* to the adversary.

(5). *Phase 2.* The simulator continues to answer queries as in Phase 1.

(6). *Guess.* The adversary will eventually output a guess b' of b . The adversary outputs 0 to guess that Key^* is random, and outputs 1 to guess that $Key^* = e(g, g)^{\alpha' s'}$. The simulator outputs b if $b' = 1$; otherwise it outputs \bar{b} . Thus, if the adversary wins the selective CPA-security game with a non-negligible advantage, then \mathcal{B} can break the security of Bethencourt et al.'s scheme with the same advantage.

The Type-2 adversary is not allowed to collude with unauthorized users. Thus, he can request only the transformation keys and not the decryption keys from the key generation oracle.

Now, we prove the following theorem:

Theorem 3. *Our 2nd CP-ABE-OED KEM is selectively CPA-secure in the random oracle against Type-2 adversaries if our 2nd CP-ABE-OED KEM is selectively CPA-secure in the random oracle against Type-1 adversaries.*

It is obvious that Type-2 adversary cannot distinguish between two pre-ciphertexts $preCT_0^*$ and $preCT_1^*$ where

$$preCT_b^* = \left\{ \begin{array}{l} (M^*, \rho^*), \\ Key_b^*, \\ C^* = h^s, \\ C^{pre*} = s_B \cdot s \end{array} \right\}. \quad (26)$$

and $Key_b^* = \begin{cases} e(g, g)^{\alpha \cdot s}, & b = 0, \\ e(g, g)^R, & b = 1. \end{cases}$ where $R \in Z_p$ is a random number.

That is because C^{pre*} is random since s_B is random, and the adversary cannot retrieve s without knowing s_B . Thus, Type-2 adversary has no advantage over Type-1 adversary since the only additional information ($C^{pre*} = s_B \cdot s$) he has compared to the Type-1 adversary is not useful. Hence, Theorem 3 is proved.

5. Performance Results and Analysis

5.1. Theoretical Results and Analysis. In this section, we theoretically compare the user's computation, communication, and storage costs between our two schemes and the following schemes:

- (i) The CPA-secure construction of [12].
- (ii) The CPA-secure construction of [21] based on [12] using Pedersen Commitment as a commitment scheme.
- (iii) Zhou et al.'s scheme [23].
- (iv) Zhang et al.'s scheme [26].
- (v) Li et al.'s scheme [27].

We normalized all the schemes based on the following rules:

- (i) The transformation key is created by TA and not the user.
- (ii) We will consider all the schemes as Key Encapsulation Mechanisms (KEMs), meaning that we neglect the part where the message m is encrypted (e.g., $C = m \cdot e(g, g)^{\alpha \cdot s}$) and leave only the parts responsible for sharing the key $e(g, g)^{\alpha \cdot s}$.

TABLE 2: User-side computation cost comparison.

Phase	CP-ABE-OED1	CP-ABE-OED2	[12]	[21]	[23]	[26]	[27]
KeyGen	0	0	0	0	0	0	0
Encrypt	$1.M_{eG} + l.M_Z + l.H_Z$	$1.M_{eG} + 1.M_Z$	$(2 + 3.l).M_{eG} + l.H_G$	$(3 + 3.l).M_{eG} + (2 + l).H_G$	$3.M_{eG} + 1.H_G$	$1.M_{eG} + (1 + 3.N_l).M_G + l.M_Z$	$3.M_{eG} + l.M_G + 1.H_Z$
Decrypt	$1.M_{eG_r}$	$1.M_{eG_r}$	$1.M_{eG_r}$	$3.M_{eG_r}$	$1.M_{eG_r} + 1.M_{G_r}$	$1.M_{eGT}$	$1.M_{eG_r}$

TABLE 3: Running times (in milliseconds) of the main operations using JPBC Library [32] on a Windows 8.1 Core i7 2 GHz PC with 8 Go of RAM.

M_{eG}	M_{eG_T}	M_G	M_{G_T}	M_Z	H_G	H_Z
14.5 ms	0.924 ms	0.073 ms	0.0072 ms	0.0014 ms	31.07 ms	0.118 ms

- (iii) We consider that each user has the ability to encrypt and decrypt.
- (iv) We ignored the access structure \mathbb{A} and the set of attributes S when computing the size of the ciphertexts and the keys since they are common elements between all the schemes.

In Table 1, we define the notations used in this section.

In Table 2, we compare the number of operations executed in each phase (registration phase, encryption phase, and decryption phase) between our proposed schemes and the reviewed schemes.

Obviously, the user is not involved in the computations of the registration phase in all the schemes.

In [12, 21], the user-side encryption cost is very expensive, because the encryption in these schemes is not outsourced. Based on the results in Table 3, which were computed using a Type A curve of the JPBC Library [32] on a Windows 8.1 Core i7 2 GHz PC with 8 GB of RAM, we have the following:

- (i) $M_G = 52M_Z$
- (ii) $M_{eG} = 10357M_Z$
- (iii) $H_G = 22193M_Z$
- (iv) $H_Z = 84M_Z$

We mention that the hashes were computed using the `Element.setFromHash()` method based on SHA-256.

If we convert the encryption costs of the reviewed schemes, we get the following:

- (i) $1.M_{eG} + l.M_Z + l.H_Z = (10357 + 85.l).M_Z$ for our 1st CP-ABE-OED KEM
- (ii) $1.M_{eG} + 1.M_Z = 10358.M_Z$ for our 2nd CP-ABE-OED KEM
- (iii) $3.M_{eG} + 1.H_G = 53264.M_Z$ for [23]
- (iv) $1.M_{eG} + (1 + 3.N_l).M_G + l.M_Z = (10409 + 156.N_l + l).M_Z$ for [26]
- (v) $3.M_{eG} + l.M_G + 1.H_Z = (52.l + 31155).M_Z$ for [27]

We observe that, for access policies smaller than 425 leaf nodes, the user-side encryption in [27] is more efficient than [23].

If $N_l = l$, which is the smallest value N_l can take, the user-side encryption in [26] will be more efficient than [27] (respectively, [23]) for access policies with less than 200 leaf nodes (respectively, 270 leaf nodes). If $N_l = 5.l$, [27] (respectively, [23]) will achieve better efficiency than [26] for all the access policies bigger than 30 leaf nodes (respectively, 50 leaf nodes). Overall, we can say that [26] is more efficient than [23, 27] for small access policies; however, [23, 27] are more efficient for large access policies.

Our 1st CP-ABE-OED KEM achieves a higher user-side encryption efficiency than [26] for all the access policy sizes. It also achieves higher efficiency than [23] for access policies with less than 500 leaf nodes, and higher efficiency than [27] for access policies with less than 630 leaf nodes.

Obviously, our 2nd CP-ABE-OED KEM is more efficient than all the schemes for all the access policy sizes.

The decryption phase costs are almost the same (one modular exponentiation) in all the schemes since they all use the same key blinding technique used in [12]. In [21], the user performs 2 more modular exponentiations to reveal the commitment.

Table 4 shows the communication costs generated in the registration phase, the encryption phase, and the decryption phase between our proposed schemes and the reviewed schemes. In [12, 21, 23, 26], TA sends TK and DK to the user in the registration phase, which costs $(2 + 2.|S|)$ elements in G and $(2 + |S|)$ elements in Z_p for [26], $(1 + 2.|S|)$ elements in G and one element in Z_p for [23], and $(2 + |S|)$ elements in G and one Z_p element for [12, 21]. In [27], TA sends the encryption transformation key ETK and DK to the user, which costs two Z_p elements and $|U|$ elements in G . However, in our proposed schemes, only two elements in Z_p (EK and DK) are communicated between TA and the user. The reason is that TK in our proposed schemes is transferred by TA directly to the ABE Service Provider.

In the encryption phase, the user in [26] receives two Intermediate Ciphertexts (*ITs*) from the ABE Service Provider offline, each of them containing $(1 + 3.N_l)$ elements in G and $(1 + 3.N_l)$ elements in Z_p , and sends the ciphertext CT to CSP, which costs $(1 + 3.l)$ elements in G and $2.l$ elements in Z_p . This makes [26] the most expensive scheme for the users in terms of communication cost produced in the encryption phase. In [12, 21], the user communicates CT to CSP; this costs $(1 + 2.l)$ elements in G for [12] and an additional element in G for [21] generated by the commitment element *cm*. The user in [27] sends the partially encrypted ciphertext *preCT* and the outsourcing parameters to the ABE service provider. This costs $(2 + l)$ elements in G and $(l + 1)$ elements in Z_p , which makes [27] slightly more efficient than [12, 21]. In our 1st CP-ABE-OED KEM, the user sends *preCT* to the ABE Service Provider, which costs him one G element and l elements in Z_p . In [23], the user sends *preCT* that costs only 3 elements in G and one element in Z_p to the ABE Service Provider. In our 2nd CP-ABE-OED KEM, the transfer of *preCT* to the ABE Service Provider costs only one G element and one Z_p element, which makes it the most efficient scheme in terms of user's communication cost in the encryption phase.

In the decryption phase, the user receives the transformed ciphertext *transCT* from the ABE Service Provider in all the schemes, which costs two G_T elements in [23] and

TABLE 4: User-side communication cost comparison.

Phase	KeyGen			Encrypt			Decrypt		
	TA => Bob/Alice	Bob => CSP	Bob Service => Bob	ABE Service => Bob	Bob => ABE Service	CSP => Alice	ABE Service => Alice	Alice => ABE Service	
Link	TA => Bob/Alice	Bob => CSP	Bob Service => Bob	ABE Service => Bob	Bob => ABE Service	CSP => Alice	ABE Service => Alice	Alice => ABE Service	
CP-ABE-OED1	$ EK + DK = 2.Z_p$	0	0	0	$ preCT = 1.G + l.Z_p$	0	$ transCT = 1.G_T$	0	
CP-ABE-OED2	$ EK + DK = 2.Z_p$	0	0	0	$ preCT = 1.G + 1.Z_p$	0	$ transCT = 1.G_T$	0	
[12]	$ TK + DK = (2 + S).G + 1.Z_p$	$ CT = (1 + 2.l).G$	0	0	0	0	$ transCT = 1.G_T$	$ TK = (2 + S).G$	
[21]	$ TK + DK = (2 + S).G + 1.Z_p$	$ CT = (1 + 2.l).G$	0	0	0	1.G	$ transCT = 1.G_T$	$ TK = (2 + S).G$	
[23]	$ TK + DK = (1 + 2. S).G + 1.Z_p$	0	0	0	$ preCT = 3.G + 1.Z_p$	0	$ transCT = 2.G_T$	$ TK = (1 + 2. S).G$	
[26]	$ TK + DK = (2 + 2. S).G + (2 + S).Z_p$	$ CT = (1 + 3.l).G + 2.l.Z_p$	$ IT = 2.((1 + 3.N_1).G + (1 + 3.N_1).Z_p)$	0	0	0	$ transCT = 1.G_T$	$ TK = (2 + 2. S).G + (1 + S).Z_p$	
[27]	$ ETK + DK = U .G + 2.Z_p$	0	0	0	$ preCT + OP = (2 + l).G + (l + 1).Z_p$	0	$ transCT = 1.G_T$	$ TK = (2 + S).G$	

TABLE 5: User-side storage cost comparison.

	[12]	[21]	[23]	[26]	[27]
CP-ABE-OED1	CP-ABE-OED2				
Bob/ Alice	$ EK + DK = 2Z_p$	$ TK + DK = (2 + S)G + 1.Z_p$	$ TK + DK = (1 + 2 S)G + 1.Z_p$	$ TK + DK + IT = (4 + 2 S + 6.N_1)G + (4 + S + 6.N_1).Z_p$	$ ETK + DK = U .G + 2.Z_p$

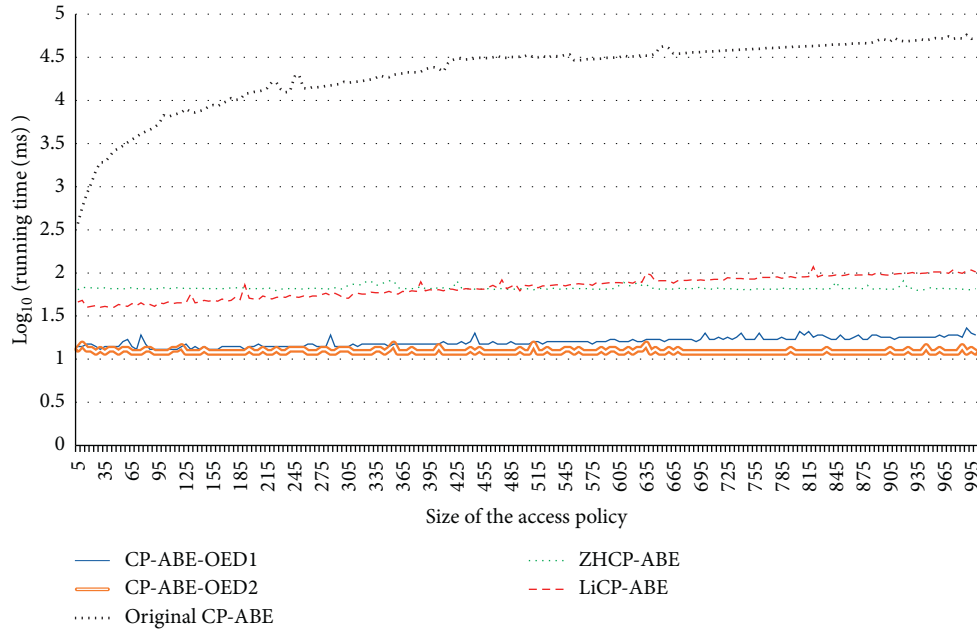


FIGURE 4: User-side encryption running time comparison.

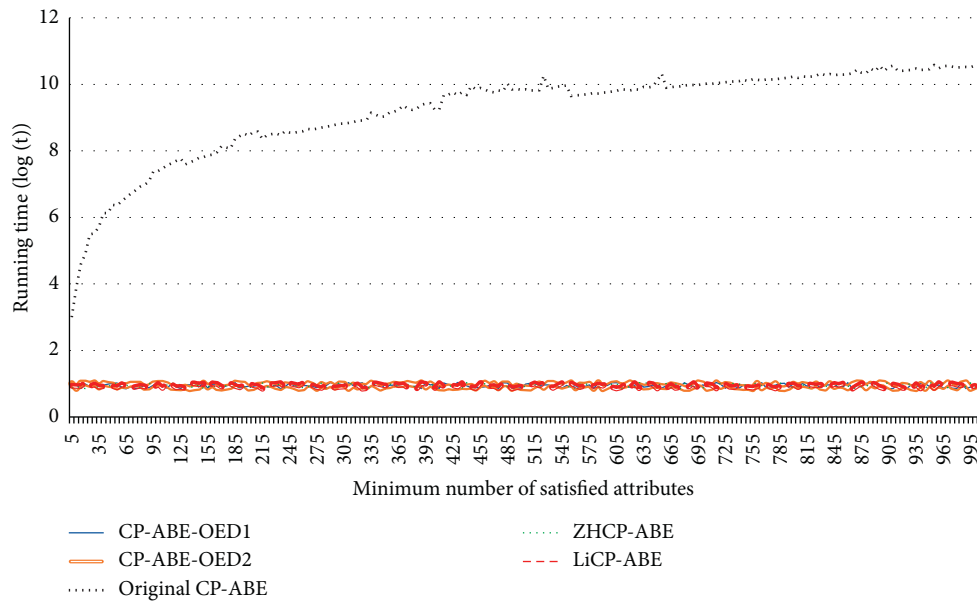


FIGURE 5: User-side decryption running time comparison.

only one G_T element in the other schemes. In addition, the user should first communicate his TK to the ABE Service Provider, which costs $(2 + 2 \cdot |S|)$ elements in G and $(1 + |S|)$ elements in Z_p for [26], $(1 + 2 \cdot |S|)$ elements in G for [23], and $(2 + |S|)$ elements in G for [12, 21, 27]. In [21], the user also receives the commitment from CSP, which costs one G element. In our proposed schemes, the user does not need to send or receive anything from CSP or the ABE Service Provider except transCT; TK is already sent by TA to the ABE Service Provider in the Registration Phase. Thus, our proposed schemes are the most efficient schemes in terms of

user's communication cost produced in the decryption phase.

Table 5 compares the user's storage cost for each scheme. In general, [26] is the scheme that requires the biggest user-side storage space to store TK, DK, and two Intermediate Ciphertexts (ITs). The user in [27] stores $|U|$ elements in G and one Z_p element for the encryption transformation key ETK, and one Z_p element for the decryption key DK. Thus, for large universe applications, [27] is considered the most storage space consuming scheme for users. In [12, 21, 23], the user stores TK and DK. DK costs one Z_p element in all

the schemes and TK costs $(1 + 2 \cdot |S|)$ elements in G for [23] and $(2 + |S|)$ elements in G for [12, 21]. In our proposed schemes, the user stores the encryption key EK and the decryption key DK; each of them costs only one Z_p element. Therefore, our proposed schemes are the most lightweight schemes in terms of user-side storage.

5.2. Experimental Results and Analysis. In this section, we will experimentally compare the running times of the user-side encryption and decryption of our 1st outsourced CP-ABE scheme (CP-ABE-OED1), our 2nd outsourced CP-ABE scheme (CP-ABE-OED2), the original CP-ABE scheme [4], ZHCP-ABE [23], and LiCP-ABE [27]. The implementations of the studied schemes were developed in Java using the JPBC Library [32] and the hashes were computed using setFromHash method of the *Element* class based on *SHA-256*.

We run 200 experiments for each $N = \{5, 10, 15, 20, 25, \dots, 1000\}$ on a Windows 8.1 Core i7 2 GHz PC with 8 GB of RAM where the access policy is defined as follows $(A_1 \text{ AND } A_2 \text{ AND } A_3 \dots \text{ AND } A_N)$ and the user's set of attributes is $\{A_1, A_2, A_3, \dots, A_N\}$. This approach simulates the worst-case scenario where the decryption phase depends on all the access policy's components. For each N , we repeat the experiment 10 times and calculate the average running time in milliseconds to smooth any experimental variability.

In Figure 4, the x -axis represents the size of the access policy and the y -axis represents the Log_{10} of the user-side encryption running time in milliseconds.

The experimental results confirmed our theoretical results. Besides, the theoretical results showed that CP-ABE-OED1 is more efficient than ZHCP-ABE [23] (respectively, LiCP-ABE [27]) for access policies with less than 500 leaf nodes (respectively, for access policies with less than 630 leaf nodes). However, the experimental results showed that CP-ABE-OED1 is more efficient than ZHCP-ABE [23] and LiCP-ABE [27] for all the access policy sizes up to 1000.

We observe that the difference in running time between CP-ABE-OED1 and CP-ABE-OED2 is linearly increasing with a relatively small slope, and this is due to the number of multiplications and hashing operations performed in CP-ABE-OED1 that is linear to the size of the access policy.

In Figure 5, the x -axis represents the size of the user's set of attributes and the y -axis represents the Log_2 of the user-side decryption running time in milliseconds.

As expected, the running times of the user-side decryption in all the studied outsourced CP-ABE schemes are constant and equivalent; that is because they all used the same decryption outsourcing technique firstly proposed by [12]. The user needs only about 2 ms (since $\text{Log}_2(t) = 1$ according to Figure 5) to decrypt a ciphertext regardless the size of the access policy or the length of her set of attributes.

6. Conclusion

In this paper, we proposed two efficient CP-ABE Key Encapsulation Mechanisms that can be provided as services in the cloud, minimizing the user-side computation,

communication, and storage costs. The first scheme is suitable for applications where the ABE Service Provider is untrusted, whereas the second scheme, which is more efficient, requires the ABE Service Provider to be at least semi-trusted. Both schemes are proved to be selectively CPA-secure in the random oracle. However, our systems support only one TA that is responsible for the registration of all the users. Hence, our systems will face a bottleneck problem if TA does not use a very powerful device or if the registration requests are very frequent. Therefore, in the future, it will be interesting to extend our schemes to use a multi-authority architecture to handle this problem. Converting our schemes to support a multi-authority architecture might also improve the security of the systems by preventing the key-escrow problem produced when attackers compromise the TA's master key. In a multi-authority approach, compromising some authorities' master keys by attackers will not break the system.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the National Center for Scientific and Technical Research (CNRST) (scholarship number: 4UIZ2017).

References

- [1] M. Armbrust, A. Fox, R. Griffith et al., "A View of Cloud Computing: Clearing the clouds away from the true potential and obstacles posed by this computing capability," *Communications of the ACM*, vol. 53, no. 4, pp. 50–58, 2010.
- [2] A. Sahai and B. Waters, "Fuzzy identity-based encryption," in *Proceedings of the 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 457–473, Aarhus, Denmark, May 2005.
- [3] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *Proceedings of the 13th ACM Conference on Computer and Communications Security—CCS '06*, p. 89, Alexandria, VA, USA, October 2006.
- [4] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *Proceedings of the 2007—IEEE Symposium on Security and Privacy*, pp. 321–334, Berkeley, CA, USA, May 2007.
- [5] Y. Yacobi, "A fast attribute based encryption," *IACR Cryptology ePrint Archive*, vol. 304, 2016.
- [6] S. Hohenberger and B. Waters, "Attribute-based encryption with fast decryption," in *Proceedings of the International Conference on Public-Key Cryptography—PKC 2013*, pp. 162–179, Nara, Japan, February 2013.
- [7] K. Zhang, J. Ma, J. Zhang, Z. Ying, T. Zhang, and X. Liu, "Online/offline traceable attribute-based encryption," *Journal*

- of *Computer Research and Development*, vol. 55, pp. 216–224, 2018.
- [8] S. Hohenberger and B. Waters, *Online/Offline Attribute-Based Encryption*, in *Proceedings of the IACR International Conference on Public-Key Cryptography*, Buenos Aires, Argentina, March 2014.
- [9] S. Ding, C. Li, and H. Li, “A novel efficient pairing-free CP-ABE based on elliptic curve cryptography for IoT,” *IEEE Access*, vol. 6, pp. 27336–27345, 2018.
- [10] V. Odelu and A. K. Das, “Design of a new CP-ABE with constant-size secret keys for lightweight devices using elliptic curve cryptography,” *Security and Communication Networks*, vol. 9, no. 17, pp. 4048–4059, 2016.
- [11] V. Odelu, A. K. Das, and A. Goswami, “An efficient CP-ABE with constant size secret keys using ECC for lightweight devices,” *IEEE Transactions on Consumer Electronics*, vol. 62, pp. 1–15, 2016.
- [12] M. Green, S. Hohenberger, and B. Waters, “Outsourcing the decryption of ABE ciphertexts,” in *Proceedings of the 20th USENIX Conference on Security*, p. 34, Berkeley, CA, USA, August 2011.
- [13] B. Waters, “Ciphertext-Policy Attribute-Based Encryption: An Expressive, Efficient, and Provably Secure Realization,” in *Proceedings of the IACR International Conference on Public-Key Cryptography*, vol. 6571, pp. 1–25, Taormina, Italy, March 2011.
- [14] T. Elgamal, “A public key cryptosystem and a signature scheme based on discrete logarithms,” *IEEE Transactions on Information Theory*, vol. 31, pp. 469–472, 1985.
- [15] M. Green, A. Akinyele, and M. Rushanan, *Libfenc*, The Functional Encryption Library.
- [16] E. Fujisaki and T. Okamoto, “Secure integration of asymmetric and symmetric encryption schemes,” in *Proceedings of the Annual International Cryptology Conference*, pp. 537–554, Santa Barbara, CA, USA, August 1999.
- [17] J. Lai, R. H. Deng, C. Guan, and J. Weng, “Attribute-based encryption with verifiable outsourced decryption,” *IEEE Transactions on Information Forensics and Security*, vol. 8, pp. 1343–1354, 2013.
- [18] Q. Li, J. Ma, R. Li, X. Liu, J. Xiong, and D. Chen, “Secure, efficient and revocable multi-authority access control system in cloud storage,” *Computers & Security*, vol. 59, pp. 45–59, 2016.
- [19] B. Qin, R. H. Deng, S. Liu, and S. Ma, “Attribute-based encryption with efficient verifiable outsourced decryption,” *IEEE Transactions on Information Forensics and Security*, vol. 10, pp. 1384–1393, 2015.
- [20] S. Lin, R. Zhang, H. Ma, and M. Wang, “Revisiting attribute-based encryption with verifiable outsourced decryption,” *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 10, pp. 2119–2130, 2015.
- [21] X. Mao, J. Lai, Q. Mei, K. Chen, and J. Weng, “Generic and efficient constructions of attribute-based encryption with verifiable outsourced decryption,” *IEEE Transactions on Dependable and Secure Computing*, vol. 13, no. 5, pp. 533–546, 2016.
- [22] T. P. Pedersen, “Non-interactive and information-theoretic secure verifiable secret sharing,” in *Proceedings of the 12th Annual International Cryptology Conference*, pp. 129–140, Santa Barbara, CA, USA, August 1992.
- [23] Z. Zhou and D. Huang, “Efficient and secure data storage operations for mobile cloud computing,” in *Proceeding of the 2012 8th international conference on network and service management (cnsm) and 2012 workshop on systems virtualization management (svm)*, pp. 37–45, Las Vegas, NV, USA, October 2012.
- [24] J. Li, C. Jia, J. Li, and X. Chen, “Outsourcing encryption of attribute-based encryption with MapReduce,” in *Proceedings of the 14th International Conference, ICICS 2012*, pp. 191–201, Hong Kong, China, October 2012.
- [25] M. Asim, M. Petković, and T. Ignatenko, “Attribute-based encryption with encryption and decryption outsourcing,” in *Proceedings of the 12th Australian Information Security Management Conference*, pp. 21–28, Perth, Western Australia, December 2014.
- [26] R. Zhang, H. Ma, and Y. Lu, “Fine-grained access control system based on fully outsourced attribute-based encryption,” *Journal of Systems and Software*, vol. 125, pp. 344–353, 2017.
- [27] J. Li, X. Li, L. Wang, D. He, H. Ahmad, and X. Niu, “Fuzzy encryption in cloud computation: efficient verifiable outsourced attribute-based encryption,” *Soft Computing*, vol. 22, no. 3, pp. 707–714, 2018.
- [28] P. Zhang, Z. Chen, J. K. Liu, K. Liang, and H. Liu, “An efficient access control scheme with outsourcing capability and attribute update for fog computing,” *Future Generation Computer Systems*, vol. 78, pp. 753–762, 2018.
- [29] J. Blömer, P. Günther, V. Krummel, and N. Löken, “Attribute-based encryption as a service for access control in large-scale organizations,” in *Proceedings of the 11th International Symposium, Foundations and Practice of Security*, pp. 3–17, Montreal, QC, Canada, November 2018.
- [30] A. Beigel, “Secure schemes for secret sharing and key distribution,” *Tech. Inst. Technol. Fac. Comput. Sci.*, 1996.
- [31] A. Lewko and B. Waters, “Decentralizing attribute-based encryption,” in *Proceedings of the 30th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 568–588, Tallinn, Estonia, May 2011.
- [32] A. De Caro and V. Iovino, “jPBC: Java pairing based cryptography,” in *Proceedings of the 16th IEEE Symposium on Computers and Communications, ISCC 2011*, pp. 850–855, Kerkyra, Corfu, Greece, June 28 - July 1 2011.

Research Article

PurExt: Automated Extraction of the Purpose-Aware Rule from the Natural Language Privacy Policy in IoT

Lu Yang ^{1,2} Xingshu Chen ^{2,3} Yonggang Luo ^{2,3} Xiao Lan ^{2,3} and Li Chen^{1,2}

¹College of Computer Science, Sichuan University, Chengdu 610065, China

²Cyber Science Research Institute, Sichuan University, Chengdu 610065, China

³College of Cyber Science and Engineering, Sichuan University, Chengdu 610065, China

Correspondence should be addressed to Yonggang Luo; iamlyg98@scu.edu.cn

Received 15 February 2021; Revised 14 April 2021; Accepted 24 April 2021; Published 8 May 2021

Academic Editor: Ahmed Meddahi

Copyright © 2021 Lu Yang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The extensive data collection performed by the Internet of Things (IoT) devices can put users at risk of data leakage. Consequently, IoT vendors are legally obliged to provide privacy policies to declare the scope and purpose of the data collection. However, complex and lengthy privacy policies are unfriendly to users, and the lack of a machine-readable format makes it difficult to check policy compliance automatically. To solve these problems, we first put forward a purpose-aware rule to formalize the purpose-driven data collection or use statement. Then, a novel approach to identify the rule from natural language privacy policies is proposed. To address the issue of diversity of purpose expression, we present the concepts of explicit and implicit purpose, which enable using the syntactic and semantic analyses to extract purposes in different sentences. Finally, the domain adaption method is applied to the semantic role labeling (SRL) model to improve the efficiency of purpose extraction. The experiments that are conducted on the manually annotated dataset demonstrate that this approach can extract purpose-aware rules from the privacy policies with a high recall rate of 91%. The implicit purpose extraction of the adapted model significantly improves the F1-score by 11%.

1. Introduction

IoT applications cover all aspects of people's lives, such as smart homes, wearable devices for health management, and traffic monitoring. While enjoying the convenience brought by IoT applications, users also bear a greater risk of personal information leakage than general applications. The data collected by IoT applications comes not only from smartphones but also from various smart devices that are closer to users' daily lives, so the data is more sensitive [1]. The collection and use of data should be more strictly in compliance with regulations. Furthermore, because of the potentially unobtrusive nature of IoT data collection, users may not be aware of what information is collected and why it is collected [2, 3]. To solve the above issues, data protection laws and regulations, such as the General Data Protection Regulation (GDPR), require that before collecting and using data, IoT application providers must inform users of the

privacy policies and obtain their consent to this statement [4]. A privacy policy describes the data practices of an application [5], especially what data is collected and how it is used. However, the tedious and complicated writings of the privacy policy hinder users from reading and understanding these policies [6, 7]. As a legal agreement, the natural language privacy policy lacks a machine-readable form to handle automated compliance verification, that is, whether the privacy policy provides all the information for legal requirements and the implementation complies with the privacy statements [8, 9].

There are various analysis tools to extract key information from the privacy policies to help users quickly access the policies of interest. A common practice for these tools is to classify and label the statements in the privacy policy into categories such as first-party collection and third-party sharing [10–12]. The result of rough classification is that users still need to read the statements to obtain the details,

such as the intention of the data collection. To overcome this problem, some tools introduce manual labor with specific domain knowledge to annotate the fine-grained information in the privacy policy [13], one of which requires a mean of 72 minutes per policy [5]. This solution suffers from inefficiency and is time-consuming. In terms of compliance, recent works have begun to extract structured data collections or usage statements from privacy policies to analyze data violations [14, 15]. These studies are focused on the use of undeclared data and rarely consider whether the data are used for the eligible purpose stated in policy privacy. The purpose is the key concept in data protection regulations [16]. The GDPR clearly spells out the purpose limitation; that is, “personal data shall be collected for specified, explicit, and legitimate purposes and not further processed in a manner that is incompatible with those purposes.” Consequently, the expression and extraction of the purpose statement in the privacy policy are essential for data compliance [17]. However, the purpose expression is various and can be a noun or a verb phrase in natural language documents. It is not like the action element, which is naturally a verb, nor organization entities, such as “Google,” “Facebook,” or data objects, such as “name” and “e-mail addresses,” which are types of named entities. Therefore, the extraction of purpose not only can rely on just lexical and syntactic analysis like extracting actions, entities, and data objects but also requires semantic analysis.

To overcome the above issues, we propose an approach for expressing and automatically extracting purpose-aware rules from the privacy policy, which is called PurExt. First, data collection and use statement in the privacy policy is formalized as a purpose-aware rule, which is a tuple of four elements as actor, action, data object, and purpose. The rule indicates that some *data objects* are collected or used (*action*) by a certain *actor* for a certain *purpose*. Then, through the investigation of the privacy policies, this study defines two types of purpose expressions, explicit purpose and implicit purpose, and implements the automated extraction of rule elements by combining syntactic and semantic analyses. Finally, the effectiveness of PurExt is verified on privacy policy datasets that were collected from IoT apps and manually annotated. This method can be applied in the following ways. First, this approach can be utilized to mine access control policies (ACPs) from security specifications because purpose-aware rules are nearly a draft of the ACPs. Second, PurExt can help privacy policy analysis tools to enhance the ability of fine-grained information extraction. Third, the extraction results of PurExt can be the basis and starting point for conducting data purpose compliance research.

The major contributions of this work are as follows:

- (i) A purpose-aware rule is proposed to formalize and express the data collection and use statement in privacy policies. The elements of the rule are driven by the data collection and use purpose, which is the core concept of privacy protection.
- (ii) The automated rule extraction of natural language privacy policies is proposed. Explicit and implicit

purposes are defined to specify the purpose expression and are identified by combining SRL with syntactic analysis. To the best of our knowledge, this is the first study to extract purpose-centric rules from privacy policies. Experiments on datasets from the real IoT-related apps have proved the effectiveness of the algorithm.

- (iii) Domain adaption is utilized to improve the efficiency of SRL in the field of privacy policies. The experiments show that the F1-score of the implicit purpose extraction by the domain-adapted SRL model increases 11%.

The rest of this paper is organized as follows: in Section 2, the related work in the field of privacy policy analysis in IoT, policy extraction with NLP, and SRL tools is briefly introduced. And we elaborate on the core concept of PurExt in Section 3. After that, a detailed description of PurExt is presented in Section 4. Section 5 presents the results of our extensive experimental evaluation of the proposed approach. The last section concludes the paper and points out the future research direction.

2. Related Work

2.1. Privacy Policy Analysis in IoT. By being aware of the user-unfriendly problem with respect to the privacy policy and its importance for a compliance evaluation, several studies on privacy policy analysis have emerged in recent years. Parvaneh et al. [18] utilized classification and graph-based methods to make privacy policies that are structured and categorized to help users understand them better. Onu et al. [19] defined a taxonomy framework, which uses a tree-like hierarchical form to model privacy policies within IoT environments. A framework called Polisis [20] was proposed to divide a privacy policy into fragments and assign the fragments with a set of labels that describes its data practices. Subahi and Theodorakopoulos [21] proposed eight criteria for the IoT privacy policy and implemented a test bed for ensuring the compliance of the IoT data disclosure to the corresponding privacy policy. To verify whether an application behaves according to its privacy policy, Zimmeck et al. [22] proposed using an automated analysis system that is based on machine learning and static analysis to identify potential privacy requirement inconsistencies. By considering the potential contradictions in a single privacy policy, Benjamin et al. presented an automated analysis tool called PolicyLint [14], to extract the structured data collection and sharing statements and identify contradictions among them. Furthermore, he proposed POLICHECK [23], which is an entity-sensitive flow-to-policy consistency model that is based on the results extracted from PolicyLint. Several works [24, 25] had modeled the identification of the data practice statements in privacy policies as a classification problem. The classification results that are obtained in this manner are some coarse-grained labels, which are not conducive to the compliance analysis of specific terms, which includes whether the processing of specific data adheres to the declared intention. Bhatia and Breau [11, 26] presented a

semantic frame-based representation for data practice that can be used to identify incompleteness in a data action context. This work introduces semantic roles into the analysis of privacy policy. However, they use an inefficient way of manual annotation instead of the automatic semantic role labeling tool.

2.2. Policy Extraction with NLP. The earliest attempt to extract the data access information from the natural language security specifications is to automatically extract the ACPs with NLP. Xiao et al. [27] proposed the first work to extract ACPs from natural language software documents and produce the formal specifications, called Text2Policy. The sentences describing the ACPs were first separated from other unrelated texts by performing matching with four predefined patterns. Then, by using the annotated portions of the matched pattern, they identified the subject, action, and resource elements from the sentence. The major drawback of this approach is that ACP sentences other than the predefined patterns cannot be discerned. It has been confirmed that only 34.4% of the ACP sentences were found by matching the four patterns [28]. Nevertheless, this work is still enlightening, resulting in many follow-up studies. Slinkas and Williams [29] proposed the concept of access control relation extraction (ACRE), which is a method of incorporating machine learning and NLP to extract ACP elements. They used classification algorithms to determine whether these sentences are related to the access control. A bootstrapping process was adopted to extract the ACP instance from a small set of seeded dependency graph patterns. Subsequently, an extended ACRE was proposed [28]. Unlike the previous approach, the votes for the K-nearest neighbor (KNN), naive Bayes, and simple vector machine were replaced by the KNN classification algorithm in the sentence identification phase. Narouei et al. [30] introduced four different types of features to improve the effect of distinguishing the ACP sentences from other sentences. Subsequently, they tried to use semantic role labeling to identify ACP elements [31], but the identification effects of this method on different data sets are quite different. All the above methods can be used for policy extraction. However, their extracted objects are the elements of ACPs, such as roles, resources, and attributes and barely involve data collection and use purpose, which is an important concept in the privacy policy.

2.3. SRL Tools. SRL is a shallow semantic parsing task, in which the goal is to identify the arguments of the verb predicate in a sentence and assign semantic labels to those arguments [32]. SRL starts with the action predicate of a sentence to determine other sentence constituents that correspond to who did what to whom, when, where, and why. This information is useful for identifying the key concepts of the data collection and usage statements from a sentence. To evaluate the effects of different SRL tools in extracting the structured data collection and usage statements, the PurExt algorithm is implemented based on four SRL tools. EasySRL [33], which is written in Java, provides a

semantic role labeler and combinatory categorial grammar parser. Mate-tools [34] provide a pipeline of modules that perform the lemmatization, part-of-speech (POS) tagging, dependency parsing, and SRL of a sentence. The tools are language-independent and have high accuracy. Semantic/syntactic extraction using a neural network architecture (SENNA) [35] is a SRL program that is trained on a PropBank corpus, which also offers other common NLP tasks such as POS tagging, chunking, and named entity recognition (NER). Unlike other SRL systems, SENNA assigns semantic roles to sentence constituents without the help of a syntax tree, thus resulting in better efficiency. SwiRL [36] is a SRL system for English that is constructed on top of the full syntactic analysis of the text. SwiRL has a user-friendly feature; that is, the model can be retrained through the application programming interface (API) that is provided by the system, which enables the user to add domain-specific knowledge.

3. Purpose-Aware Rule

In this chapter, we define the purpose-aware rule based on the meaning of purpose in the privacy policy, which specifies the objects to be extracted. And the purpose expression in natural language is analyzed to clarify the extraction method.

3.1. Purpose in Privacy Policy. Purpose is the key concept in privacy policy, which explains the reason for data collection and use. It determines whether the user will agree to the data collection and use behavior of an application. Although the purpose is shown as a few words or phrases in the privacy policy, the connotation of purpose is a constraint on what data is collected and how it is used.

Example 1. We illustrate the meaning of purpose using the privacy policy from Xiaomi Wear App. A statement of “to facilitate the registration of your smart wearables in the app, we may collect the information related to your Mi Account, identifier of smart wearables, identifier of your phone (IMEI number encrypted via Hash algorithm), phone model, OS version, and Bluetooth information of smart wearables” in the privacy policy indicates that the data such as account, identifier of device, and OS version is only allowed to be collected for the purpose of the registration of smart wearables, as shown in Figure 1.

From the example, we can see that a purpose is specific to the related data objects, the action performed on the data, and the actor that performs the action. Driven by this insight, a data collection and use statement is formalized as a purpose-aware rule, which can be regarded as a collection of actors, actions, and data objects driven by a purpose.

Definition 1 (purpose-aware rule). *A purpose-aware rule is defined as a tuple:*

$$\text{rule} = \{\text{actor, action, data object, purpose}\}, \quad (1)$$

where *actor* is the entity that performs the action on a data object, *action* is the operation performed on the data objects,

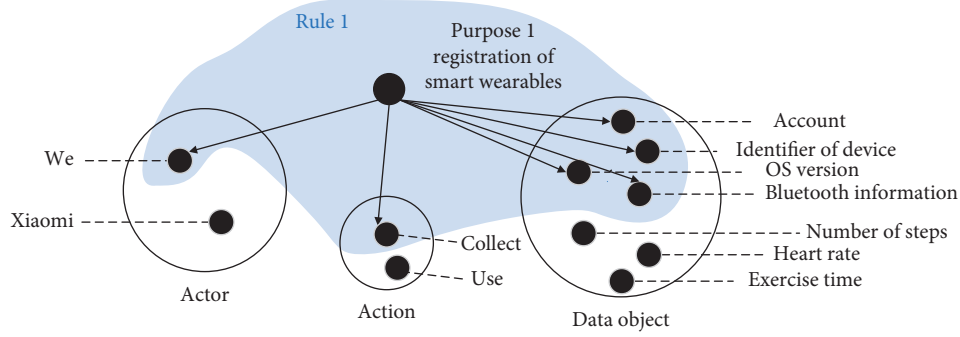


FIGURE 1: An example of the purpose constraint.

data object is the data related to the purpose, and *purpose* is the intention of data collection and usage.

Example 2. The statement in Example 1 can be formalized as a purpose-aware rule like $r_1(\{we\}, \{collect\}, \{Mi \text{ Account, identifier of smart wearables, identifier of your phone (IMEI number encrypted via Hash algorithm), phone model, OS version, Bluetooth information of smart wearables}\}, \text{the registration of your smart wearables in the app})$.

A benefit of the rule is the direct or internal connection between its elements and the concepts of data access control at the implementation layer. Although the rule elements extracted from the privacy policies are mostly abstract, the high-level user-oriented rule can be refined into low-level operational policies via hierarchical and deductive methods [37, 38]. For example, the data object “personal information” that most frequently appears in the privacy policy can be refined and mapped into fields such as “name” and “age” in the database. On the contrary, it is also possible to judge whether the underlying data practice complies with the data use statement in the privacy policy through induction of elements.

3.2. Purpose Expression in Natural Language. Concerning the expression of the rule elements in natural language, actor and data object are usually nouns and regarded as a type of named entity [14], and action is present as a verb. Consequently, they can be identified and extracted from the natural language sentences based on these lexical features. However, the appearance of purpose in sentences is ambiguous.

Because the purpose is to explain the reason for the data collection and use behaviors, it is usually acted as the semantic role of the purpose attach with some keywords, such as “in order to” and “for,” to modify the verbs of collection and use action in the sentence; for example, “we use purchase records in order to analyze user preferences.” A statistic shows that 78% of the usage actions in the privacy policy are modified by the purpose semantic role [26]. According to our observations on a large number of privacy policies, this is also the most common way in which the purpose appears in the privacy policy document. Meanwhile, we found another appearance of purpose like “the purpose of collecting purchase records is preference analysis.” In this

case, the predicate of the sentence is no longer a collection or use (CoU) verb, so the purpose cannot be attached to the predicate as a semantic role but can only appear independently. Through analyzing the structure and key constituents of the two kinds of sentences, two patterns of purpose expression in natural language privacy policies are concluded as follows:

3.2.1. Explicit sentence. The backbone of this kind of sentence is to state the purpose straightforwardly. The pattern is $P_{\text{noun}} + V_{\text{link}}/V_{\text{contain}} + \text{Purpose}$, where P_{noun} is the noun representing purpose, V_{link} is the linking verb such as “is” and “are,” and V_{contain} is the verb representing the meaning of containing.

Example 3. “The purpose of collecting your location data and speed is to analyze your train statistics” and “the reason of using your name and e-mail address includes website account registration” are explicit sentences.

Implicit sentence: the backbone of this kind of sentence is to state the data collection or use behavior, and the collection or use action is modified by the purpose semantic role. The pattern is

$$sbj + V_{CoU} + \text{Data} + \text{Purpose}, \quad (2)$$

where V_{CoU} is the verb representing collection and use actions, sbj is the subject of the V_{CoU} , and Data represents the collected and used data objects.

Example 4. “The app will collect your heart rate and pulse to make suggestions for future workout” is an implicit sentence.

The two patterns only represent different manifestations of the purpose in the natural language sentences, but the semantics of the purpose will not differ depending on its manifestation. The definition and feature of the two patterns also determine the extraction methods for the two purposes of them. The purpose in the implicit sentence can be readily identified using semantic role labeling, while the purpose in the explicit sentence can be extracted based on the syntactic features of the pattern. To distinguish the source of the extracted purpose, the purpose extracted from an explicit sentence is called the *explicit purpose*, and the purpose from an implicit sentence is called the *implicit purpose*.

4. PurExt

In order to identify the purpose-aware rules from the privacy policies, we propose an automated rule extraction framework, PurExt. An overall view of the proposed approach is shown in Figure 2. Details of each step are described in the following sections.

4.1. Preprocessing. This study used the open-source library, spaCy [39], to perform a series of essential natural language preprocessing on sentences. SpaCy is a fast NLP toolkit that is implemented in Python. The tokenization, POS tagging, and NER of a sentence can be easily and serially completed by constructing a pipeline task. The tokenization splits the text into individual words, which allows us to obtain the smallest unit of processing. The POS tag that is assigned to the token provides the lexical information that is needed for NER and dependency parsing. As an important preliminary task of relationship extraction, the goal of NER is to identify phrases that represent real-world objects, such as the name, geographic location, organization, and date, and label them with corresponding named entity tags. PurExt applies a domain-adapted NER model to label the entities concerning the field of the privacy policy. For example, “name” and “age” are annotated as the named entity type of *Data Object*, and “we” and “advertisers” are annotated as the named entity type of *Entity*. The NER model has the ability to identify *Data Object* and *Entity* that are specific to privacy policies by retraining spaCy’s NER engine with 500 annotated sentences from the privacy policies.

4.2. Sentence Classification. The goal of sentence classification is to divide the sentences into explicit sentences, implicit sentences, and other sentences. Both explicit sentences and implicit sentences are related to the data collection or use statement, as well as the target of the rule extraction, so we call them CoU sentences for brevity. Other sentences describe the information such as terms of service, data retention, policy updates, or contact information. Because they have nothing to do with the data collection or use statement, they are not processed.

As described in Algorithm 1 SentenceClassification (), the first step of sentence classification is to build a dependency tree for each sentence. The parsed dependency tree sketches the lexical structure of a sentence by constructing a dependency relationship between words.

Sentence classification starts from the root node of the dependency tree. Explicit sentences have two methods of stating the purpose. (1) The predicate of the sentence indicates the relationship of existence, which is usually a linking verb (denoted as V_{link}), meaning “what is the purpose.” (2) The predicate indicates the containment relationship, which is usually a verb that represents the meaning of containing (denoted as $V_{contain}$), such as “include” and “contain,” which conveys “what does the purpose include.” Therefore, if the predicate of the sentence satisfies one of the conditions, the sentence has the potential to be an explicit sentence. However, if the predicate describes the data

collection or use behaviors (for brevity, these verbs are called CoU verbs, which are listed in Table 1), the sentence is a potential implicit sentence. Those that do not meet the above two conditions are classified as other sentences and will not be analyzed. Candidate sentences that are filtered by the predicate are identified as explicit sentences in two ways. First, the subject of its predicate is the purpose noun (called Pnoun for brevity, which is listed in Table 1), that is, a noun that refers to the purpose. Second, the Pnoun is modified by a complement that describes the data collection or use statement. This constraint aims to avoid the wrong purpose affiliation. The additional condition of the implicit sentence is to contain at least one *Data Object*. Through sentence classification, the processing range is narrowed, and the location of the elements that is to be extracted is roughly located.

4.3. Rule Extraction. The goal of this step is to extract the actor, action, data object, and purpose from the sentences to form the data security rules. Because of the different expressions of the purpose, the element extraction of explicit sentences is based on the syntactic analysis, whereas the extraction of implicit sentences is mainly based on the semantic analysis.

4.3.1. Element Extraction of Explicit Sentences. According to the definition of the explicit sentence, the subject of the predicate, that is, Pnoun, is modified by the data collection or use statement, which indicates the affiliation of the purpose. Therefore, the extraction of the action and data object begins by parsing the prepositional complement of the Pnoun. PurExt traverses down the branch to identify the node matching CoU verbs as an action element and extracts all the nodes that are annotated as data objects. The purpose expression in explicit sentences is derived into verbs and nouns. The first one is usually used as a complement to modify the predicate, while the other appears in the form of an object. The verbal purpose expression is associated with the predicate by the “xcomp” dependency label, which means that it is the open clause complement of the predicate. PurExt parses this branch and forms a purpose phrase. As for the purpose expression in the noun form, PurExt will parse the branch, linking to the predicate with a “dobj” (direct object) dependency tag or a “pobj” (object of a preposition) tag, beginning with a preposition. This is because there is a case where the purpose phrase and predicate are connected by a preposition; for example, “the purpose of collecting your personal information is for legal obligations.” Actor elements are usually not involved in explicit sentences.

Example 5. The two explicit sentences in Example 3 have the purpose of verb form and noun form, respectively. The dependency trees of the sentences are shown in Figure 3. And the two sentences can be extracted as $r_1(\{\}, \{\text{collect}\}, \{\text{your location data, speed}\}, \text{analyze your train statistics})$, and $r_2(\{\}, \{\text{use}\}, \{\text{your name, e-mail address}\}, \text{website account registration})$.

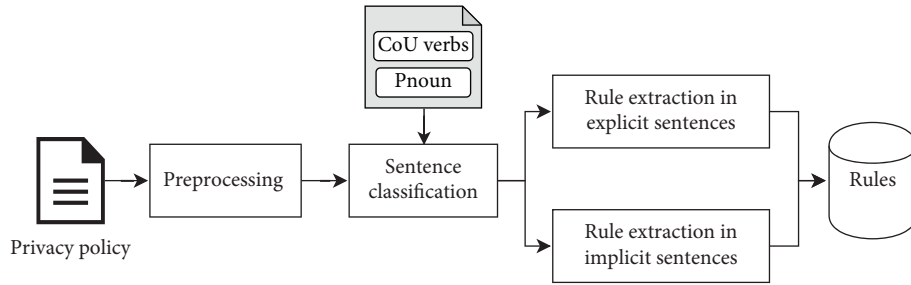


FIGURE 2: Overview of PurExt.

Input: a sentence S to be classified.

Output: a sentence category label in (E, I, O) , where E for explicit sentence, I for implicit sentence, and O for other sentences.

- (1) Construct a dependency tree structure T of S
- (2) Let p be the root of T
- (3) **if** p is a V_{link} or $V_{contain}$ **then**
- (4) **if** the subject s of p is a Pnoun **then**
- (5) **if** s is modified by a complement containing at least one CoU verb **then**
- (6) **return** E .
- (7) **else**
- (8) **return** O
- (9) **end if**
- (10) **else**
- (11) **return** O
- (12) **end if**
- (13) **else if** p is CoU verb **then**
- (14) **if** the object of p contains at least one *Data Object* **then**
- (15) **return** I
- (16) **else**
- (17) **return** O
- (18) **end if**
- (19) **else**
- (20) **return** O
- (21) **end if**

ALGORITHM 1: SentenceClassification ().

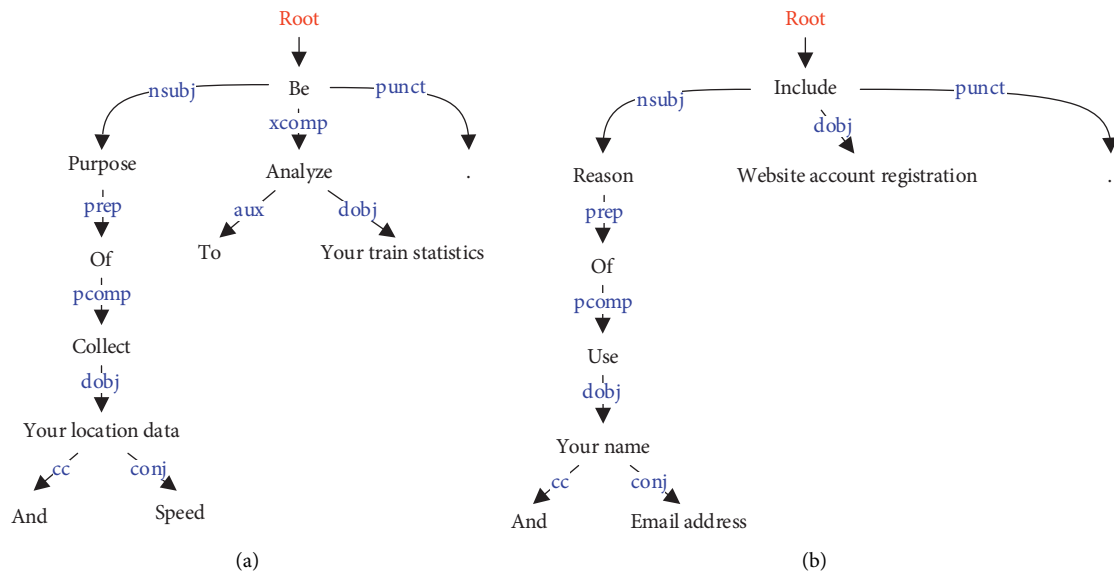


FIGURE 3: Dependency tree of the explicit sentence whose purpose representation is in (a) verb form and (b) noun form.

4.3.2. Element Extraction of Implicit Sentences. The purpose phrase is not the main component in implicit sentences. As the role of modifying other ingredients, it has various forms. Semantic analysis can cross the different syntactic structures of a sentence to present its semantic information. For example, although the grammatical construction of “A car hit Bob” and “Bob was hit by a car” are different, the semantic information for them is the same and can be unified as hit (car, Bob). Therefore, PurExt adopts SRL [40], a shallow semantic parsing, to infer the purpose of the data collection or use statements in implicit sentences. The SRL detects the semantic arguments that are related to the verbs in a sentence, to assign semantic roles that are defined by the PropBank project to the constituents of the sentence. The meaning of common semantic roles is shown as follows:

- V:** verb.
- A0:** agent of the verb.
- A1:** patient.
- AM-MOD:** modal.
- AM-PNC:** purpose.

The connotation of a data security rule is that an actor performs collection or use actions on data objects for a certain purpose. Reasonably, the actor corresponds to the “A0” role of the CoU verbs, the data objects match the “A1” role, and the purpose is the “AM-PNC” role. Hence, through this mapping, four elements of a rule can be extracted with the help of SRL. To improve the accuracy of the extraction, the constraint of the named entity type is also attached. In addition to being marked as “A0,” the actor element needs to be marked as *Entity* by NER. Similarly, data objects should be annotated as the “A1” role, as well as the entity type of *Data Object*.

Example 6. The sentence in Example 4 is annotated as “[The app **A0**] [will **AM-MOD**] [collect **V**] [your heart rate and pulse **A1**] [to make suggestions for future workout **AM-PNC**]” by the SRL tool. And the rule extracted from the sentence is $r(\{\text{the app}\}, \{\text{collect}\}, \{\text{your heart rate, pulse}\}, \text{make suggestions for future workout})$.

4.4. Domain Adaption. The success of the extraction primarily depends on the accurate annotation of the SRL. By considering the labeling effect and user-friendliness, we choose SwiRL to implement PurExt. SwiRL is a SRL tool that is trained on the PropBank corpus, which is a collection of annotated sentences from the Wall Street Journal. The terms and expressions in financial magazines are different from those in privacy policies. Therefore, the model that migrates into this problem domain is prone to mislabeling. For example, the purpose phrase that should be labeled as “AM-PNC” is incorrectly assigned to the “A2” role, which leads to false negative results. This is also confirmed by the experiment in Section 5.3.

To mitigate this problem, this study tried to adapt the SRL tool SwiRL to the privacy policy domain; that is, we use

a small number of domain-specific datasets to retrain SwiRL, such that it can obtain more accurate annotation results in the target domain. For the preparation, 400 purpose-related sentences of privacy policies are collected from real apps and manually marked with the semantic roles. The experiment in Section 5.4 proved that the effect of implicit purpose extraction, which uses SRL annotation, improves significantly after the domain adaption.

5. Experiments

In this section, we present three evaluations conducted to assess the effectiveness of our proposed approach. In our evaluations, we specifically focus on the following questions:

- RQ1: how effectively does PurExt extract the explicit purpose and implicit purpose?
- RQ2: is there any improvement in PurExt with domain adaption applied?
- RQ3: how *effectively* does PurExt extract the other three elements, that is, actor, action, and data object?

5.1. Dataset. Because of the lack of a public dataset annotated with the purpose statement, we collected and annotated a dataset from the privacy policies of real APPs, which mainly belong to the Wear OS, healthcare, and other IoT-related APP categories in Google Play Store. First, the privacy policy HTML files of each app were downloaded with a crawler program developed by authors. Then, a tool named HtmlToPlaintext [41] was used to convert these HTML files into plaintext policy documents. Finally, a doctoral student and a master student with background knowledge selected 1,000 sentences from these documents and annotated the phrase in each sentence with the element labels that they related to.

There are 750 CoU sentences, which consist of 584 implicit sentences with purpose statement, 46 without purpose, and 120 explicit sentences. In order to align the distribution of the dataset with the real privacy policy, there are 250 sentences describing another nine types of data practice [5], such as third-party sharing/collection, data retention, data security, and policy change. To perform and verify the domain adaptation experiment, about 68 percent of the 584 implicit sentences were used for training and 32 percent for testing. SwiRL model was retrained on the dataset composed of the 400 implicit sentences and its original training corpus. Apart from the 400 sentences, the remaining 600 sentences constitute the test dataset. Table 2 summarizes the number of different sentences along with their annotated elements in the test dataset.

5.2. Evaluation Criteria. To assess the effectiveness of the element extraction, the experimental results were measured with respect to the precision (P), recall (R), and F1-score (F1) [42], which are defined as follows: To compute these values, the experimental results are divided into four

TABLE 1: Word lists.

Type	Word
CoU verbs	Access, check, collect, disclose, gather, keep, know, obtain, process, provide, receive, request, retain, save, share, store, transfer, update, use, utilize
Pnoun	Purpose, reason, intention, goal, motivation, way

categories. True positive (TP) means extracting the element correctly. False positive (FP) represents a case in which an unrelated constituent is identified as an element. False negative (FN) is the prediction that a true element in a sentence is not recognized. Finally, true negative (TN) is a situation in which the approach correctly identifies a constituent to be unrelated:

$$\begin{aligned}
 P &= \frac{TP}{TP + FP}, \\
 R &= \frac{TP}{TP + FN}, \\
 F1 &= 2 * \frac{P * R}{P + R}
 \end{aligned}
 \tag{3}$$

5.3. Effect of Purpose Extraction

5.3.1. Experiment Setup. We separately counted the TP, FP, and FN values for the explicit and implicit purposes, to show the extraction effect of PurExt for the two types of purposes. To determine how different SRL tools affect implicit purpose extraction, PurExt is implemented based on the semantic annotations from four different SRL systems. The four SRL systems are EasySRL, Mate-tools, SENNA, and SwiRL, which were introduced in Section 2.3. Because explicit purpose extraction does not use SRL, the explicit purpose extraction results of the four SRL tools are the same.

5.3.2. Experiment Results. The result of the explicit purpose extraction is shown in Table 3. The position of the explicit purpose in the sentence is relatively fixed; thus, the precision of the extraction based on the syntactic structure is high. However, the recall rate for explicit purposes is comparatively lower. We analyzed the FNs in the results and found that the sentences, which were not detected by PurExt, do not have a clause modification of the subject that describes the data usage behavior, which is one of the criteria for an explicit sentence. For example, consider “the purpose is to process your payment.” The privacy policy document is rich in context; hence, the subordination of the purpose may appear in the title or elsewhere. This method represents sentence-level NLP, which will misjudge the sentences that are separated from their dependency.

Table 4 shows that the purpose identified from the sentences with four SRL tools did not reach half of the total. By digging into the annotations of the four SRL tools, we observed a phenomenon that several purpose phrases, which should be marked as the *AM-PNC* role, are assigned as the *A2* role. As mentioned in Section 4.4, these tools are trained

on another domain corpus, such as PropBank and FrameNet. Because of the difference in word usage and expression habits, they do not perform well in assigning semantic labels for the privacy policy documents.

5.4. Effect of Domain Adaption

5.4.1. Experiment Setup. Domain adaption is introduced to improve the recall rate of implicit purpose extraction via SRL. Considering the trainability of the tools and the effect that is shown in the first experiment, SwiRL, which is user-friendly and provides a retraining API, was chosen as the target. The training dataset is described in Section 4.4. This section demonstrates the overall effect of PurExt that is achieved with the domain-adapted SwiRL and a separate result of the implicit purpose extraction, which can directly reflect the effect of the domain adaption.

5.4.2. Experiment Results. Table 5 shows that retraining with a small amount of domain-specific data significantly improves the effect of SRL on implicit purpose extraction. In detail, the recall rate and F1-score increase by almost 13% and 11%, respectively. In the future, we will continue to explore the effect of the training dataset size on the retrained model. The overall extraction results of PurExt that are realized by the domain-adapted SwiRL are shown in Table 6. The precision of each element extraction is over 90%, and the precision of the rule reaches 97%. In terms of the recall rate, except for the purpose, which is 69%, the other elements are all over 85%. The F1-score for the rule reaches 91%.

5.5. Effect of Other Elements Extraction

5.5.1. Experiment Setup. To determine how effectively PurExt extracts the elements of the actors, actions, and data objects, PurExt is compared with a recent work PolicyLint [14] that is consistent with the targets that were extracted from the privacy policies in this investigation. PolicyLint is a privacy policy analysis tool that identifies potential contradictions that may arise inside the same privacy policy. It provides a sentence-level NLP method to capture sharing and collection statements in privacy policies as a four-tuple (actor, action, data object, entity), where entity corresponds to the object of data sharing. We run PolicyLint on the privacy policy dataset and compare its extraction results, just the three-tuple (actor, action, data object), with the results of our approach to evaluate the efficiency of PurExt on the extraction of actor, action, and data object.

TABLE 2: Test dataset statistics.

		#sentence	#actor	#action	#data object	#purpose
CoU sentences	Explicit	120	0	112	107	120
	Implicit	184	125	184	362	184
	Other	46	45	46	205	0
Unrelated sentences	250	0	0	0	0	

TABLE 3: Effect of explicit purpose extraction.

	P (%)	R (%)	F1 (%)
Explicit purpose	100.00	82.50%	90.41%

TABLE 4: Effect of implicit purpose extraction.

	Implicit purpose		
	P (%)	R (%)	F1 (%)
EasySRL	100.00	29.89	46.03
Mate-tools	97.75	47.28	63.74
SENNa	98.91	49.46	65.94
SwiRL	97.75	47.28	63.74

TABLE 5: Effect of domain adaption on implicit purpose extraction.

	Implicit purpose		
	P (%)	R (%)	F1 (%)
SwiRL	97.78	47.57	64.00
SwiRL_DA	99.12	60.54	75.17

TABLE 6: The overall effect of PurExt.

	P (%)	R (%)	F1 (%)
Actor	92.45	85.47	88.82
Action	96.18	95.06	95.61
Data	97.90	89.91	93.74
Purpose	99.53	69.08	81.55
Rule	97.07	86.35	91.39

TABLE 7: Effect of other elements extraction.

	Actor			Action			Data		
	P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)
PolicyLint	74.86	77.91	76.35	90.50	69.53	78.64	95.04	57.50	71.65
PurExt	92.45	85.47	88.82	94.32	92.70	93.51	97.48	88.71	92.89

5.5.2. *Experiment Results.* Table 7 demonstrates the performance of PolicyLint and PurExt to extract three elements. As demonstrated, PurExt performs better than PolicyLint in all aspects. On the one hand, the advantage of PurExt is that it benefits from the combination of syntactic and semantic analyses compared to PolicyLint, which only uses the former. On the other hand, PolicyLint uses 82 templates that were learned from 560 example sentences to match the sentences to be extracted; thus, it passes over the sentences that do not follow the patterns.

6. Conclusions

Because IoT devices collect a considerable amount of personal and sensitive information, the privacy issues for IoT are a major concern for the users and laws. Privacy policy is an important way for IoT vendors to obtain users' trust and to adhere to legal requirements. Therefore, how to make the privacy policy better serve users and regulatory compliance has aroused our interest. This study explores the expression and automated extraction of the purpose-centric data usage

purposes in privacy policies. More precisely, we propose a purpose-aware rule to formalize the data access statements and combine syntactic and semantic analyses to realize the automated extraction of rules from the natural language privacy policies. To the best of our knowledge, this is the first attempt to extract the structural purpose-centric statement from privacy policies. The experimental results on real datasets show that this approach can achieve a 91% recall rate and 97% precision.

Because PurExt performs sentence-level extraction, a separate description of the data objects and purposes in different sentences will lead to incomplete rules. When considering the future directions of research, the entire document should be analyzed to obtain more context. In terms of the promotion, we will proceed to apply the extracted purpose-aware rules for the research of compliance verification.

Data Availability

The data used to support the findings of this study are available from the first author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (nos. U19A2081 and 61802270).

References

- [1] C. B. Foltz and L. Foltz, "Mobile users' information privacy concerns instrument and IoT," *Information & Computer Security*, vol. 28, no. 3, pp. 359–371, 2020.
- [2] E. Zeng and F. Roesner, "Understanding and improving security and privacy in multi-user smart homes: a design exploration and in-home user study," in *Proceedings of the 28th USENIX Security Symposium (USENIX Security 19)*, pp. 159–176, Berkeley, CA, USA, May 2019.
- [3] S. Zheng, N. Apthorpe, M. Chetty, and N. Feamster, "User perceptions of smart home IoT privacy," *Proceedings of the ACM on Human-Computer Interaction*, vol. 2, pp. 1–20, 2018.
- [4] T. Pasquier, J. Singh, J. Powles, D. Eysers, M. Seltzer, and J. Bacon, "Data provenance to audit compliance with privacy policy in the Internet of Things," *Personal and Ubiquitous Computing*, vol. 22, no. 2, pp. 333–344, 2018.
- [5] S. Wilson, F. Schaub, A. A. Dara et al., "The creation and analysis of a website privacy policy corpus," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pp. 1330–1340, Berlin, Germany, August 2016.
- [6] J. Mohan, M. Wasserman, and V. Chidambaram, "Analyzing gdpr compliance through the lens of privacy policy," in *Heterogeneous Data Management, Polystores, and Analytics for Healthc*, pp. 82–95, Springer, Berlin, Germany, 2019.
- [7] W. B. Tesfay, P. Hofmann, T. Nakamura, S. Kiyomoto, and J. Serna, "I read but don't agree: privacy policy benchmarking using machine learning and the EU GDPR," in *Proceedings of the Web Conference 2018 - Companion of the World Wide Web Conference, WWW 2018, Association for Computing Machinery, Inc*, pp. 163–166, Lyon, France, April 2018.
- [8] G. Kapitsaki, J. Ioannou, J. Cardoso, and C. Pedrinaci, "Linked USDL privacy: describing privacy policies for services," in *Proceedings of the 2018 IEEE International Conference on Web Services (ICWS)*, pp. 50–57, IEEE, Seattle, WA, USA, June 2018.
- [9] R. N. Zaeem, R. L. German, and K. S. Barber, "Privacycheck: automatic summarization of privacy policies using data mining," *ACM Transactions on Internet Technology*, vol. 18, no. 4, pp. 1–18, 2018.
- [10] D. A. Audich, R. Dara, and B. Nonnecke, "Privacy policy annotation for semi-automated analysis: a cost-effective approach," in *IFIP Advances in Information and Communication Technology*, pp. 29–44, Springer, Berlin, Germany, 2018.
- [11] J. Bhatia and T. D. Breaux, "Semantic incompleteness in privacy policy goals," in *Proceedings of the 2018 IEEE 26th International Requirements Engineering Conference (RE)*, pp. 159–169, IEEE, Banff, AB, Canada, August 2018.
- [12] N. M. Nejad, P. Jabat, R. Nedelchev, S. Scerri, and D. Graux, "Establishing a strong baseline for privacy policy classification," in *IFIP International Conference on ICT Systems Security and Privacy Protection*, pp. 370–383, Springer, Berlin, Germany, 2020.
- [13] S. Wilson, F. Schaub, R. Ramanath et al., "Crowdsourcing annotations for websites' privacy policies: can it really work?" in *Proceedings of the 25th International Conference on World Wide Web*, pp. 133–143, Montreal, Canada, April 2016.
- [14] B. Andow, S. Y. Mahmud, W. Wang, J. Whitaker, W. Enck, B. Reaves et al., "Policylint: investigating internal privacy policy contradictions on google play," in *Proceedings of the 28th USENIX Security Symposium (USENIX Security 19)*, pp. 585–602, Berkeley, CA, USA, August 2019.
- [15] L. Yu, X. Luo, C. Qian, S. Wang, and H. K. Leung, "Enhancing the description-to-behavior fidelity in android apps with privacy policy," *IEEE Transactions on Software Engineering*, vol. 44, no. 9, pp. 834–854, 2017.
- [16] M. C. Tschantz, A. Datta, and J. M. Wing, "Purpose restrictions on information use," in *Lecture Notes in Computer Science*, pp. 610–627, Springer, Berlin, Germany, 2013.
- [17] D. Basin, S. Debois, and T. Hildebrandt, "On purpose and by necessity: compliance under the GDPR," in *Financial Cryptography and Data Security*, pp. 20–37, Springer, Berlin, Germany, 2018.
- [18] P. Shayegh, V. Jain, A. Rabinia, and S. Ghanavati, "Automated approach to improve iot privacy policies," 2019, <http://arxiv.org/abs/1910.04133>.
- [19] E. Onu, M. M. Kwakye, and K. Barker, "Contextual privacy policy modeling in iot," in *Proceedings of the 2020 IEEE International Conference on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing*, pp. 94–102, Calgary, AB, Canada, August 2020.
- [20] H. Harkous, K. Fawaz, R. Leuret, F. Schaub, K. G. Shin, and K. Aberer, "Polisis: automated analysis and presentation of privacy policies using deep learning," in *Proceedings of the 27th USENIX Security Symposium (USENIX Security 18)*, pp. 531–548, Berkeley, CA, USA, August 2018.
- [21] A. Subahi and G. Theodorakopoulos, "Ensuring compliance of IoT devices with their privacy policy agreement," in *Proceedings of the 2018 IEEE 6th International Conference on Future Internet of Things and Cloud (FiCloud)*, pp. 100–107, Barcelona, Spain, August 2018.

- [22] S. Zimmeck, Z. Wang, L. Zou et al., “Automated analysis of privacy requirements for mobile apps,” in *Proceedings of the NDSS*, San Diego, CA, USA, May 2017.
- [23] B. Andow, S. Y. Mahmud, J. Whitaker, W. Enck et al., “Actions speak louder than words: entity-sensitive privacy policy and data flow analysis with polichex,” in *Proceedings of the 29th USENIX Security Symposium (USENIX Security 20)*, pp. 985–1002, Anaheim, CA, USA, August 2020.
- [24] P. Story, S. Zimmeck, A. Ravichander et al., “Natural language processing for mobile app privacy compliance,” in *Proceedings of the AAAI Spring Symposium on Privacy-Enhancing Artificial Intelligence and Language Technologies*, Palo Alto, CA, USA, March 2019.
- [25] S. Zimmeck, P. Story, D. Smullen et al., “Maps: scaling privacy compliance analysis to a million apps,” *Proceedings on Privacy Enhancing Technologies*, vol. 2019, no. 3, pp. 66–86, 2019.
- [26] J. Bhatia, M. C. Evans, and T. D. Breau, “Identifying incompleteness in privacy policy goals using semantic frames,” *Requirements Engineering*, vol. 24, no. 3, pp. 291–313, 2019.
- [27] X. Xiao, A. Paradkar, S. Thummalapenta, and T. Xie, “Automated extraction of security policies from natural-language software documents,” in *Proceedings of the ACM SIGSOFT 20th International Symposium on the Foundations of Software Engineering*, pp. 1–11, Cary, NC, USA, November 2012.
- [28] J. Slankas, X. Xiao, L. Williams, and T. Xie, “Relation extraction for inferring access control rules from natural language artifacts,” in *Proceedings of the 30th Annual Computer Security Applications Conference*, pp. 366–375, New York, NY, USA, December 2014.
- [29] J. Slankas and L. Williams, “Access control policy extraction from unconstrained natural language text,” in *Proceedings of the 2013 International Conference on Social Computing*, pp. 435–440, IEEE, Alexandria, VA, USA, September 2013.
- [30] M. Narouei, H. Khanpour, and H. Takabi, “Identification of access control policy sentences from natural language policy documents,” in *Data and Applications Security and Privacy XXXI*, pp. 82–100, Springer, Berlin, Germany, 2017.
- [31] M. Narouei, H. Takabi, and R. Nielsen, “Automatic extraction of access control policies from natural language documents,” *IEEE Transactions on Dependable and Secure Computing*, vol. 17, no. 3, pp. 506–517, 2018.
- [32] V. Punyakanok, D. Roth, and W.-t. Yih, “The importance of syntactic parsing and inference in semantic role labeling,” *Computational Linguistics*, vol. 34, no. 2, pp. 257–287, 2008.
- [33] M. Lewis, L. He, and L. Zettlemoyer, “Joint a* ccg parsing and semantic role labelling,” in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 1444–1454.
- [34] A. Björkelund, L. Hafdel, and P. Nugues, “Multilingual semantic role labeling,” in *Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL 2009): Shared Task*, pp. 43–48, Boulder, CO, USA, June 2009.
- [35] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, “Natural language processing (almost) from scratch,” *Journal of Machine Learning Research*, vol. 12, pp. 2493–2537, 2011.
- [36] M. Surdeanu and J. Turmo, “Semantic role labeling using complete syntactic analysis,” in *Proceedings of the Ninth Conference on Computational Natural Language Learning (CoNLL-2005)*, pp. 221–224, 2005.
- [37] A. I. Antón, E. Bertino, N. Li, and T. Yu, “A roadmap for comprehensive online privacy policy management,” *Communications of the ACM*, vol. 50, no. 7, pp. 109–116, 2007.
- [38] X. Yang and J. Alves-Foss, “Security policy refinement: high-level specification to low-level implementation,” in *Proceedings of the 2013 International Conference on Social Computing*, pp. 502–511, IEEE, Washington, DC, USA, September 2013.
- [39] M. Honnibal and I. Montani, “Spacy 2: Natural language understanding with bloom embeddings,” *Convolutional Neural Networks and Incremental Parsing*, vol. 7, no. 1, 2017.
- [40] X. Carreras and L. Màrquez, “Introduction to the CoNLL-2005 shared task: semantic role labeling,” in *Proceedings of the ninth conference on computational natural language learning (CoNLL-2005)*, pp. 152–164, Ann Arbor, MI, USA, June 2005.
- [41] Github Htmltoplaintext. <https://github.com/benandow/HtmlToPlainText>.
- [42] M. Sokolova and G. Lapalme, “A systematic analysis of performance measures for classification tasks,” *Information Processing & Management*, vol. 45, no. 4, pp. 427–437, 2009.

Research Article

An Adaptive Protection of Flooding Attacks Model for Complex Network Environments

**Bashar Ahmad Khalaf,^{1,2} Salama A. Mostafa¹,¹ Aida Mustapha,¹
Mazin Abed Mohammed³,³ Moamin A. Mahmoud,⁴ Bander Ali Saleh Al-Rimy,⁵
Shukor Abd Razak⁵,⁵ Mohamed Elhoseny,⁶ and Adam Marks⁷**

¹Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia, Johor 86400, Malaysia

²Bilad Alrafidain University College, Ba'aqubah 32001, Diyala, Iraq

³College of Computer Science and Information Technology, University of Anbar, Ramadi 31001, Iraq

⁴College of Computer Science and Informatics, Universiti Tenaga Nasional, Kajang, Selangor 43000, Malaysia

⁵Faculty of Engineering, Universiti Teknologi Malaysia, Johor 81310, Malaysia

⁶Department of Computer Science, College of Computer Information Technology, American University in the Emirates, Dubai 503000, UAE

⁷Zayed University, Dubai, UAE

Correspondence should be addressed to Salama A. Mostafa; salama@uthm.edu.my

Received 11 February 2021; Revised 29 March 2021; Accepted 13 April 2021; Published 23 April 2021

Academic Editor: Chalee Vorakulpipat

Copyright © 2021 Bashar Ahmad Khalaf et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Currently, online organizational resources and assets are potential targets of several types of attack, the most common being flooding attacks. We consider the Distributed Denial of Service (DDoS) as the most dangerous type of flooding attack that could target those resources. The DDoS attack consumes network available resources such as bandwidth, processing power, and memory, thereby limiting or withholding accessibility to users. The Flash Crowd (FC) is quite similar to the DDoS attack whereby many legitimate users concurrently access a particular service, the number of which results in the denial of service. Researchers have proposed many different models to eliminate the risk of DDoS attacks, but only few efforts have been made to differentiate it from FC flooding as FC flooding also causes the denial of service and usually misleads the detection of the DDoS attacks. In this paper, an adaptive agent-based model, known as an Adaptive Protection of Flooding Attacks (APFA) model, is proposed to protect the Network Application Layer (NAL) against DDoS flooding attacks and FC flooding traffics. The APFA model, with the aid of an adaptive analyst agent, distinguishes between DDoS and FC abnormal traffics. It then separates DDoS botnet from Demons and Zombies to apply suitable attack handling methodology. There are three parameters on which the agent relies, normal traffic intensity, traffic attack behavior, and IP address history log, to decide on the operation of two traffic filters. We test and evaluate the APFA model via a simulation system using CIDDs as a standard dataset. The model successfully adapts to the simulated attack scenarios' changes and determines 303,024 request conditions for the tested 135,583 IP addresses. It achieves an accuracy of 0.9964, a precision of 0.9962, and a sensitivity of 0.9996, and outperforms three tested similar models. In addition, the APFA model contributes to identifying and handling the actual trigger of DDoS attack and differentiates it from FC flooding, which is rarely implemented in one model.

1. Introduction

A Distributed Denial of Service attack (DDoS) is the most common type of flooding attack, which floods computer networks. Complex network environments consist of a

variety of servers, including web, Internet of Things (IoT), cloud, fog, etc., that are exposed to huge requests that slow down networks and interrupt services [1]. These attacks occur for different reasons such as financial, personal, political, ransom, and cyberwar at different security levels and

cause various attack impacts [2]. Accordingly, DDoS attacks affected nearly 2,500 organizations with 75,000 computer systems and over 100 countries with four million computers in 2010 and 2011 [3]. In the first quarter of 2016, a 602 Gbps DDoS attack was launched against the BBC website and crashed the website for several hours [4]. Basically, before the attack, the attackers (known as Demons) hack personal computer users who access the web and take over these computers. Subsequently, attackers exploit these computers by planting harmful codes or other strategies to gain control of the computers [5]. The number of these hacked computers (known as Zombies) can reach into thousands. Such a number of Zombies' creates a "botnet," which is a network of private computers that has been planted with malicious software and manipulated as a group without the owners' knowledge, e.g., to send spam. The severity of attacks depends on the size and scale of a botnet. A bigger botnet is usually associated with increasingly severe and catastrophic attacks.

There are two main types of DDoS attacks. The first type targets the Network Application Layer (NAL) such as HTTP flood, DNS flood, and FTP [6, 7]. In this type, the attacker issues vindictive or noxious bundles/packets aimed at the unfortunate casualty to cause disarray concerning the convention or any application that keeps running on it (e.g., vulnerability or defencelessness attack) [5]. The second type targets the Network and Transport layers such as UDP flood, TCP flood, and ICMP flood [8]. In all of these attacks, the attacker targets to (i) exhaust system assets, transfer speed, or the handling limit of switching to upset the network of an authentic client and (ii) exhaust the servers' assets such as memory, CPU, I/O, transmission capacity, and HDD/database transmission capacity to interfere with the administrations of legitimate clients. This study focuses on attacks targeting the Hypertext Transfer Protocol (HTTP) of the NAL.

A kind of abnormal network traffic is the Flash Crowd (FC) that causes a refusal of administration for an Internet administration's real clients [9]. The FC closely resembles the DDoS attack, whereby enormous legitimate clients simultaneously access a specific processing asset (e.g., a website). For instance, important news created worldwide, the distribution of the Olympic timetable, or organizations like Apple, Sony, and Samsung initiating a novel item brings about an unexpected flood in authentic traffic [10]. These outcomes of the ill-timed and undue conveyance of reactions by the web administration require prompt action. As DDoS attacks and FC traffic contrast in only a couple of metrics, distinguishing them is a major hurdle [11]. Researchers have suggested and actualized various cybersecurity models to defend network systems and applications from DDoS and FC attacks. However, the harmful streams disguised in authentic traffic are a scourge for these security prototypes. Many of these models cannot distinguish between real and pernicious streams with respect to negatives generations and false-positives.

An agent is a programming component or an integration of programming and equipment entities that can be executed in parallel in its clients' interest. It includes numerous

helpful functions, such as learning capability, cooperation, responsiveness, and effectiveness [3]. The agent is deployed in this area either in the attacker or defense teams [12, 13]. For instance, in Kotenko et al. [14], an agent or agents are employed with an assailant system to produce and control a vast number of deceitful DDoS botnet traffic. The agent is used to oversee or handle versatile decision-making forms in the protection against DDoS attacks. The enormous hurdle in creating and strengthening the defense components of DDoS is to distinguish between the DDoS attack and an FC, in which a real action may oftentimes show up as malevolent. Cybersecurity research that focuses on distinguishing between DDoS and FC attacks has progressed over the years. Various artificial intelligence methods, such as fuzzy logic, genetic algorithm, K-Nearest Neighbor (K-NN) calculation, Bayesian networks, neural networks, software agent technology, and Support Vector Machines (SVM), are discussed in the literature.

We are inspired to design an agent-based defense model that has the ability to protect against DDoS and FC targeting the NAL. We consolidate the agent with the protective or defensive archetype. We assume that it is important to develop an effective method that detects DDoS attacks and expunge malicious traffics at the application layer level before they cause harm to the web servers and applications. We propose an Adaptive Protection of Flooding Attacks (APFA) model to protect the NAL against DDoS and FC. Four modules form the APFA model: (i) Abnormal Traffic Detection Module (ATDM), (ii) DDoS Attack Detection Module (DADM), (iii) Adaptive Traffic Control Module (ATCM), and (iv) Kalman and Bloom Filters Module (KBFM). The ATCM represents our main contribution, which integrates an adaptive agent with the belief-desire-intention (BDI) architecture to identify, classify, and control traffics of network systems. The test results of the APFA model show that the adaptive agent does not just give an upper hand by enhancing procedure value or capacity but coordinates the process of the innovative modules and improves the overall performance of the simulated network system.

We organize this paper into six sections having the first section as an introduction to the research work. In Section 2, we review the related work. Section 3 presents the research methods and materials. Section 4 illustrates the main components of the APFA model. Section 4 describes the simulation environment and testing platforms. In Section 6, we discuss the results and review the contributions and limitations of this work. Finally, Section 7 presents the conclusion and highlights a key point for future work.

2. Related Work

Several well-established studies have focused on the defense against DDoS attacks and control FC traffics that targeted the NAL. In this section, we review the details of the most effective and related well-established works that have been presented and discussed in the literature.

Shiales et al. [15] accomplished a DDoS attack recognition with enhanced time constraints using a

nonasymptotic fuzzy evaluator. The evaluator is implemented on average packet inter-arrival durations. The complication is divided into two units: recognition of the actual DDoS attack and identification of the IP addresses of the victims. The former task is accomplished by employing stringent, real-time boundaries for DDoS attack discovery. The latter goal is achieved using comparatively lenient constraints, which identify the IP addresses of the victims promptly, thereby starting embedded anti-attack functions on the affected hosts employing the arriving time of the packet as the primary statistic of DDoS attack detection.

Kaur et al. [16] use the “survival of the fittest” principle in which when many clients try to get scarce assets; the stronger clients overcome the weaker ones. Consequently, to replace clients with low fitness, a chain of repetitions or successive approximations is implemented using a fitness or suitability function. In this instance, GAs could be used with information captured from inbound streams of packets and in selecting optimum metrics to detect and distinguish attacks from normal packets. Katkar et al. [17] recommend using a network intrusion detection system model that uses signatures to identify DDoS attacks on HTTP servers by using shared handling and a naive Bayesian classifier. They use observational outcomes to validate the efficiency of the model. The naive Bayes classifies attacks that are slow and have 97.82% precision, and regular behavior is detected with a precision of 96.46%.

Barrionuevo et al. [18] propose an approach and an analysis of its practicability on three known attacks of service denial: Fraggle, Land, and Smurf. They solve the execution problem using the HPC techniques in the GPU to quicken the procedure and produce the outcomes. They evaluate the approach via several indices. The proposed approach achieves 40% to 70% accuracy and 60% to 83% sensitivity. The F-measure, which is employed to estimate the framework’s execution, is 0.5 to 0.83. Sreeram and Vuppala [19] propose a Bio-Inspired Anomaly-based application layer DDoS attack (App-DDoS Attack) to defend against DDoS attack by using the CIDDs dataset. Furthermore, the proposed model aims to achieve fast and early detection. As shown in the results, the proposed model achieves an excellent result in defending against DDoS attacks with 99.64% accuracy. However, the proposed model lacks the ability to deal with the legitimate traffics that stream with pernicious DDoS traffics, but it has the ability to detect only limited types of flooding attacks.

A multilevel DDoS mitigation framework (MLDMF) is recommended for all levels of the IoT systems architecture [20] that is built upon the edge-, fog-, and cloud-computing levels. IoT gateways are utilized at the edge-computing level to manage and secure IoT nodes based on the SDN. An IoT management control unit (IMCU) is employed at the fog-computing level, which consists of SDN controllers and software to detect and neutralize DDoS attacks. On the other hand, the cloud-computing level analyzes the network traffics using big data and AI to protect against DDoS attacks by establishing an intelligent attack identification and mitigation structure. The simulation outcomes of the three computing level architecture of the IoT show that the edge-computing level’s quick response capability, fog-computing

level’s state recognition feature, cloud-computing level’s computing capability, and SDN’s network programmability could solve the DDoS problem in IoT.

Verma and Ranga [21] present the measurable examination of the marked stream-dependent CIDDs dataset utilizing K-NN grouping and K-Means bundling calculation. Some noticeable assessment parameters are utilized to assess IDS, including accuracy, recognition rate, and false-positive rate. In another work of Verma and Ranga [22], they lead an itemized investigation of the CIDDs dataset and report the discoveries. They utilize a wide scope of familiar AI procedures to examine the multifaceted nature of the dataset. The assessment measurements that they use include recognition rate, precision, false-positive rate, kappa insights, and root mean squared deviation to appraise implemented AI approaches.

Mohamed et al. [23] come up with an identification framework of HTTP DDoS attacks in a Cloud domain that depends on Information-Theoretic Entropy and Random Forest collection learning calculation. They utilize a time-sensitive sliding window calculation to appraise the measure of randomness of the network header attributes of the approaching system traffic. At the point when the evaluated entropy surpasses its typical range, the preprocessing and the characterization exercises are activated. To evaluate the suggested methodology, they carry out different tests on the CIDDs-001 open dataset. The recommended methodology accomplishes acceptable outcomes with a precision of 99.54% and FPR of 0.46%. Moreover, the framework has been proposed to protect the cloud environment against DDoS attacks. However, the proposed framework is inefficient in handling FC, and it can only detect limited types of flooding attacks.

An agent-based methodology and programming condition (which is based on the OMNeT++ INET framework) is designed by Kottenko et al. [24] to model shared protection techniques for installation on the web to neutralize network attacks. This method is characterized by various agent groups that collaborate to neutralize malicious traffics and as a protection mechanism against attacks. Similarly, Juneja et al. [25] suggest a multi-agent architecture to identify, protect, and track the origin of a DDoS attack. While this approach is able to locate the source of a DDoS attack, a number of agents are needed to produce the best results.

Kesavamoorthy and Soundar [26] develop a technique, which uses a self-contained multi-agent system for detecting and protecting against DDoS attacks. In this technique, agents use particle swarm enhancement/optimization to attain an excellent correspondence or interaction. DDoS attacks are recognized when many connected agents are deployed to communicate new attacks to the coordinator agent. The cloud-based system protects against many types of DDoS attacks with an accuracy of 98%. A multi-agent-based distribution system identifies and prevents DDoS attacks within the ISP boundaries and is presented in the work of Singh et al. [27]. The agents and their coordinating partners implement the task of preventing the attacks in all ISPs. These agents work together by checking the incoming traffics on the edge router and using an entropy threshold-

based technique to detect the existence of DDoS attacks. If an attack occurs, the coordinator agent communicates this information with the neighboring ISPs to create a distributed protection environment. The authors adopt certain metrics to assess the performance of the defense system. However, the system's efficiency is evaluated against the system's performance in the absence of suitable metrics.

Lin et al. [28] suggest two versatile sampling calculations to gather security-associated information using agent technology. The agent has adaptive mechanisms to enhance acquisition productivity, guarantee to gather precision, and reduce the measure of gathered information. The aim of these mechanisms is to limit the impact of information capturing on the regular activities of a network. The outcomes demonstrate the benefits of the versatile security-associated information gatherer with respect to the productivity and flexibility of adaptive agents.

Generally, we can ascribe a DDoS attack as a scalable network security issue. While researchers have developed many detection and defensive mechanisms against DDoS attacks, success has been limited in implementing the mechanisms across a range of computing networks. The use of the artificial intelligence approach is limited to identifying whether clients' requests are valid or malicious based on the requests' attributes. However, the above discussions clearly enlighten the software agent's suitability as a technology that could be used in our proposed model to make the system more flexible and adaptable in dealing with the various cases of DDoS and FC targeting network traffics.

3. Materials and Methods

This section discusses the research materials and methods of this work, starting with a review of the adaptive agent architecture and mechanisms related to this work, followed by a description of the CIDDs testing dataset and its attributes. Subsequently, we explain the threats model design and the evaluation methods.

3.1. Adaptive Agent. An agent is a mix of equipment or programming elements that is responsive, for the benefit of its clients, in an autonomous manner. It has numerous helpful attributes like adaptivity, autonomy, connectivity, learning, reactivity, and proactivity. An adaptive agent provides applicability in vast domains, for example, portable processing, data recovery and processing, smart communication, media communications, and electronic commerce [29]. These agents interact in a multi-agent framework and are directed in different manners to serve particular clients or perform specific tasks. The qualities that spurred the utilization of the agent technology in this work include its self-governance, adaptation to failures, dynamic setup, autonomous decisions, situatedness, and scalability [25]. The agent may now and again endeavor to adjust to be more adept to its new or dynamic condition or to manage new or evolving objectives [30]. Contemplations of agent alteration or acclimatization incorporate what calculation can be utilized to alter the agent behavior? What is the utmost

measure of progress anticipated in the agent framework? How is the framework going to stop development from going beyond control? And how to recognize and manage an alteration whose impact is not ideal? Versatile identification is the learning capacity to recognize any alteration in chance markings or configurations in an environment or system to be more adept to its condition [31]. Figure 1 demonstrates the deployment of adaptive mechanisms in agents based on the agent's dynamics and the related system.

The motivation behind the adaptation behavior can be a response to changes, evaluation of situations, or dealing with uncertainty. Adaptive procedures can be time-differing when receptive or responsive to a disturbance with a continuous interior shift of the choice procedure through repeated choice, successive choice, or audited rules [29, 32]. The reactive or responsive adaptive type is considered the most effective in this domain because it portrays the limit of the protection prototype (e.g., time) to respond against the DDoS attack. For instance, Cheng et al. [33] propose a DDoS attack recognition model that utilizes responsive adaptation in an agent to recognize and control attack streams. The agent utilizes the responsive adaptation to screen the conduct of approaching streams of information and afterward control the traffic movement.

3.2. The Testing Dataset and Parameters. Coburg Intrusion Detection Data Sets (CIDDs) is a marked stream-dependent dataset [6, 34]. It is created essentially for the assessment of IDS and IPS. The dataset comprises OpenStack and External Servers traffics. We ignore Attack ID and Attack Description's features in this study because they just offer extra insights into the executed attacks without significantly contributing to the analysis. We collect about 153,026 occurrences from the outer servers and 172,839 occurrences from the OpenStack Server information for examination. The dataset classes' occurrences are labeled or marked as expected, assailant, unfortunate casualty, suspicious, and obscure classes. Table 1 gives a representation of CIDDs dataset features.

Basically, the CIDDs dataset is chosen because it is the most recent dataset, produced in 2017; available online for free; and can simulate real-time processing due to its duration attribute. It also has the attributes of both DDoS attack and FC flooding traffics and the other existing datasets such as KDD, DARPA, and CAIDA, which lack the above attributes. Many methods have been used in defending against DDoS and FC. Each one of them used specific parameters that are suitable for the simulated systems. Table 2 presents the used parameters in building the simulated study of this work.

3.3. The Threats Model Design. This study is mostly involved with three sorts of flooding attacks or attacks, in which each is more clandestine than the previous one. (i) The assailants put forth countless HTTP solicitations to expand the framework asset and make the framework useless for the legitimate-client, which we refer to as the DDoS targeting application layer [2]. (ii) The assailants assume responsibility

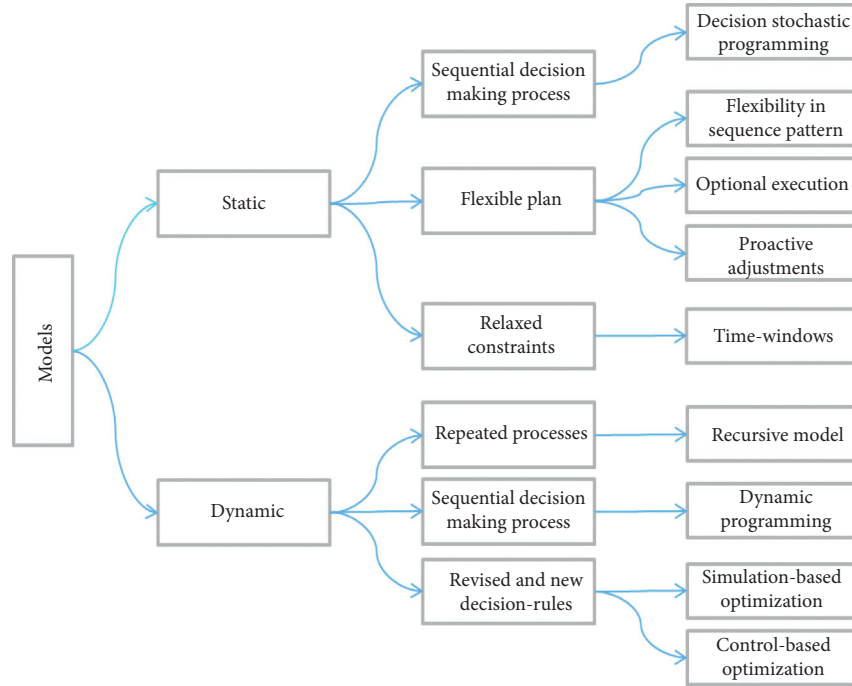


FIGURE 1: The models of adaptive agent.

TABLE 1: The CIDDs dataset attributes [34].

No.	Feature name	Feature description
1	Src IP	IP address of the source node
2	Src port	Port of the source node
3	Dest IP	IP address of the destination node
4	Dest port	Port of the destination node
5	Proto	Protocol
6	Data are first seen	Start time flow is first seen
7	Duration	Flow period
8	Bytes	Conveyed bytes
9	Packets	Conveyed packets
10	Flags	TCP flags
11	Attack description	Additional information about the attack
12	Attack type	Type of attack
13	Attack ID	Unique attack ID
14	Class	Category or label of the instance

TABLE 2: The testing parameters.

No.	Abbreviation	Parameters	Value
1	Window size	The size of dataset segmentation	7
2	Period	The duration of the dataset	7
3	SSM	Special sequence matrix	130
4	MCP	Model-checking period	24
5	MST	Model similarity threshold	Dynamic
6	P0	Normal traffic intensity	—
7	P1	Current traffic intensity	—
8	P2	Traffic behavior	—
9	P3	IP history log	—

for some PC machines through the web, leaving these PCs in a defenseless and helpless situation [5]. The assailants at that point begin misusing the shortcomings of these PCs by planting noxious codes or other hacking procedures to deal

with the machines; they are called “Zombies.” It is very simple to accomplish and certainly difficult to detect because the irregular traffic is utilized into a gathering of targets and behaves increasingly like an authentic visiting. (iii) A kind of system traffic is FC that could initiate a stop of administration for an Internet administration’s legitimate-clients. The FC is closely similar to the DDoS attack, in which a specific figure of traffic requests legitimate service. For example, a site is accessed by a huge number of legitimate-clients at the same time. Breaking news produced far and wide, for example, the distribution of the Olympic calendar or organizations like Apple, Samsung, and so on, launching another product brings about an unexpected flood in a legitimate-increase in legitimate-traffics [11]. All those types of DDoS attacks are generated from the CIDDs dataset because it has the required attributes and attack scenarios.

3.4. Evaluation Metrics. In this analytical study, our system’s performance is evaluated using eminent metrics, such as accuracy, precision, and sensitivity. Those measurements are assessed from the components of the confusion matrix. True-Positive (TP), True-Negative (TN), False-Positive (FP), and False-Negative (FN) are the components of a confusion matrix, where TN is the number of actual nonoccurrences of an attack. TP is the number of actual occurrences of an attack. FP is the number of inaccurately identified attack occurrences. Thus, FN is the number of inaccurately identified nonoccurrences as attack cases. Accuracy or exactness is characterized as the proportion of all effectively delegated occurrences (TP, TN) to every one of the cases (TP, TN, FP, and FN). Precision or preciseness (positive predictive quality) is the proportion of TP to a sum of TP and FP.

Sensitivity is the proportion of TP to a sum of TP and FN. Accuracy is calculated using

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}. \quad (1)$$

While precision is calculated using

$$Precision = \frac{TP}{TP + FP}. \quad (2)$$

and, sensitivity is calculated using

$$Sensitivity = \frac{TP}{TP + FN}. \quad (3)$$

4. The Adaptive Protection of Flooding Attacks Model

This work proposes the Adaptive Protection of Flooding Attacks (APFA) model, an engineered or structural expansion to protect web applications and servers against DDoS and FC attacks. It is targeted at huge-scale online organizations, including nonbusiness entryway websites. The APFA consists of four accompanying units or modules: Abnormal Traffic Detection Module (ATDM), DDoS Attack Detection Module (DADM), Adaptive Traffic Control Module (ATCM), and the Kalman Bloom Filters Module (KBFM), as shown in Figure 2. The role of each module is described in the following subsections. The base work of the model is the AL-DDoS model, which is taken from [35, 36]. Therefore, some of the model's basic parts are not detailed in this paper.

4.1. The Abnormal Traffic Detection Module. The Abnormal Traffic Detection Module (ATDM) is the first part of the APFA model. This module's major aim is to monitor and analyze the traffic to detect sudden changes in HTTP GET requests. It does not take any action if no anomalies are detected in the traffic. If it detects abnormal traffic from the incoming HTTP traffic, an "attention" signal is sent to the next module, which is the DADM, for further analyses, as shown in Figure 2. Several steps are taken before sending an attention signal starting with the measurement of the incoming traffic. This can be done in many different ways, but the APFA model measures traffic intensity by using an Auto Regression (AR) mechanism [35]. In regression, previous values affect future values. Therefore, the AR mechanism uses previously observed traffic to predict the change of traffic intensity in the future. Initially, the HTTP GET traffic stream is monitored. A time-series $\{y_1, y_2, \dots, y_t\}$ is formed by the traffic intensity, which is studied in constant time intervals. The traffic intensity is calculated by the total number of packages received in a time interval [36]. If major changes are detected, it can potentially be a DDoS attack. The AR predicts the current traffic intensity by using

$$y_t = \sum_{k=1}^p a_t^k x_{t-k} + e_t. \quad (4)$$

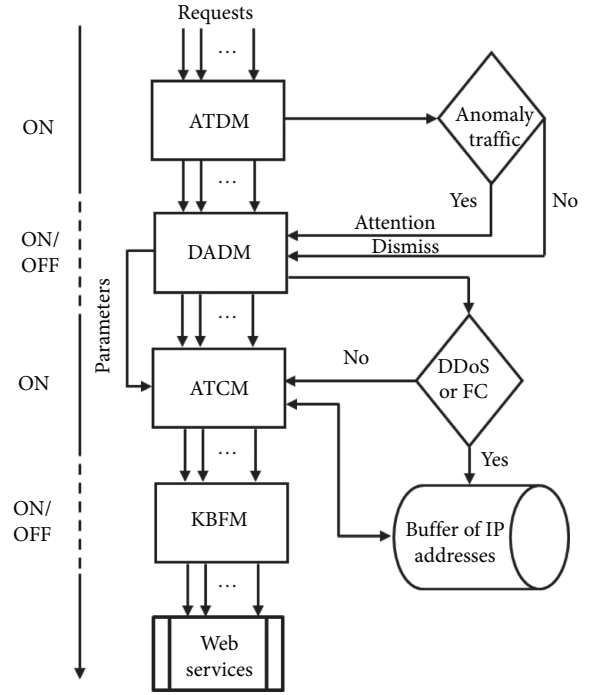


FIGURE 2: The APFA model.

The variable y_t predicts x_t , which is the observed value at time t . The variable a_t^k is a "constant model parameter," which means that it remains constant with time, and e_t is the observed error [36]. Secondly, at a certain time t , the difference between the observed x_t and the predicted y_t gives the residual error x_t [35].

$$d_t = |y_t - x_t|. \quad (5)$$

From the residual error at time t , a standard deviation, σ_d^2 , is calculated:

$$\sigma_d^2 = \frac{\left(\sum_{i=t-p}^t (d_t - \text{avg}(d_{t-p}^t))^2\right)}{p} \quad (6)$$

Subsequently, a threshold is calculated as in equation (7), which determines abnormal traffic. If d_t is greater than $k\sigma_d^2$, then, abnormal traffic is detected, and an attention signal is sent to the DDoS attack detection module. Otherwise, no abnormal traffic is detected, and the ATDM sends a "dismiss" signal to DADM, which inactivates itself, as shown in Figure 2. The constant k adjusts the sensitivity of the threshold and is set to a specific value.

$$d_t > k\sigma_d^2. \quad (7)$$

4.2. The DDOS Attack Detection Module. The DDOS Attack Detection Module (DADM) is the second part of the APFA model. It uses a trading strategy for dependably deciding a packet's source on the web. This strategy is well-known in many DDoS protection models to distinguish the legitimate origins of attacking packets existing in a network server [37]. It

contains an adaptable stream-dependent labeling plan that uses the attendant switch's load to alter stamps or labels [24]. Based on this strategy, the DADM uses Special Sequence Matrix (SSM), which denotes zero as a normal request and one as an anomaly request to give notable attributes for origin tracing the IP bundles to furnish better tracing ability [19]. Appraisals of embedded overload avoidance instruments enable this module to provide an appropriate trace-back outcome, notwithstanding when there is a substantial burden on the server. Aside from tracking DDoS attacking packets, DADM assists in enhancing the filtering or sifting of attacking traffic.

At the point when attention signals are sent from the ATDM, the DADM starts tracing the source of each IP address that sends the anomaly traffic. It then measures the mean occurrences of the associated Real-time Frequency Vector (RFV) of the traffic. The RFV holds the variation range of daily traffic for a particular server. In enormous traffics, the mean occurrence of the RFVs can be seen as the likelihood of every needed website page. Indeed, it is essential to find the value of RFV for significant traffic to deliver the progress of traffic occurrences. For the traffic model, M_1 , we register RFVs possibilities:

$$p(v_i) = \frac{\sum_{i=1}^{|V|} S_{ij}}{\sum_{i=1}^{|V|} S_{ij} \sum_{j=1}^{|V|} S_{ij}}. \quad (8)$$

For a subsequent traffic model, M_2 , we can determine their support values using equation (9),

$$P(v_i \longrightarrow v_j | v_i) = \frac{S_{ij}}{\sum_{i=1}^{|V|} S_{ij}}. \quad (9)$$

The certainty of the M_1 according to M_2 is obtained using equation (10). This indicates the likelihood of the upcoming traffic models, M_1, M_2, \dots from $v_i \longrightarrow v_j$:

$$P(v_i \longrightarrow v_j) = \frac{S_{ij}}{\sum_{i=1}^{|V|} S_{ij} \sum_{j=1}^{|V|} S_{ij}}. \quad (10)$$

The DADM contrasts the present prototype and the prototypes of typical traffic in the traffic model set if the present model's likelihood is more than an assumed threshold. This unusual traffic is seen as a DDoS attack model, or if the likelihood of the present prototype is lesser than an assumed threshold, this irregular traffic is viewed as a normal model [19]. In the training phase, the agent sets some of its beliefs with thresholds. These thresholds are used to reason and estimate the incoming traffic types between normal, abnormal, FC, or DDoS, as explained in Section 4.3. In addition, to distinguish the attack traffic from the typical or normal traffic for every peculiarity or anomaly traffic, the estimations of entropy on every model (M_1, M_2, \dots) are determined to portray the appropriation of the approaching origins and the targeted URLs. For the purpose of the investigation, S is the RFV of source IP addresses; T is the URLs of needed website pages, numeral one as the "HTTP requests," numeral two as the "normal." According to the meanings of each DDoS attacks and FC, the entropy $En(S)$ or $En(T)$ is determined by equation (11):

$$\frac{En(S)_2}{En(T)_2} > \frac{En(S)_1}{En(S)_1} > \frac{En(S)_1}{En(S)_3} > \frac{En(S)_4}{En(S)_4}. \quad (11)$$

Hypothetically, as appearing in equation (11), normal traffic, for the most part, has the smallest proportion of entropy quality and hence, differentiates the normal traffic from the DDoS attacks. At this point, the traffic is not investigated to check for the possibility of FC.

4.3. The Adaptive Traffic Control Module. This work contributes an agent-based Adaptive Traffic Control Module (ATCM), which has a Belief-Desire-Intention (BDI) agent architecture. With the BDI architecture, the adaptive agent facilitates the task selection decisions based on mapping desires with states of beliefs. These beliefs help the agent make decisions on the course of actions required to complete the tasks. The tasks involve monitoring the behavior of the incoming traffics data and controlling the flow of the traffic. The ATCM agent has a reactive component with which it adapts the traffic through implementing three functions: anomaly traffic identification *ati* function, anomaly traffic diagnosis *atd* function, and anomaly traffic handling *ath* function. These functions process according to the values of preexisting parameters or beliefs, including traffic attack behavior, normal traffic intensity, and history log of IP address. The belief constituents include information about traffics in the normal case as well as in the abnormal case. Desires, also referred to as goals, are reflective of what the agents intend to achieve. The agent can create desires or goals explicitly or generate them during runtime. However, in the ATCM agent, the desires are predetermined by the corresponding tasks, which are explained in the following paragraphs. Lastly, intentions are interwoven with plans, which are sequences of actions structured toward achieving the goals if there is a means of achieving them. The BDI architecture of the ATCM agent reasoning cycle is as follows:

Step 1: observe the network traffic conditions and update beliefs

Step 2: deliberate some defense desires to pursue based on the updated beliefs

- (i) Determine the available defense alternative desires
- (ii) Filter out unrelated or unachievable desires

Step 3: generate intentions of carrying out tasks to satisfy the selected desires

Step 4: execute actions to complete the corresponding task

In addition, with these components of BDI, the agents goals are differentiated from plans. There may be several plans prepared for achieving a goal so that if one plan fails, the agent considers other plans according to the reasoning cycle. In a case wherein there are multiple plans to achieve the goal, there is a cost-based selection function so that a less time-consuming plan is selected. Figure 3 shows the architecture of the proposed ATCM and the related adaptive functions.

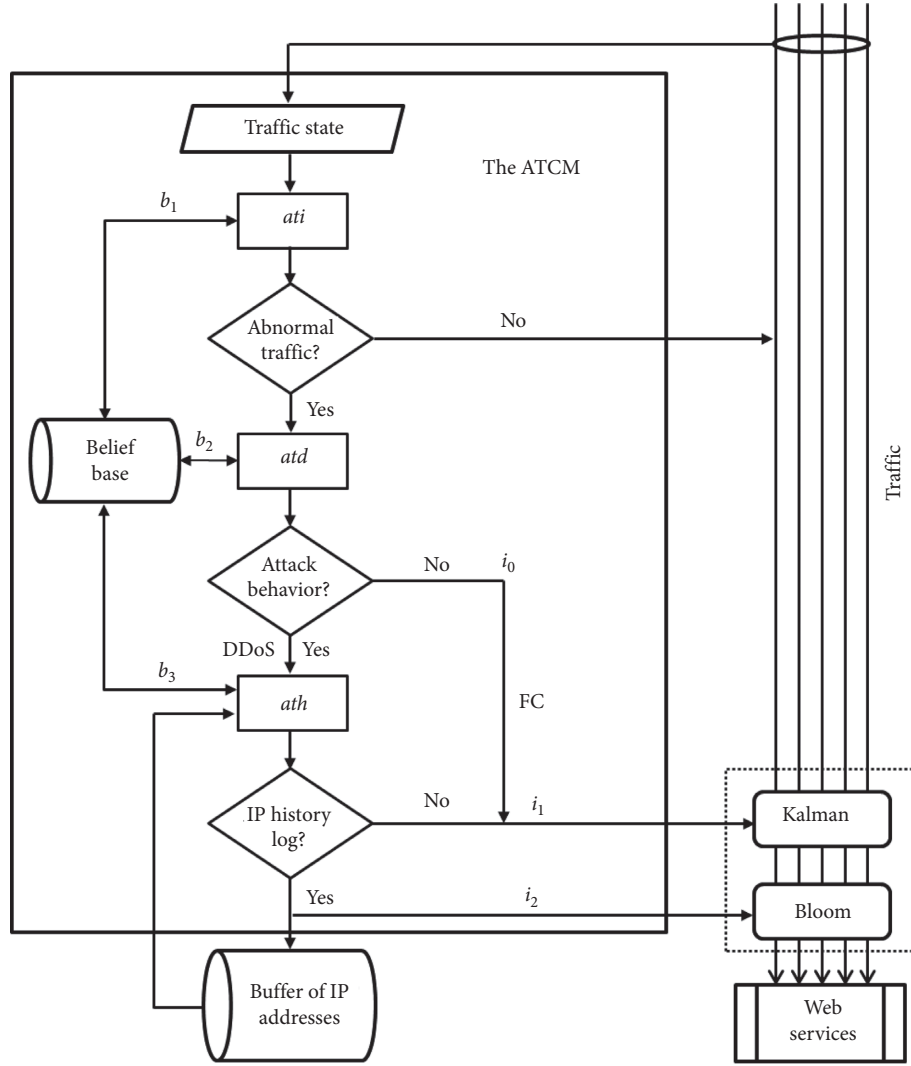


FIGURE 3: The architecture of the ATCM Agent.

Let the beliefs set, B , represent the network traffic parameters, which are: normal intensity, b_0 ; traffic intensity, b_1 ; traffic behavior, b_2 ; and IP history log, b_3 . The agent's beliefs trigger the desires set, D , to react based on the traffic conditions. Three functions: ati , atd , and ath , filter the desires, D , and translate the D to intentions, I . The I include the options of filters, i_0 ; block, i_1 ; and lock, i_2 traffic actions. They are defined as follows:

- (i) i_0 temporarily filters the traffic signals by random dropping of network requests. i_0 is invoked when FC is detected
- (ii) i_1 temporarily blocks the DDoS zombie network requests. i_1 is triggered when DDoS zombie IP addresses are detected
- (iii) i_2 permanently locks the DDoS demon network requests. i_2 is invoked when DDoS demon IP addresses are detected

Based on Figure 3, when the anomaly traffic with the source IP address reaches the agent of the DADM, the

ATCM agent controls the incoming traffic according to the three predefined intentions. In the first step of an agent cycle, the ati function checks the current traffic intensity with the b_1 . In case the current traffic intensity is more than b_1 , it means there is an attack traffic state. In case the current traffic intensity is less than b_1 , it means a normal traffic state, and the traffic is allowed to pass to the web service. Subsequently, in the second step of the agent cycle, and after it determines that the incoming traffic is a potential attack, then the second function, which is the atd and based on b_2 classifies the type of traffic into DDoS or FC according to equation (12).

$$r = \frac{n(\sum xy) - (\sum x)(\sum y)}{\sqrt{n\sum x^2 - (\sum x)^2} \sqrt{n(\sum y^2) - (\sum y)^2}} \quad (12)$$

At this point, any traffic that cannot be confirmed to be DDoS is labeled as FC. In the case of FC, the agent invokes the execution of random Kalman filter, rkf function, i.e., in the KBFM to block some of the traffic in a random manner

temporarily. In the case of DDoS, the exclusive decision is sent to the agent's last function, which is *ath* for further analysis. The *ath* based on b_3 , separates the DDoS traffic into Demons and Zombies. Then, the Demons' IP addresses are sent to the specific bloom filter for the *sbf* function to block the Demons permanently. Finally, the Demons' IP addresses are saved in the buffer IP address for future processing. Consequently, the Zombies' IP addresses are sent to the specific Kalman filter for the *skf* function to block the Zombies temporarily. All the filter functions are described in the KBFM.

4.4. Kalman and Bloom Filter Module. The Kalman and Bloom Filter Module (KBFM) comprise Kalman and Bloom filters. These filters are sequentially associated with network traffics. In the following segments, we clarify the significance and utilization of these filters.

4.4.1. Kalman Filter. The Kalman filter includes expressions that permit assessing the procedure state via productive and recursive computation such that the average of the squared error is limited [38]. In our suggested prototype, the Kalman filter is controlled by the agent. The agent sends signals to the Kalman filter for actuation or shut-off depending on the prearranged metrics and measure of approaching traffics by invoking one of the two functions. The first function is the random Kalman filter, *rkf*, function that performs impermanent blocks to random IP addresses. The second function is the specific Kalman filter, *skf*, function that performs impermanent blocks to the Zombies' IP addresses.

4.4.2. Bloom Filter. In 1970, Burton Howard Bloom created a filter named after him, called the Bloom filter, which can be described as a probabilistic data structure that is space-efficient. This filter can be utilized to test and decide whether a component is a member of a set. There is a plausibility of false-positive matches but not false-negatives. Eventually, a query can return as just "certainly not in set" or "potentially in set" in which components can be included to the set but not expelled when all things are considered as continuous events. The likelihood of false-positives becomes bigger when the number of components in the set increases [39]. In our suggested prototype, the bloom filter is controlled by the agent. It signals the bloom filter for initiation or shut-off depending on the predetermined metrics and measure of approaching traffics by invoking specific bloom filter, *sbf* function. This function performs permanent locking of the Demons' IP addresses.

5. Simulation Environment

This segment discusses the implementation of the simulator, the tests performed, and the execution measurements that are utilized in evaluating the APFA model. The simulator includes implementing the AL-DDoS model of Zhou et al. [36] as a base model. It also includes attack visualization and analysis modules to monitor the performance of the attack

traffics and the protection models. The simulation illustrates the impact of the DDoS and FC on the application layer with and without the AL-DDoS and APFA models.

5.1. Simulator Description. We build the simulation process design based on the attributes of the CIDDS dataset, and the AL-DDoS and APFA models. We use the CIDDS dataset to generate a large number of HTTP requests, including normal and abnormal HTTP requests. We divide the dataset into four weeks and model it with predetermined settings, which we describe in the following section. We design the APFA model in an almost similar design to the AL-DDoS model, except that we add the ATCM agent and some related changes.

Figure 4 shows the complete simulator design, which starts with a connection of the dataset to the simulator and dividing the data into training and testing sets. Traffic data from the training set is fed to the ATDM to determine the simulator thresholds by monitoring and analyzing the incoming traffics during the training phase (Steps 1–4 as shown by the ellipses). These thresholds are also received by other modules and the ATCM to form the agent's initial beliefs. In the subsequent testing phase, the ATDM distinguishes between the normal and the abnormal traffic, and passes the attention or dismiss the signal to the DADM. If an attention signal is received, the DADM traces the source of IP addresses that send the anomaly traffics (Step 5). Consequently, it sends these IP addresses in the form of SSMS to the ATCM agent for further analysis. While this study contributes the ATCM agent as discussed in the previous section, the BDI architecture of the ATCM agent controls the execution of three plans (Step 6) [40].

These plans identify traffic conditions (Step 6.1), classify the traffic type (Step 6.2), and control the traffic flow (Step 6.3). These could be selected sequentially or arbitrarily based on the traffic conditions and changes in the agent's beliefs. Finally, the filtering operation that satisfies the analysis of the traffic conditions is invoked (Step 7). Figure 5 shows the sequence of the interactions between the four modules of the APFA model of the simulator.

In this diagram, the rectangles show the modules, and the squares represent the procedures of each module, whereas the arrows show the direction of processing and the interaction in a time frame as follows:

- (i) User: exports the CIDDS dataset through a GUI
- (ii) ATDM: monitors and analyzes the incoming traffics to set thresholds
 - o sends attention signal to the DADM in the case of abnormal traffics
 - o sends dismiss signal to the DADM in the case of normal traffics
- (iii) DADM: traces traffic sources in the case of abnormal traffic based on the received signal
 - o Attention: traces the source of abnormal traffics and saves the IP addresses

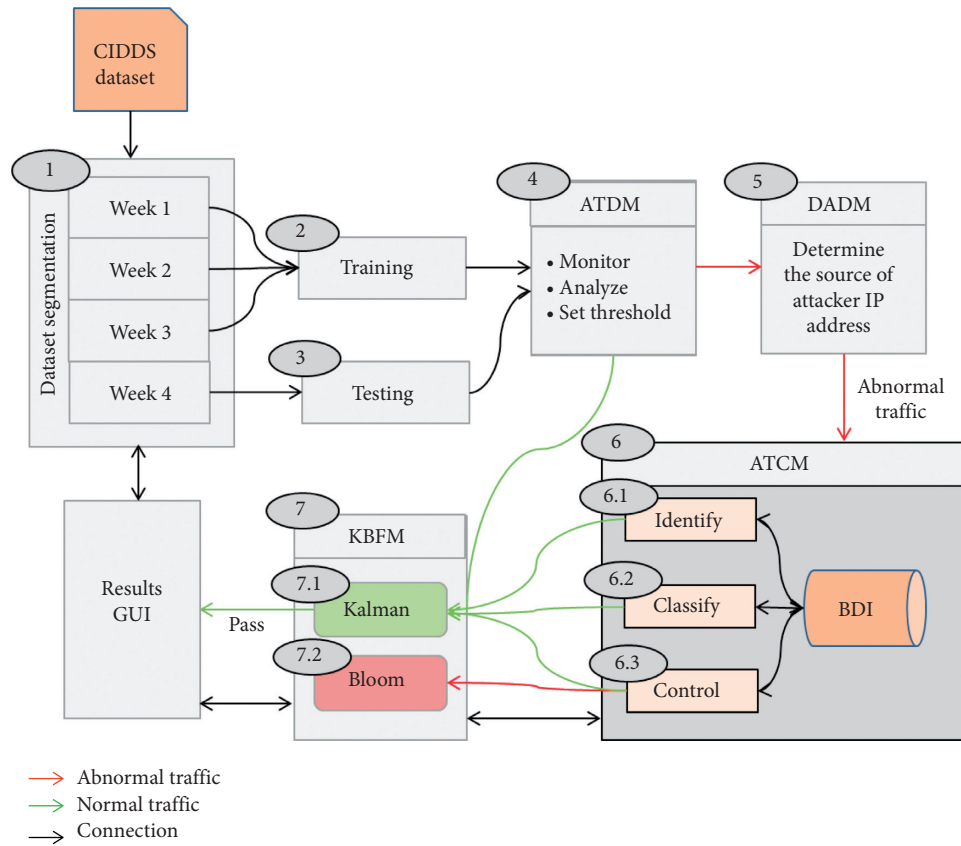


FIGURE 4: The steps of the simulation design.

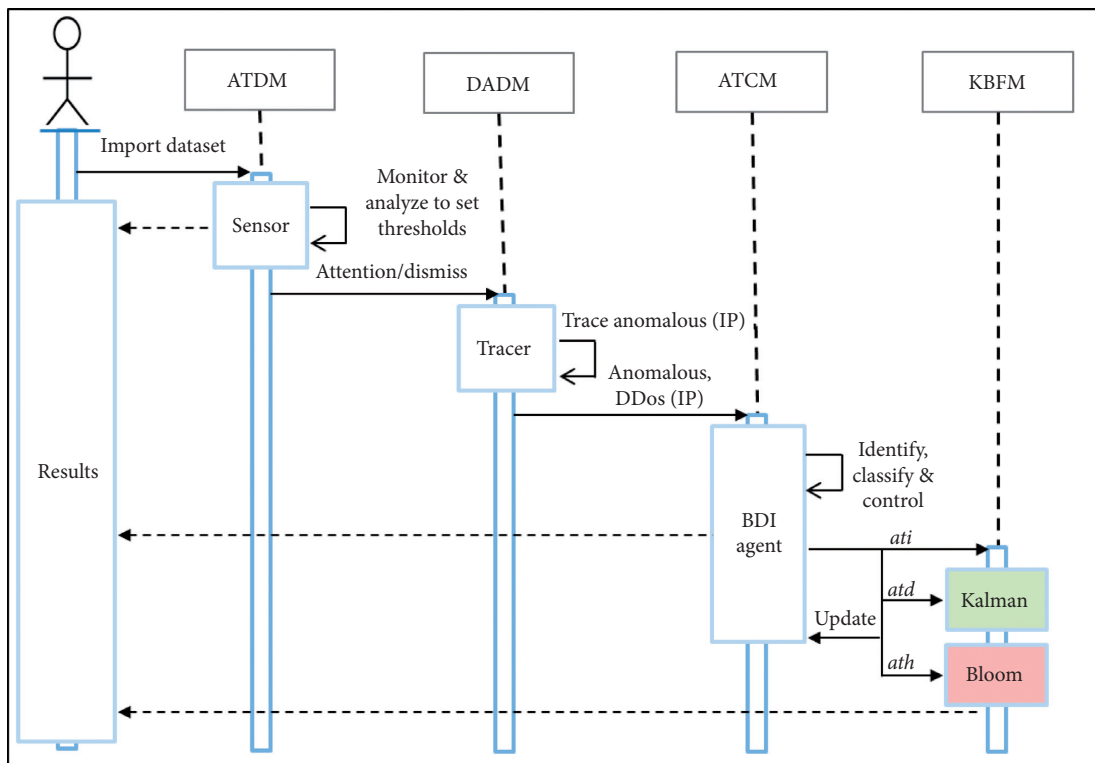


FIGURE 5: The sequence diagram of the APFA model simulation.

- o Dismiss: stops the tracing process
- (iv) ATCM: controls traffic flow in the case of abnormal traffics by invoking
 - o *ati*: identifies traffic conditions
 - o *atd*: classifies the traffic type
 - o *ath*: controls the traffic flow
- (v) KBFM: filters traffic flow in the case of abnormal traffics by invoking
 - o *rkf*: temporarily filters the traffic according to random IP addresses and specific thresholds
 - o *skf*: temporarily filters the traffic according to specific IP addresses and specific thresholds
 - o *sbf*: permanently filters the traffic according to specific IP addresses and specific thresholds
- (vi) Results: displays the information of the data analysis, processing cycles, and the simulation results through a GUI.

We specifically develop the simulator for this work by using C#, which is available on Visual Studio 2013 and Windows 7. For the implementation and testing of the simulator, the hardware used includes a 2.40 GHz Intel (R) Core (TM) i7-5500U processor and 16 GB RAM.

5.2. Dataset Setting. The original Coburg Intrusion Detection Data Sets (CIDDS) is a flow-based benchmark data segmented into five different groups of traffics, which are (normal, suspicious, unknown, attacker, and victim). The CIDDS dataset is used in the simulation to generate a large number of HTTP requests which include normal and abnormal HTTP requests. We neglect the Attack ID and Description features because they just give extra information about executed attacks [6]. This dataset was also used in similar recent studies, and the settings of the dataset in our work follow the work of Sreeram and Vuppala [19] and Mohamed et al. [23]. Figure 6 shows the CIDDS dataset network environment.

The performance of the IDS and IPS against flooding types of attack is specifically evaluated using the CIDDS dataset. Figure 7 shows the segmentation of the original dataset into four weeks and seven days. The week 1 folder contains 9,412 IP addresses and sends 172,838 requests; the week 2 folder consists of 8,357 IP addresses and sends 159,373 requests. The week 3 folder holds 2,605 IP addresses and sends 70,533 requests, and the week 4 folder contains 15,369 IP addresses and sends 303,024 requests. We compile these weeks in a file and reorganize the data instances accordingly. We then segment the CIDDS dataset into 60% training and 40% testing sets, as shown in Figure 7. Hence, the training and testing ratio of the CIDDS dataset is segmented according to the related work for which the comparison is made with them.

5.3. Simulation Setting. The advantages of using the CIDDS dataset in this study are that it is current and customizable. The simulation program is written with the C# programming

language in a virtual environment to regenerate customized datasets that are used in this work. However, the original CIDDS dataset does not include FC labels. Subsequently, we set the ground truth of DDoS and FC traffics to train for the thresholds and methods in the simulation based on the actual data of the CIDDS dataset and statistical analysis of the data using equations (4) and (5). The analysis of the training phase results shows the average frequency of incoming requests. The high request frequency signifies the possibility of DDoS or FC traffic. Moreover, any traffic that cannot be confirmed to be DDoS and have DDoS characteristics are labeled as FC. Figure 8 shows an example of the statistical analysis, which identifies an average frequency of 37000 requests from the clock time of 4:20 to 19:20 on day 1 of week 1.

We set the simulation parameters during the training phase for both AL-DDoS and APFA models. The training set almost represents 60%, and the testing set represents the other 40% of the original dataset. It includes the Support, Confidence, and Possibility results when the window parameter is set to be 130. Correspondingly, the support, confidence, and possibility represent the values of the up triangle, diagonal, and down triangle.

We perform different tests to choose an optimal value for all the testing parameters. Figure 9 shows an example of the CIDDS dataset that generates web traffic, with original derivations and 2-step Kalman calibration. The results show the detection of noticeable deviation for the abnormal traffics.

The default M traffic model here represents the traffics of the three weeks, and it has been calculated as discussed before. Table 3 shows the support, confidence, possibility, entropy, and minimum and maximum values of the M model. The system is implemented based on these parameters, in which the period is set to 7, and the SSM is fixed to be 130.

During the training phase, the agent architecture includes three cases, anomaly traffics, DDoS traffics, and FC traffics, along with the agent's reaction setting for the three cases. The conditions of the anomaly traffic are classified based on the traffic behavior into irregular, t_0 ; discrete, t_2 ; and continuous, t_2 , as shown in Table 4. This classification helps to identify the traffic types during the testing phase.

Based on Table 4 and as described in Section 4.3, the *ati* has the elementary objective of identifying whether the incoming traffic is normal or abnormal. In a scenario where 7.6428 is a threshold value for traffic intensity according to the ATDM analysis, the current incoming traffic value is updated in b_1 , then it is compared with b_0 . If the b_1 value is lesser than that of b_0 (i.e., 7.6428), then the case is recognized as regular traffic. If the b_1 ' value is higher than b_0 , then the next stage of calling the *atd* is triggered to determine the traffic condition. For FC traffic, with a traffic intensity of 7.6428, we follow the same steps as the first case. Abnormal traffics are diagnosed by the *atd* according to b_2 . In this scenario, based on the correlation coefficient, the traffic behaves as a discrete flow, $b_2 \rightarrow t_1$ and *atd*: $b_2 \rightarrow i_0$. As a result, the agent instructs the KBFM to invoke the *rkf* with 2745 capacity, temporarily filtering out 2745 IP addresses. When the volume of the DDoS traffics, which are detected in this stage, is 10838 IP addresses, each IP address sends a random number of requests. This traffic model is sent to the

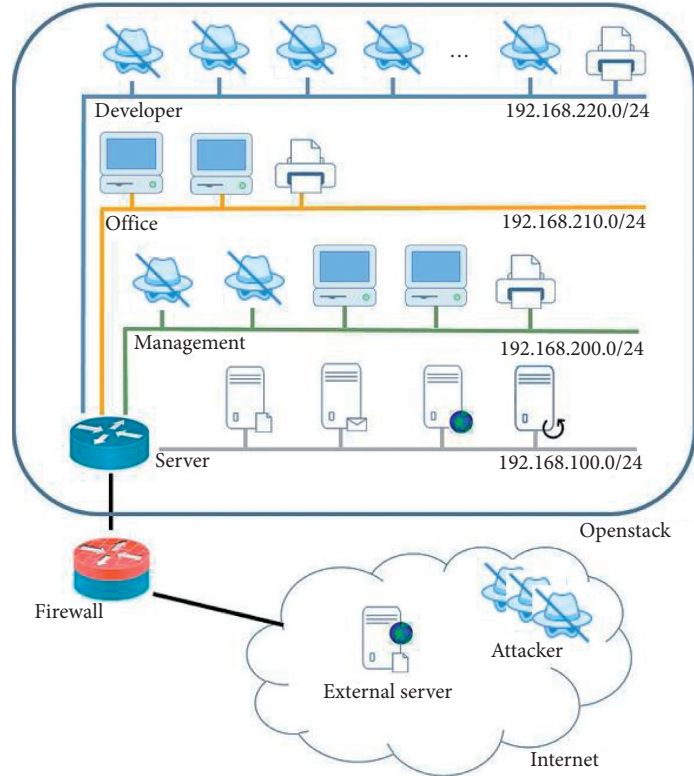


FIGURE 6: The CIDDS dataset network environment [6].

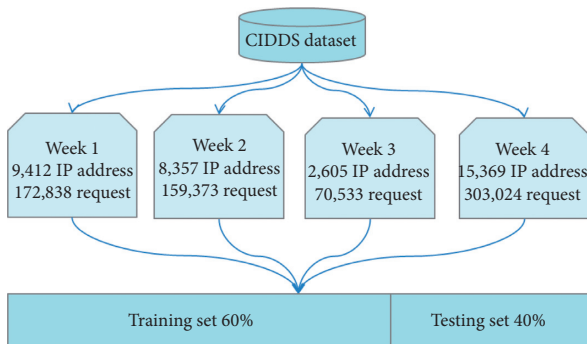


FIGURE 7: The sectioned parts of the CIDDS dataset.

ATCM agent to identify, classify, and control the traffic according to the agent functions. When incoming traffic is 10838 and has a continuous flow, $b_2 \rightarrow t_2$, then, the $atd: b_2 \rightarrow i_1 \wedge i_2$ is considered as a DDoS attack. This case implies invoking ath , which handles the Zombies, r_1 and Demons, and r_2 requests, $ath: r_1 \rightarrow i_1 \wedge r_2 \rightarrow i_2$. The agent instructs the KBFM to invoke the skf in which $ath: r_1 \rightarrow skf$ and the sbk in which $ath: r_2 \rightarrow sbk$ with the corresponding SSM information, which temporarily locks the r_1 and permanently blocks the r_2 .

6. Results and Discussion

The test results evaluate the performance of the APFA model. Then, the performance of the model is compared with three similar models of Sreeram and Vuppala [19], Mohamed et al.

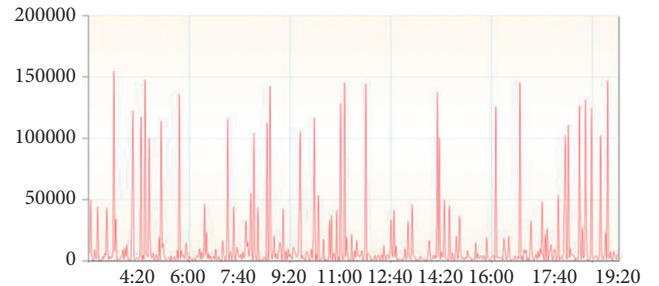


FIGURE 8: The average frequency of incoming requests of the training phase.

[23], and Zhou et al. [36]. We perform two tests on the CIDDS dataset to evaluate the APFA model in which the CIDDS dataset generates normal and anomaly traffics. We perform the first test for the AL-DDoS model, which only detects normal and DDoS traffics. We conduct the second test for the APFA model, which detects normal, DDoS, and FC traffics. The data of week 4 (after the modification, it becomes 40% of the dataset as explained in Section 5.2) are used for testing the model. They contain a discrete and random series of incoming requests, including DDoS and FC targeting the NAL. Table 5 shows the daily frequency of the incoming requests of week 4. The traffics of week 4 are divided into seven days, starting with traffic day 1 with 38,919 requests and ending with day 7 with 33,228 requests. We observe from the table that day 7 has visibly lower requests than the daily average incoming requests, which are 43,289, and day 4 has visibly greater requests than the daily average of incoming requests.

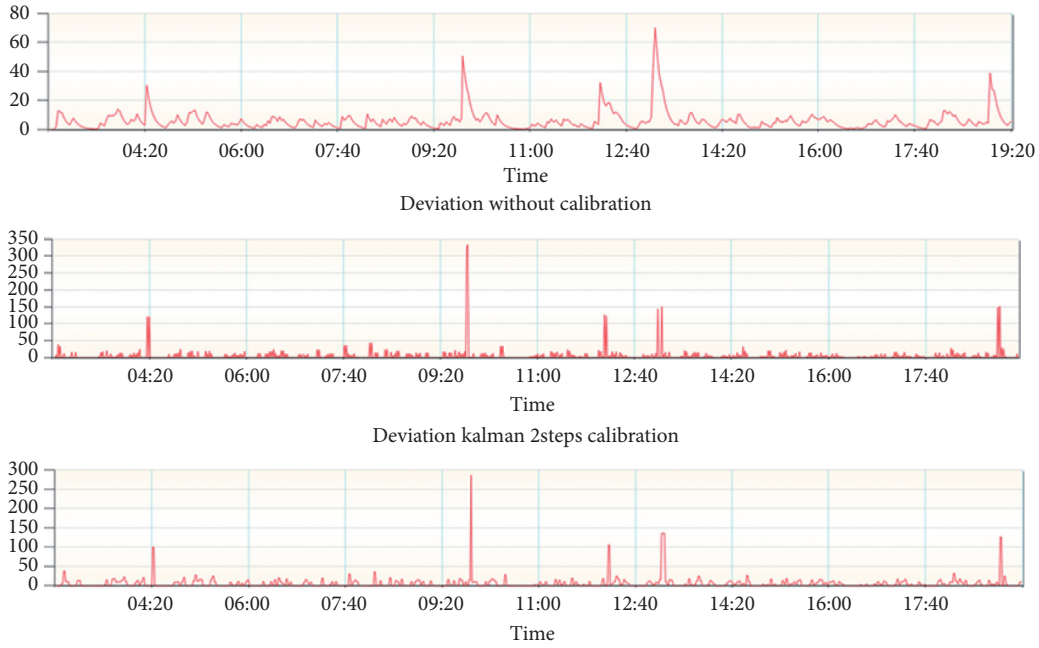


FIGURE 9: A sample of web traffics from original derivations and 2-step Kalman.

TABLE 3: Values of the M traffic model.

	Support	Confidence	Possibility	Entropy	Min	Max
M	0.0000	0.0021	0.050	7.6428	0.00000	244,557.99

TABLE 4: Agent parameters setting.

Traffic conditions	Traffic	Behavior
Anomaly	13,583	t_0
FC	2,745	t_2
DDoS	10,838	t_2

TABLE 5: The average incoming requests in the testing phase.

Day	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7
Frequency	38,919	47,328	44,921	56,991	39,835	41,802	33,228

6.1. Results of the AL-DDoS Model. The performance of the AL-DDoS base model is calculated according to the window size, period, and SSM parameters for every execution of external traffic data. The AL-DDoS model performance is evaluated based on correctly classifying traffic instances into normal, and DDoS attack traffics only. The volume of attack traffics detected by the AL-DDoS model is 26,8496 requests triggered by 13,583 IP addresses. Subsequently, the results show that the AL-DDoS model detects DDoS attacks and blocks the IP addresses with an accuracy of 99.13%, precision of 99.14%, and sensitivity of 99.99%. However, the AL-DDoS model lacks handling FC and Zombies traffics and considers all DDoS traffic as Demons.

6.2. Results of the APFA Model. The results of the APFA also present information about the number of anomaly requests of

the same week 4 from the dataset. The data are first passed through the *ati* function in the identify traffic conditions phase of the ATCM agent. The *ati* detects a total number of 34,528 requests as normal and 268,496 as abnormal according to the traffic intensity parameters with a total cost of 15,369 cycles. Then, the attack requests are classified by the *atd* function, in the classify traffic type phase, into 264,551 requests as DDoS and 3,945 requests as FC, with a total cost of 13,583 cycles. The *ath* function, in the control traffic flow phase, controls the traffic according to the DDoS types of 175,624 Demons and 88,927 Zombies, with a total cost of 10,838 cycles. Then, the KBFM implements the required blocking and locking of the requests. Table 6 shows the input, processing, and output of each function in the ATCM agent.

Figure 10 shows the classification results of the daily attack requests of week 4 by DDoS and FC. The average daily DDoS

TABLE 6: The results of the agent run cycle during week 4.

No.	Run phase	Cost	Input req.	Output			
				Normal		Abnormal	
1	Identify normal/abnormal	15,369	303,024	IP 1,786	Req. 34,528	IP 13,583	Req. 268,496
2	Classify DDoS/FC	13,583	268,496	DDoS		FC	
			IP 10,838	Req. 264,551	IP 2,745	Req. 3,945	
3	Control demons/zombies	10,838	264,551	Demons		Zombies	
			IP 7,186	Req. 175,624	IP 3,670	Req. 88,927	

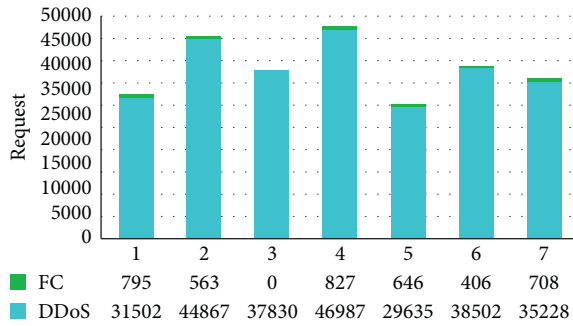


FIGURE 10: The daily traffics of DDoS and FC.

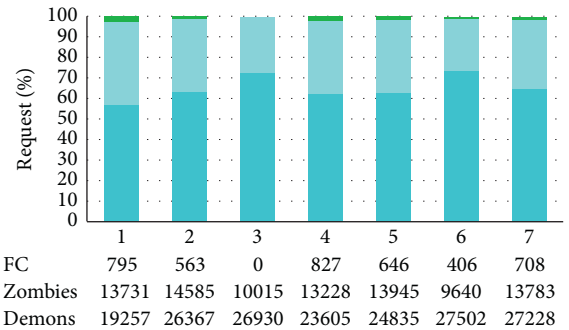


FIGURE 11: The daily traffics of Demons and Zombies.

requests are 37,793, which is almost 96% of the average daily abnormal requests, while the average daily FC requests is 563, which is almost 2% of the average daily abnormal requests.

Figure 11 shows the number of Demons and Zombies requests by the DDoS traffic. The average daily requests of Demons is 25,103, which is almost 66% of the DDoS requests, while the average daily requests of Zombies is 12,703, which is almost 44% of the DDoS requests.

In general, the results show that the APFA model is able to detect and distinguish the DDoS and FC traffics. It then recognizes Demon and Zombie requests of the DDoS traffic. The phase that is responsible for identifying the possibility of abnormal traffic achieves the results of 99.11% accuracy, 99.14% precision, and 99.99% sensitivity. The phase that is responsible for classifying DDoS and FC traffics achieves the results of 99.92% accuracy, 99.85% precision, and 99.96% sensitivity. The phase that is responsible for controlling Demons and Zombies traffics achieves the results of 99.91% accuracy, 99.89% precision, and 99.93% sensitivity. Ultimately, the APFA model achieves an overall accuracy of 99.64%, precision of 99.62%, and sensitivity of 99.96%. Table 7 shows the performance results of the APFA model.

Figure 12 shows the daily performance results of the APFA model. As observed from the figure, the APFA model's performance improves day by day due to the system's ability to progress its adaptive behavior with time.

6.3. Analysis and Discussion. In the deep view of the Internet network, there are many components that participate in making up the web application framework. The HTTP requests sent from web clients are processed by a web server and forwarded to the application server based on many

TABLE 7: The performance results of the APFA model.

Evaluation metrics			
Run phase	Accuracy %	Precision %	Sensitivity %
Identify normal/abnormal	99.11	99.14	99.99
Classify DDoS/FC	99.92	99.85	99.96
Control demons/zombies	99.91	99.89	99.93
Overall results	99.64	99.62	99.96

configuration parameters like URL path prefix. These requests are directed to one of the web applications hosted by the application server. A DDoS attack is a malicious event that targets web servers without the need for internal system access. Consequently, the attack is not easily detected in its early stage. The attack entails the involvement of a huge army of Zombies to cause conceivable damage to the network. Critical attacks include concentrating a huge number of nodes as a single target to inflict devastating damage to users and completely overwhelm the network. Another type of flooding traffics, which is FC, is depicted as network traffic that is quite similar to DDoS traffic, but it comes from valid users when a huge number of them access a particular website simultaneously.

The benchmarking works of Sreeram and Vuppala [19], Mohamed et al. [23], and Zhou et al. [36] only deal with two types of traffics, which are normal traffic and attack traffic. The AL-DDoS base model of Zhou et al. [25] only detects anomaly traffic requests, determines the source of each IP address, saves these IP addresses in a bloom filter, and locks

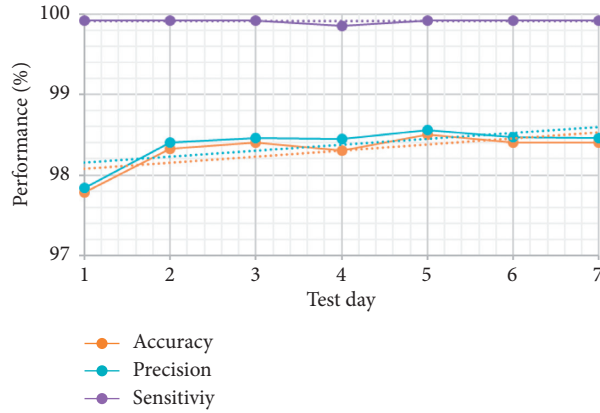


FIGURE 12: The results of the daily performance.

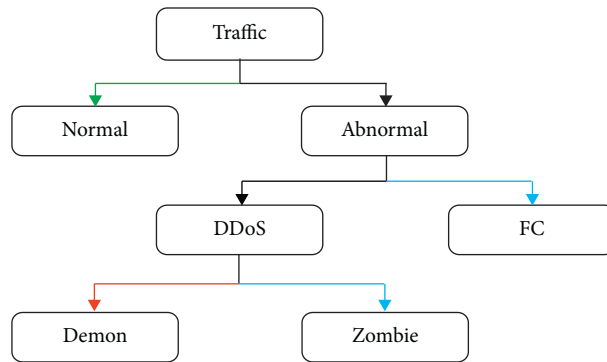


FIGURE 13: The hierarchical traffic control of the APFA model.

TABLE 8: The performance comparison with the benchmark models.

Model	Technique	Dataset	Traffic type				Accuracy %
			FC	DDoS	Zombies	Demons	
Sreeram and Vuppala [19]	Bat algorithm	CIDDS	—	✓	—	—	94.80
Mohamed et al. [23]	Random forest	CIDDS	—	✓	—	—	99.54
Zhou et al. [36]	Statistical	CIDDS	✓	✓	—	—	99.13
APFA model	Adaptive agent and statistical	CIDDS	✓	✓	✓	✓	99.64

all the anomaly traffic requests. However, among those IP addresses, there are many cases of FC and Zombies’ requests that belong to legitimate users yet are confined to permanent lock. Subsequently, this work proposes an Adaptive Protection of Flooding Attacks (APFA) model that identifies abnormal traffic requests and then further classifies the abnormal traffic requests to DDoS and FC. It then further classifies the DDoS to Demons and Zombies and applies control procedures to temporarily block FC and Zombies’ IP addresses and permanently lock Demons IP addresses. Figure 13 shows the hierarchy of the APFA model control to the NAL against flooding traffics.

Researchers have proposed, developed, and implemented numerous techniques to safeguard the NAL against DDoS and FC attacks. However, hidden malicious traffic behind valid traffic and the FC continue to plague these defense models. Many of these models are unable to

differentiate between valid and invalid malicious traffics positively. In this paper, we proposed the APFA model, the performance of which is evaluated by comparing it with three benchmark models. The three models are tested for similar properties and in similar conditions, and there is no bias to declare. The comparison results are summarized in Table 8, which show that the APFA model outperforms the other three models.

Eventually, defense methods are continuously evolving to improve and protect networks and computer infrastructures. The APFA model, like any other model, represents another attempt to provide variable effectiveness against DDoS attacks and FC flooding. Three sources of limitations need to be highlighted according to the scope of this work. Firstly, this work does not consider the processing time in the evaluation in which the adaptation and decision-making capabilities of the agent might slow down the

performance of the model compared with the other tested models. Secondly, the model is only tested using the CIDDS dataset, which could be another constraint on the evaluation. Finally, the testing of the model does not cover the low-rate cases of DDoS attacks that are difficult to discover with existing solutions. Nevertheless, such DDoS attacks have no harmful impact on real-world network systems.

7. Conclusion and Future Work

Progressively, there are assortments of administrations and applications that utilize cyberspace, including web applications, cloud-computing applications, and Internet of things applications. DDoS attack and FC flooding could be a legitimate annoyance for the cybersecurity of network systems. The advancement of innovative communication technology of the current computer applications has brought along the danger of these sorts of threats. Subsequently, various investigations have focused on these threats to embed variable protection prototypes. The proposed mainstream models lack the ability to deal with illegitimate DDoS traffics, which are accompanied by FC traffics. They permanently lock all DDoS traffic and treat legitimate traffic of FC and Zombies as Demons. Consequently, this paper proposes an Adaptive Protection of Flooding Attacks (APFA) model, which attempts to protect the NAL against DDoS attack and FC flooding and solve the problem of permanently locking the traffics of legitimate users. The APFA model consists of the Abnormal Traffic Detection Module (ATDM) and DDoS Attack Detection Module (DADM) that are adopted from the AL-DDoS model of Zhou et al. (2014). It further includes a new Adaptive Traffic Control Module (ATCM) and Kalman and Bloom Filters Module (KBFM). Our main contribution is the ATCM module, which integrates an adaptive agent to recognize and isolate normal from abnormal traffic and hinders all ill-conceived traffics. The APFA model is implemented and tested using the CIDDS dataset to produce standard scenarios targeting web servers. The test results show that the APFA model outperforms three similar models and achieves an accuracy of 0.9964, a precision of 0.9962, and a sensitivity of 0.9996. Two points of improvement can be further investigated, which are the effect of cost function on agent adaptive behavior and enabling the DADM to detect low rate and FC-like DDoS attack patterns. In addition, putting and testing the APFA model online could furnish greater confidence in its capability to perform in real-time.

Data Availability

The used dataset for this research is available online and has a proper citation within the paper contents.

Conflicts of Interest

The authors declare that they have no conflicts of interest to be addressed related to this work.

Acknowledgments

The authors would like to thank the Center of Intelligent and Autonomous Systems (CIAS), Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia (UTHM) for supporting this work.

References

- [1] N. S. Rao, K. C. Sekharaiah, and A. A. Rao, "A survey of distributed denial-of-service (DDoS) defence techniques in ISP domains," in *Innovations in Computer Science and Engineering*, pp. 221–230, Springer, Singapore, Asia, 2019.
- [2] Z. K. Maseer, R. Yusof, N. Bahaman, S. A. Mostafa, and C. F. M. Foozy, "Benchmarking of machine learning for anomaly based intrusion detection systems in the CICIDS2017 dataset," *IEEE Access*, vol. 9, pp. 22351–22370, 2021.
- [3] B. A. Khalaf, S. A. Mostafa, A. Mustapha, M. A. Mohammed, and W. M. Abdulllah, "Comprehensive review of artificial intelligence and statistical approaches in distributed denial of service attack and defense methods," *IEEE Access*, vol. 7, pp. 51691–51713, 2019.
- [4] S. Khandelwal, Cybercrime technical report, 2016, <http://thehackernews.com/2016/01/biggest-ddos-attack.html>.
- [5] A. Zulhlimi, S. A. Mostafa, B. A. Khalaf, A. Mustapha, and S. S. Tenah, "A comparison of three machine learning algorithms in the classification of network intrusion," in *Proceedings of the International Conference on Advances in Cyber Security*, pp. 313–324, Penang, Malaysia, July 2020.
- [6] M. Ring, S. Wunderlich, D. Grödl, D. Landes, and A. Hotho, "Flow-based benchmark data sets for intrusion detection," in *Proceedings of the 16th European Conference on Cyber Warfare and Security*, pp. 361–369, Dublin, Ireland, June 2017.
- [7] O. S. Babatunde, A. R. Ahmad, S. A. Mostafa et al., "A smart network intrusion detection system based on network data analyzer and support vector machine," *International Journal of Emerging Trends in Engineering Research*, vol. 8, no. 1, pp. 213–220, 2020.
- [8] K. Lee, J. Kim, K. H. Kwon, Y. Han, and S. Kim, "DDoS attack detection method using cluster analysis," *Expert Systems with Applications*, vol. 34, no. 3, pp. 1659–1665, 2008.
- [9] S. Behal, K. Kumar, and M. Sachdeva, "Characterizing DDoS attacks and flash events: review, research gaps and future directions," *Computer Science Review*, vol. 25, 2017.
- [10] B. A. Khalaf, S. A. Mostafa, A. Mustapha, and N. Abdullah, "An adaptive model for detection and prevention of DDoS and flash crowd flooding attacks," in *Proceedings of the 2018 International Symposium on Agent, Multi-Agent Systems and Robotics (ISAMSR)*, pp. 1–6, IEEE, Putrajaya, Malaysia, August 2018.
- [11] S. Yu, W. Zhou, W. Jia, S. Guo, Y. Xiang, and F. Tang, "Discriminating DDoS attacks from flash crowds using flow correlation coefficient," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 6, pp. 1073–1080, 2012.
- [12] A. Bhandari, A. L. Sangal, and K. Kumar, "Characterizing flash events and distributed denial-of-service attacks: an empirical investigation," *Security and Communication Networks*, vol. 9, no. 13, pp. 2222–2239, 2016.
- [13] S. Bhatia, G. Mohay, A. Tickle, and E. Ahmed, "Parametric differences between a real-world distributed denial-of-service attack and a flash event," in *Proceedings of the 2011 Sixth International Conference on Availability, Reliability and*

- Security (ARES)*, pp. 210–217, IEEE, Vienna, Austria, August 2011.
- [14] I. Kotenko and A. Ulanov, “Agent-based simulation of DDOS attacks and defense mechanisms,” *International Journal of Computing*, vol. 4, pp. 113–123, 2014.
 - [15] S. N. Shiaeles, V. Katos, A. S. Karakos, and B. K. Papadopoulos, “Real time DDOS detection using fuzzy estimators,” *Computers & Security*, vol. 31, no. 6, pp. 782–790, 2012.
 - [16] H. Kaur, G. Singh, and J. Minhas, “A review of machine learning based anomaly detection techniques,” *International Journal of Computer Applications Technology and Research*, vol. 2, no. 2, pp. 1307–1319, 2013.
 - [17] V. Katkar, A. Zinjade, S. Dalvi, T. Bafna, and R. Mahajan, “Detection of DoS/DDoS attack against HTTP servers using naive Bayesian,” in *Proceedings of the 15th International Conference on Computing Communication Control and Automation*, pp. 280–285, IEEE, Pune, India, February 2015.
 - [18] M. Barrionuevo, M. Lopresti, N. Miranda, and F. Piccoli, “An anomaly detection model in a LAN using K-NN and high-performance computing techniques,” *Argentine Congress of Computer Science Journal*, vol. 790, no. 17, pp. 219–228, 2017.
 - [19] I. Sreeram and V. P. K. Vuppala, “HTTP flood attack detection in application layer using machine learning metrics and bio inspired bat algorithm,” *Applied Computing and Informatics*, vol. 15, 2017.
 - [20] Q. Yan, W. Huang, X. Luo, Q. Gong, and F. R. Yu, “A multi-level DDOS mitigation framework for the industrial internet of things,” *IEEE Communications Magazine*, vol. 56, no. 2, pp. 30–36, 2018.
 - [21] A. Verma and V. Ranga, “Statistical analysis of CIDDS-001 dataset for network intrusion detection systems using distance-based machine learning,” *Procedia Computer Science*, vol. 125, pp. 709–716, 2018.
 - [22] A. Verma and V. Ranga, “On evaluation of network intrusion detection systems: statistical analysis of CIDDS-001 dataset using machine learning techniques,” *Pertanika Journal of Science & Technology*, vol. 26, no. 3, 2018.
 - [23] I. Mohamed, K. Afdel, and M. Belouch, “Detection system of HTTP DDOS attacks in a cloud environment based on information theoretic entropy and random forest,” *Security and Communication Networks*, vol. 2018, Article ID 1263123, 13 pages, 2018.
 - [24] I. Kotenko and A. Ulanov, “Agent-based simulation of distributed defense against computer network attacks,” in *Proceedings of the 20th European Conference on Modelling and Simulation*, pp. 1–6, Springer, Bonn, Germany, May 2006.
 - [25] D. Juneja, R. Chawla, and A. Singh, “An agent-based framework to counter attack DDOS attacks,” *International Journal of Wireless Networks and Communications*, vol. 1, no. 2, pp. 193–200, 2009.
 - [26] R. Kesavamoorthy and K. R. Soundar, “Swarm intelligence based autonomous DDOS attack detection and defense using multi agent system,” *Cluster Computing Journal*, vol. 7, no. 11, pp. 1–8, 2018.
 - [27] K. Singh, K. Singh Dhindsa, and B. Bhushan, “Performance analysis of agent based distributed defense mechanisms against DDOS attacks,” *International Journal of Computing*, vol. 17, no. 1, pp. 15–24, 2018.
 - [28] H. Lin, Z. Yan, and Y. Fu, “Adaptive security-related data collection with context awareness,” *Journal of Network and Computer Applications*, vol. 126, no. 3, pp. 88–103, 2019.
 - [29] S. A. Mostafa, S. S. Gunasekaran, M. S. Ahmad, A. Ahmad, M. Annamalai, and A. Mustapha, “Defining tasks and actions complexity-levels via their deliberation intensity measures in the layered adjustable autonomy model,” in *Proceedings of the 2014 International Conference on Intelligent Environments*, pp. 52–55, IEEE, Shanghai, China, June 2014.
 - [30] L. Xiong, S. Goryczka, and V. Sunderam, “Adaptive, secure, and scalable distributed data outsourcing: a vision paper,” in *Proceedings of the 2011 Workshop on Dynamic Distributed Data-Intensive Applications, Programming Abstractions, and Systems ACM*, pp. 1–6, San Jose, CA, USA, June 2011.
 - [31] D. Chefrour, “Developing component based adaptive applications in mobile environments,” *Applied Computing Journal*, vol. 77, no. 19, pp. 1146–1150, 2005.
 - [32] A. Evesti, H. Abie, and R. Savola, “Security measuring for self-adaptive security,” in *Proceedings of the 2014 European Conference on Software Architecture Workshops*, pp. 5–11, IEEE, Vienna, Austria, August 2014.
 - [33] J. Cheng, C. Zhang, X. Tang, V. S. Sheng, Z. Dong, and J. Li, “Adaptive DDOS attack detection method based on multiple-Kernel learning,” *Security and Communication Networks*, vol. 2018, Article ID 5198685, 2018.
 - [34] “CIDDS, coburg-intrusion-detection-data-sets,” 2017, <https://www.hs-coburg.de/forschung-kooperation/forschungsprojekte-oeffentlich/ingenieurwissenschaften/cidds-coburg-intrusion-detection-data-sets.html>.
 - [35] S. Wen, W. Jia, W. Zhou, W. Zhou, and C. Xu, “CALD: surviving various application-layer DDOS attacks that mimic flash crowd,” in *Proceedings of the 2010 4th International Conference on Network and System Security (NSS)*, pp. 247–254, IEEE, Victoria, Australia, September 2010.
 - [36] W. Zhou, W. Jia, S. Wen, Y. Xiang, and W. Zhou, “Detection and defense of application-layer DDOS attacks in backbone web traffic,” *Future Generation Computer Systems*, vol. 38, pp. 36–46, 2014.
 - [37] M. De Donno, N. Dragoni, A. Giarretta, and A. Spognardi, “DDoS-capable IoT malwares: comparative analysis and mirai investigation,” *Security and Communication Networks*, vol. 2018, 2018.
 - [38] K. Fujii, “Extended Kalman filter,” *Reference Manual*, pp. 14–22, 2013.
 - [39] J. Bruck, J. Gao, and A. Jiang, “Weighted bloom filter,” in *Proceedings of the 2006 IEEE International Symposium on Information Theory*, pp. 2304–2308, IEEE, Seattle, WA, USA, July 2006.
 - [40] S. A. Mostafa, A. Mustapha, A. A. Hazeem, S. H. Khaleefah, and M. A. Mohammed, “An agent-based inference engine for efficient and reliable automated car failure diagnosis assistance,” *IEEE Access*, vol. 6, pp. 8322–8331, 2018.

Research Article

Privacy-Preserving Publication of Time-Series Data in Smart Grid

Franklin Leukam Lako ^{1,2}, Paul Lajoie-Mazenc ², and Maryline Laurent ¹

¹Department of Network and Telecommunication Services, Samovar Lab, Télécom SudParis, Institut Polytechnique de Paris, 19 rue Marguerite Perey, Palaiseau 91120, France

²EDF Lab Paris-Saclay, 7 Boulevard Gaspard Monge, Palaiseau 91120, France

Correspondence should be addressed to Franklin Leukam Lako; franklin.leukam-lako@edf.fr

Received 4 December 2020; Revised 18 February 2021; Accepted 2 March 2021; Published 26 March 2021

Academic Editor: Chalee Vorakulpipat

Copyright © 2021 Franklin Leukam Lako et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The collection of fine-grained consumptions of users in the smart grid enables energy suppliers and grid operators to propose new services (e.g., consumption forecasts and demand-response protocols) allowing to improve the efficiency and reliability of the grid. These services require the knowledge of aggregate consumption of users. However, an aggregate can be vulnerable to re-identification attacks which allow revealing the users' individual consumption. Revealing an aggregate data is a key privacy concern. This paper focuses on publishing an aggregate of time-series data such as fine-grained consumptions, without indirectly disclosing individual consumptions. We propose novel algorithms which guarantee differential privacy, based on the discrete Fourier transform and the discrete wavelet transform. Experimental results using real data from the Irish Commission for Regulation of Utilities (CRU) demonstrate that our algorithms achieve better utility than previously proposed algorithms.

1. Introduction

A smart city is a designation given to a city that incorporates information and communication technologies to enhance the quality and performance of urban services such as energy, transportation, and utilities in order to reduce resource consumption, wastage, and overall costs. The overarching aim of a smart city is to enhance the quality of living for its citizens through smart technology [1–3].

The smart grid is an important part of the smart city. Indeed, the smart grid allows greater penetration of highly variable renewable energy sources such as solar and wind power in the smart city.

The smart grid modernizes the traditional electricity grid by establishing a communication infrastructure in parallel to the energy delivery network. This infrastructure is used by the grid operators and suppliers to remotely collect fine-grained consumptions from household smart meters and to provide new energy services such as consumption forecasts or demand-response. These services are suitable for improving the efficiency and reliability of the grid, saving energy and, more generally, for optimizing energy usage. In

particular, forecasting enables the supplier to predict future consumptions based on past aggregate data in order to improve the grid and retail operations and enhance energy trading [4], while demand-response (DR) aims to shift the users' consumption from peak to off-peak periods in order to avoid consumption peaks in the smart city.

However, aggregates are vulnerable to reidentification attacks, such as set difference attacks [5] in which two aggregates that differ by a single consumer allow learning this individual consumption. Since the individual consumption data collected by smart meters reflect the use of all electric appliances by inhabitants in a household over time and enable to deduce the behaviors, activities, age, or preferences of the inhabitants [6–11], revealing an aggregate is a key privacy concern.

Differential privacy (DP) [12] allows publishing an aggregate data while guaranteeing that an attacker does not learn any individual inputs from the aggregate. However, the noise added by DP often leads to a loss of utility. Moreover, publishing time-series data such as users' consumption, which are correlated, by using DP, results in more noise added than publishing a single aggregate for the same

privacy guarantee. Thus, disclosing time-series data leads to more loss of utility. Utility can be improved by increasing the size of the aggregate. Eibl and Engel [13] showed that for real-world smart metering, the aggregation group size must be of the order of thousands of smart meters in order to have reasonable utility. This paper shows how to obtain good utility with a group size smaller than 600. We obtain a mean relative error lower than 10% between the original data and the published one, which is considered practically suitable by energy experts for consumption forecasts.

The Laplace mechanism [14] is a popular mechanism to enable DP, by adding independent and identically distributed (IID) Laplace noise to each component of the time-series. However, adding IID noise for correlated time-series is not appropriate. In fact, an adversary can use refinement methods, such as filtering, to sanitize the IID noise and improve the probability of disclosing individual data [15, 16].

This paper focuses on disclosing an aggregate of users' consumption data without learning individual data and proposes methods with improved utility. We summarize our contributions as follows:

- (i) We revisit the Fourier perturbation algorithm (FPA) [17] in order to correct some mistakes leading to poor users' privacy protection. We show that, in order to ensure the desired budget of privacy ϵ , a factor $\sqrt{2T}$ must be added to the noise, where T is the size of the time-series. However, this reduces the utility of FPA.
- (ii) We propose the "clamping Fourier perturbation algorithm (CFPA)" using the clamping mechanism proposed in [18], for reducing the sensitivity, and thus the noise introduced in FPA. This new algorithm is an improvement of the Fourier perturbation algorithm (FPA). Experimental results show a utility improvement by a factor more than 6.
- (iii) We also propose the "clamping wavelet perturbation algorithm" (CWPA), a similar adaptation of wavelet perturbation algorithm (WPA) [19], with a utility improvement by a factor 2.
- (iv) We compare FPA, CFPA, WPA, and CWPA by analyzing their relative errors on a real dataset, and we explain why CFPA obtains the best utility.

The remainder of this paper is structured as follows. Section 2 provides an overview of the literature, while Section 3 presents preliminaries. Section 4 correctly computes the sensitivity of DFT in order to make FPA ϵ -differentially private. Section 5 details our privacy-preserving publication techniques using clamping mechanism, DFT, and DWT. Section 6 reports our experimental results. Section 7 concludes the paper.

Table 1 lists the acronyms used in this paper.

2. Related Work

Demand-response protocols [20–23], and secure aggregation protocols [24–30] aim to protect the privacy of users

TABLE 1: List of acronyms.

Acronym	Meaning
CFPA	Clamping Fourier perturbation algorithm
CWPA	Clamping wavelet perturbation algorithm
DFT	Discrete Fourier transform
DP	Differential privacy
DWT	Discrete wavelet transform
FPA	Fourier perturbation algorithm
MRE	Mean relative estimation error
WPA	Wavelet perturbation algorithm

while supporting energy services such as demand-response, smart metering, billing, or forecasting.

In this paper, we investigate tools enabling forecasting and demand-response. In particular, we are interested in publishing an aggregate of individual consumptions, while preserving privacy.

Differential privacy (DP), introduced by Dwork in 2006, guarantees that the publication of an aggregate does not indirectly reveal the individual data [12]. Moreover, DP guarantees that two aggregates that differ by a single consumer are almost indistinguishable. DP has evolved over-time [31] and was adopted by organizations such as the US Census Bureau [32], Google [33], Apple [34], and Microsoft [35]. The Laplace mechanism [14] is a popular mechanism that allows guaranteeing DP by adding a noise drawn from the Laplace distribution $\mathcal{L}(\cdot)$ to the aggregate.

The Laplace mechanism takes as input two parameters: the privacy budget ϵ and the sensitivity of the function to publish (in our case, the sum of users' consumption). Smaller values of ϵ lead to a better protection, but add a bigger noise to the aggregate.

Utility can be improved by increasing the size of the aggregate in order that the effect of noise is small enough that the result can be utilized. Eibl and Engel [13] showed that for real-world smart metering, the aggregation group size must be of the order of thousand smart meters in order to have reasonable utility. This paper shows how to obtain good utility with a group size smaller than a thousand.

DP is typically applied to static data, i.e., to a single query. In this paper, we consider time-series consumption, which is equivalent to multiple queries on correlated data. Applying the Laplace mechanism independently to each data point of the time-series is not appropriate. Indeed, an adversary can use refinement methods, such as filtering, to sanitize the Laplace noise and improve the probability disclosing individual data [15, 16]. Thus, the data points of the time-series are correlated. The composition theorem [14] states that the privacy budget ϵ of T correlated queries adds up, i.e., setting the privacy budget for a single query to $\epsilon_q = 0.5$, the privacy budget of $T = 48$ single queries (corresponding to a day profile with a time interval of 30 min) is $\epsilon = 0.5 \times 48 = 24$. In order to guarantee a global privacy budget of ϵ , one solution is to set the privacy budget of each query to ϵ/T . Of course, this leads to more noise in the aggregate and a loss of utility.

One method to guarantee DP for correlated time-series data publishing consists in transforming the original correlated time-series into another representation while

maintaining its major characteristics before adding the Laplace noise. Rastogi and Nath [17] proposed the Fourier perturbation algorithm (FPA) that combines discrete Fourier transform (DFT) with DP to support time-series of count queries while not disclosing any individual data and ensuring good utility. We note that the sensitivity of count queries is 1, and the global sensitivity is T for a time-series of length T . Ács et al. [36] proposed an optimization of the FPA allowing to release histograms, where the global sensitivity is 1. They show through experimental evaluation that their scheme improves the utility of the initial FPA by a factor 10. Lyu et al. [19] applied FPA to time-series consumptions and proposed wavelet perturbation algorithm (WPA) by replacing DFT by discrete wavelet transform (DWT). The authors show through experimental results that WPA ensures better utility than FPA.

We apply these approaches to time-series of consumption data and refine them by reducing the sensitivity of the queries in order to reduce the relative error of the final result.

3. Preliminaries

3.1. System and Threat Model. The entities involved in this paper are as follows:

- (i) Trustworthy homes, which smart meter (SM) enables to collect their true individual time-series consumption.
- (ii) A honest aggregator, which collects users' individual consumption, and which publishes an aggregate time-series consumption of users to a forecaster in a privacy-preserving way for the forecaster not to be able to deduce any individual consumption of users.
- (iii) A forecaster, which predicts future consumptions based on the aggregate consumption received in order to improve the grid and retail operations and enhance energy trading. The forecaster is considered honest-but-curious as it provides appropriate forecasts, but it may attempt to infer the users' individual consumption from the aggregate in order to deduce the behaviors, activities, age, or preferences of the inhabitants.

Figure 1 depicts the system model. In a real scenario, the aggregator can be an energy distributor, and the forecaster can be a municipality that seeks to find out the total consumption of the inhabitants of the municipality.

Considering the case where the aggregator and the forecaster belong to two entities of the same energy provider, the publication of aggregate users' consumption to forecasters in a privacy-preserving way reduces the risk of disclosing users' individual consumption. Moreover, this avoids the need for forecasters to ask for explicit consent from customers in accordance with the GDPR [37] to process their personal data.

Let N be the number of smart meters (SMs) in a district. Let $X^j = (x_1^j, x_2^j, \dots, x_T^j)$ be the time-series of energy

consumptions collected by SM j , where x_t^j is the consumption at time slot t ($t = 1, \dots, T$) collected by SM j ($j = 1, \dots, N$), with T being the time period considered. Each time-series consumption X^j is sent to an aggregator who computes the following aggregate:

$$S = (S_1, \dots, S_T) = \left(\sum_{j=1}^N x_1^j, \sum_{j=1}^N x_2^j, \dots, \sum_{j=1}^N x_T^j \right). \quad (1)$$

To reveal S to a forecaster without indirectly disclosing individual consumptions X^j ($j = 1, \dots, N$), the aggregator can use differential privacy (DP).

3.2. Differential Privacy. Differential privacy is a framework introduced by Dwork allowing quantifying the privacy guarantees of a request on a database [38]. This request can be the publication of a database, or a more precise one, such as "what is the sum of energy consumptions of users in this database?".

A request on databases is said to be differentially private if this request makes two similar databases indistinguishable from looking only at the output of the request. Differential privacy relies on a parameter, noted ϵ , called the privacy budget. The formal definition of a differentially private algorithm is given as follows.

Definition 1 (ϵ -differentially private). A request $\mathcal{A}: \mathcal{D} \rightarrow \mathcal{S}$ is ϵ -differentially private if and only if for all databases $D_1, D_2 \in \mathcal{D}$ differing by at most one record, and for all subsets $O \subset \mathcal{S}$,

$$\Pr(\mathcal{A}(D_1) \in O) \leq \exp(\epsilon) \Pr(\mathcal{A}(D_2) \in O). \quad (2)$$

This definition can be applied not only to requests on databases but also to any function, by considering the domain of the function as a database format.

Dwork also proposes the Laplace mechanism, which allows making any (vectors of) real-valued function ϵ -differentially private [38]. This mechanism relies on the notion of sensitivity of a function, which represents how a single record of the database can influence the output of the function.

Definition 2 (sensitivity). Let $f: \mathcal{D} \rightarrow \mathbb{R}^d$ be a function; the sensitivity of f is

$$\Delta_1(f) = \max_{D_1, D_2 \in \mathcal{D} \text{ s.t. } d(D_1, D_2) \leq 1} \|f(D_1) - f(D_2)\|_1. \quad (3)$$

This sensitivity is also called L_1 -sensitivity due to the L_1 -norm used in its definition and is denoted by $\Delta_1(f)$. Similarly, the L_2 -sensitivity used later and denoted by ϵ is computed using the L_2 -norm (the L_1 -norm and the L_2 norm of a vector $S = (s_1, \dots, s_T)$ are respectively equal to $\|S\|_1 = \sum_{j=1}^T |s_j|$ and $\|S\|_2 = \sqrt{\sum_{j=1}^T s_j^2}$).

The Laplace mechanism consists of adding a random value to the original result of the query, where the random

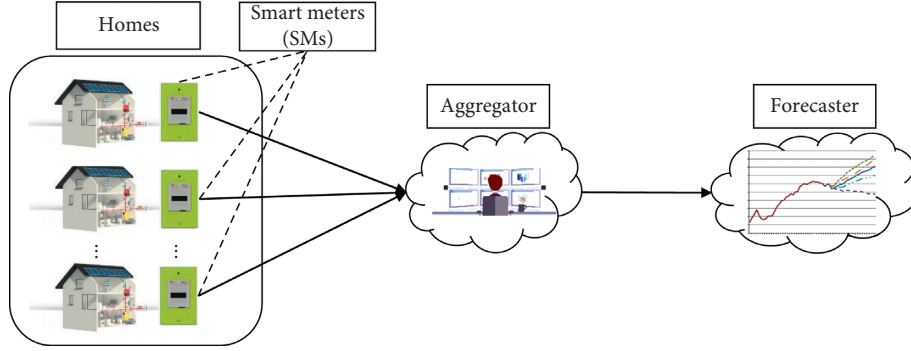


FIGURE 1: System model.

value follows the Laplace distribution, where the parameter depends on the chosen ϵ and on the sensitivity of the function, as follows.

Theorem 1 (Laplacian mechanism). *For all functions $f: \mathcal{D} \rightarrow \mathbb{R}^d$, the algorithm $\mathcal{A}(D) = f(D) + (\mathcal{L}_1(\Delta_1(f)/\epsilon), \dots, \mathcal{L}_d(\Delta_1(f)/\epsilon))$ is ϵ -differentially private, where $\mathcal{L}(\cdot)$ is the distribution of Laplace and $\Delta_1(f)$ is the sensitivity of f .*

DP introduces noise in order to guarantee privacy. This noise can decrease the utility of the function. We quantify this loss using mean relative estimation error (MRE), defined as follows.

Definition 3 (mean relative estimation error). The mean relative estimation error (MRE) between two vectors a and b of size T is $1/T \cdot \sum_{j=1}^T |a_j - b_j|/a_j + 12$ (we add 1 to the denominator in order to avoid dividing by zero. This definition is also used in [27]).

Consider the aggregate $S = (S_1, \dots, S_T)$ defined in (1). Let M be the maximum consumption in the domain. One naive solution to publish S without revealing any individual consumption is to use the Laplace mechanism to add independent Laplace noise to each component of S and to release the results: $\hat{S} = (S_1 + \mathcal{L}(M \cdot T/\epsilon), \dots, S_T + \mathcal{L}(M \cdot T/\epsilon))$, where the sensitivity of the sum of time-series consumption is $M \cdot T$. However, this simple approach leads to excessive noise rendering the aggregate useless [13].

Example 1. Figures 2(a) and 2(b), respectively, show the aggregated consumption of 250 homes from December 30th, 2009, to January 5th, 2010, taken from the CER dataset [39], and its noisy version using the naively applied Laplace mechanism, with $\epsilon = 1$ per day. Figure 2(a) shows two consumption peaks at 12 am and 6 pm which respectively correspond to lunch and dinner time. We also observe that in the night (from 12 pm to 6 am) the consumption decreases. Figure 2(b) shows that the noisy version is completely different from the original aggregate (Figure 2(a)). In this example, the MRE between the aggregate consumption and the noisy version is 141%, which is not usable.

Moreover, the noisy version has inconsistent values such as negative consumptions.

Rastogi and Nath [17] introduce the Fourier perturbation algorithm (FPA) and show that is an effective tool for reducing the noise introduced by the Laplace mechanism for time-series. Section 3.3 presents the FPA. However, there are some mistakes in this version relying on the estimation of the FPA sensitivity. These mistakes are presented in Section 4, along with the corrected FPA.

Table 2 lists the symbols used in the rest of the paper.

3.3. Fourier Perturbation Algorithm. The Fourier perturbation algorithm (FPA) presented in [17, 19, 36] takes as input a time-series $S = (S_1, \dots, S_T)$ and an integer $k \ll T$ and returns the noisy time-series $\hat{S} = (\hat{S}_1, \dots, \hat{S}_T)$, as shown in Algorithm 1.

Rastogi and Nath [17] show that FPA is ϵ -differentially private. However, there are some mistakes in their proof of Theorem 4.1 of [17] which justified that FPA is ϵ -differentially private. These mistakes rely on the estimation of the FPA sensitivity and are presented in Section 4.

3.4. Wavelet Perturbation Algorithm. By replacing the DFT with the discrete Haar wavelet transform (DWT), Lyu et al. [19] proposed the wavelet perturbation algorithm (WPA) and showed that WPA guarantees better utility than DFT. Algorithm 2 describes WPA.

Figure 3 shows the same aggregated consumption presented in Example 1 and its noisy version using WPA (Algorithm 2) with Haar wavelet, $\epsilon = 1$ per day and $k = 5$. In Figure 3, the MRE is however higher than 10% (18%). In the noisy aggregate, the first peak of the morning is masked and the peak of the evening is truncated, as well as the trough of the night.

Theorem 2. *Wavelet perturbation algorithm (WPA) is ϵ -differentially private.*

Proof. DWT is orthonormal [40], i.e., W has the same L_2 norm as S , that is, $\Delta_2(W) = \Delta_2(S)$. Furthermore, $\Delta_2(W^k) \leq \Delta_2(S)$ (because $T - k$ DWT coefficients of W are set to 0). With the inequality of norm, $\Delta_1(W^k) \leq \sqrt{k}\Delta_2(W^k)$. Then, $\Delta_1(W^k) \leq \sqrt{k}\Delta_2(S) \leq M \cdot \sqrt{kT}$. Thus, the noise

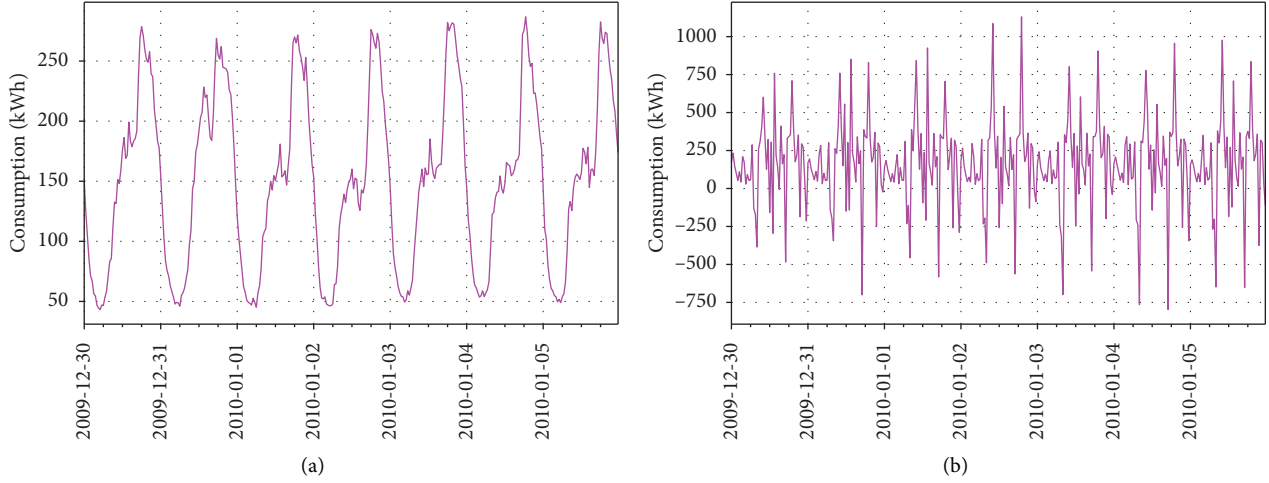


FIGURE 2: Aggregated time-series consumption of 250 homes from December 30th, 2009, to January 5th, 2010, taken from the CER dataset [39], and its noisy version using the naively applied Laplace mechanism, with $\epsilon = 1$ per day. (a) Aggregated consumption. (b) Noisy version using the naive solution.

TABLE 2: List of symbols.

Notation	Description
N	Number of smart meters (SMs) in the district
M	Maximum consumption in the dataset
T	Time period during the collection of time-series consumption
$X^j = (x_1^j, x_2^j, \dots, x_T^j)$	Time-series of energy consumptions collected by SM j , where x_t^j is the consumption at time slot t ($t = 1, \dots, T$) collected by SM j ($j = 1, \dots, N$)
$S = (S_1, \dots, S_T)$	Sum of users' time-series consumptions to be published, where $S_t = \sum_{j=1}^N x_t^j$ for $t = 1, \dots, T$
ϵ	Budget of privacy
\hat{S}	Noisy version of S
k	Number of the first DFT or DWT coefficients conserved in the Fourier perturbation algorithm (FPA), wavelet perturbation algorithm (WPA), clamping Fourier perturbation algorithm (CFPA), and clamping wavelet perturbation algorithm (CWPA)
Δ_1	L_1 -sensitivity
Δ_2	L_2 -sensitivity

Inputs: $S = (S_1, \dots, S_T)$, k , the maximum consumption M of the domain, and the privacy budget ϵ .

- (1) Compute the discrete Fourier transform of S : $F = \text{DFT}(S)$.
- (2) Keep only the first k coefficients of F , denoted by F^k .
- (3) Generate the noisy version of F^k , denoted by \tilde{F}^k by adding a Laplace noise $\mathcal{L}(M\sqrt{Tk}/\epsilon)$ to each coefficient in F^k .
- (4) Pad \tilde{F}^k to a T -dimensional vector, denoted by $\text{PAD}^T(\tilde{F}^k)$ by appending $T - k$ zeroes.
- (5) Apply the inverse DFT to $\text{PAD}^T(\tilde{F}^k)$ to obtain a noisy version of S denoted by \hat{S} .

ALGORITHM 1: Fourier perbutation algorithm [17].

Inputs: $S = (S_1, \dots, S_T)$, k , the maximum consumption M of the domain, and the privacy budget: ϵ

- (1) Compute the DWT coefficients of S : $W = \text{DWT}(S)$.
- (2) Keep only the first k coefficients of W , denoted by W^k .
- (3) Generate the noisy version of W^k , denoted by \tilde{W}^k by adding a Laplace noise $\mathcal{L}(M\sqrt{Tk}/\epsilon)$ to each coefficient in W^k .
- (4) Pad \tilde{W}^k to a T -dimensional vector, denoted by $\text{PAD}^T(\tilde{W}^k)$ by appending $T - k$ zeroes.
- (5) Apply the inverse DWT to $\text{PAD}^T(\tilde{W}^k)$ to obtain a noisy version of S denoted by \hat{S} .

ALGORITHM 2: Wavelet perbutation algorithm [19].

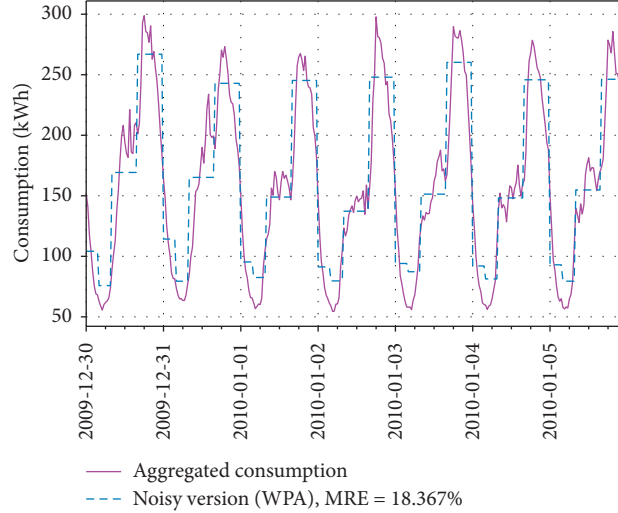


FIGURE 3: Aggregated consumption of 250 homes from 30th December 2009 to 5th January 2010 of dataset from CER [39] and its noisy version using WPA (Algorithm 2), with Haar wavelet, $\epsilon = 1$ per day and $k = 5$.

introduced in Step 3 is justified and WPA guarantees ϵ -differential privacy. \square

4. Correctly Estimating the Sensitivity of FPA

In [17], authors show that FPA, as described in Section 3, guarantees ϵ -differential privacy. The authors estimated the sensitivity of DFT to be $M\sqrt{Tk}$, while it should be $MT\sqrt{2k}$, with T being the size of the time-series and M being the maximum consumption in the domain. Thus, for a given privacy budget ϵ , the utility of FPA is worse than presented in [17].

This section correctly computes the sensitivity of DFT, which allows to make render FPA ϵ -differential private. Before that we recall the definition of DFT.

4.1. Discrete Fourier Transform (DFT). Let $S = (S_1, \dots, S_T)$ be a time-series. DFT takes S as input and returns a time-series of T complex numbers $F = (F_1, \dots, F_T)$ such that

$$F_k = \frac{1}{\sqrt{T}} \sum_{j=1}^T S_j e^{-2\pi i(j-1)(k-1)/T} \quad \text{for } k = 1, \dots, T, \quad (4)$$

where $i^2 = -1$. The inverse of the DFT is computed as follows:

$$S_k = \frac{1}{\sqrt{T}} \sum_{j=1}^T F_j e^{-2\pi i(j-1)(k-1)/T}, \quad \text{for } k = 1, \dots, T. \quad (5)$$

This version of the DFT is normalized, that is, $\|\text{DFT}(S)\|_2 = \|S\|_2$.

DFT can be defined in other ways, for instance, the $1/\sqrt{T}$, present in both the DFT and the inverse definitions above, can be replaced by a factor 1 in the DFT and $1/T$ in the inverse DFT. In that case, the DFT is not normalized.

In [17, 19, 36], the authors use the latter version of DFT, which is not normalized. However, the sensitivity computation relies on the equality $\|\text{DFT}(S)\|_2 = \|S\|_2$, while it should be $\|\text{DFT}(S)\|_2 = \sqrt{T} \cdot \|S\|_2$. Thus, the correct total privacy budget is $\sqrt{T} \cdot \epsilon$ instead of ϵ . This is the first mistake in this approach and can be resolved by using the normalized DFT.

Another error lies in the fact that the Laplacian mechanism is only applied to the real part of the Fourier coefficients, which are complex numbers. This mistake can be resolved by applying the Laplace mechanism to both real and imaginary parts of the Fourier coefficients.

The following section computes the sensitivity of the DFT, and thus of FPA, and takes into account those two errors.

4.2. Sensitivity of the DFT. Let DFT^k be the function which takes a time-series $S = (S_1, \dots, S_T)$ as input and returns the first k DFT coefficients of S . This function can be seen as a $\text{DFT}^k: \mathbb{R}^T \rightarrow \mathbb{R}^{2k}$, the function which returns the real and imaginary parts of the first k Fourier coefficients. This function is a real-valued function, we can thus use the Laplace mechanism on it. First, we need to compute the L_1 -sensitivity of DFT^k .

Lemma 1. Let DFT^k be defined as follows:

$$\begin{aligned} \text{DFT}^k: \mathbb{R}^T &\longrightarrow (\mathbb{R}^2)^k \\ S &\longmapsto \text{DFT}^k(S) = ((a_1, b_1), \dots, (a_k, b_k)). \end{aligned} \quad (6)$$

We denote $c_j = a_j + ib_j$ the j -th coefficient of $\text{DFT}(S)$, with $i^2 = -1$ and $j = 1, \dots, k$. a_j and b_j respectively represent the real and imaginary parts of c_j .

The L_1 -sensitivity of DFT^k , $\|\text{DFT}^k(S)\|_1$, is $M \cdot \sqrt{2Tk}$ when the DFT is normalized (respectively, $MT \cdot \sqrt{2k}$ when

the DFT is not normalized), with M as the maximum value in the dataset.

Proof. Let DFT^k be defined as in Lemma 1.

$$\begin{aligned}
\|\text{DFT}^k(S)\|_1 &= \|(a_1, b_1, \dots, a_k, b_k)\|_1 \\
&= \sum_{j=1}^k \|(a_j, b_j)\|_1 \leq \sqrt{2} \sum_{j=1}^k \|(a_j, b_j)\|_2 \text{ (Minkowski inequality)} \\
&\leq \sqrt{2} \sum_{j=1}^k |c_j| \\
&\leq \sqrt{2} \|(c_1, \dots, c_k)\|_1 \\
&\leq \sqrt{2} \sqrt{k} \|(c_1, \dots, c_k)\|_2 \text{ (Minkowski inequality)} \\
&\leq \sqrt{2k} \|(s_1, \dots, s_T)\|_2 \text{ as } T \text{ Fourier coefficients have the same } L_2 \text{ norm as } S.
\end{aligned} \tag{7}$$

Then,

$$\Delta_1(\text{DFT}^k) \leq \sqrt{2k} \Delta_2(S) = M \sqrt{2Tk}.$$

This result is true when the DFT is normalized (2) as in our case. In [17, 19, 36], the L_2 norm of Fourier coefficients equals to \sqrt{T} times the L_2 norm of S (Parvesal's theorem). This result is valid when the normalized DFT (2) is used as in our case. When the DFT is not normalized, as is the case in [17, 19, 36], the sensitivity of the first k DFT coefficients should be $\Delta_1(\text{DFT}^k) = \sqrt{T} \times M \sqrt{2Tk} = MT \sqrt{2k}$ instead of $(M \sqrt{Tk})$. Thus, using the normalized DFT, the function then becomes

$$\begin{aligned}
\widetilde{\text{DFT}}^k: \mathbb{R}^T &\longrightarrow (\mathbb{R}^2)^k \\
S &\longmapsto \widetilde{\text{DFT}}(S) = ((a_1, b_1), \dots, (a_k, b_k)) + ((y_{1,1}, y_{1,2}), \dots, (y_{k,1}, y_{k,2})),
\end{aligned} \tag{8}$$

which is ε -DP, with $y_{j,\ell} = \mathcal{L}(M \sqrt{2Tk}/\varepsilon)$, for all $j = 1, \dots, k$ and $\ell = 1, 2$.

For simplicity, in the following, we write $c_j + \mathcal{L}(M \sqrt{2Tk}/\varepsilon)$ instead of $(a_j, b_j) + (\mathcal{L}(M \sqrt{2Tk}/\varepsilon), \mathcal{L}(M \sqrt{2Tk}/\varepsilon))$, meaning that two independent Laplace noises $\mathcal{L}(M \sqrt{2Tk}/\varepsilon)$ are added to the real and imaginary parts of c_j .

Algorithm 3 shows the Fourier perturbation algorithm (FPA) revisited.

4.3. Differences between the Initial, yet Incorrect, FPA, and the Corrected FPA. For a budget of privacy ε , the differences between the initial incorrect FPA and the corrected one can be highlighted as follows:

- (1) The DFT used in the initial incorrect FPA [17] is not normalized, while it is normalized in the corrected FPA. Thus, a factor $\sqrt{2T}$ is missing in the Laplace noise in Algorithm 1.
- (2) In the initial incorrect FPA [17], Laplace noises are only added to the real part of the DFT coefficients,

while they should be added to the real and imaginary parts of the DFT coefficients as in the corrected FPA (Algorithm 3). Thus, k imaginary coefficients are not noised in Algorithm 1.

Figure 4 shows the same aggregated consumption presented in Example 1 and its noisy version using the corrected FPA (Algorithm 3) with $\varepsilon = 1$ per day and $k = 5$. Figure 4 shows that the corrected FPA obtains a large MRE (84%), making it useless. The noisy aggregate has negative consumptions and does not contain the peaks present in the original aggregate.

For the sake of simplicity, in the following sections, we use FPA to talk about the corrected version.

5. Clamping Transform Perturbation Algorithm

The intuition behind our approach, ‘‘Clamping transform perturbation algorithm,’’ lies in the perturbation error, caused by the Laplace mechanism, which depends on the sensitivity of the sum of consumptions. As such, by reducing the sensitivity, we expect to reduce the perturbation error.

To estimate the sensitivity of consumptions, we split our database of N users into two almost equal parts: D_1 corresponding to the consumptions of the first half of users (a training dataset) and D_2 containing the second half of users' consumptions (a validation dataset). Using D_1 , we compute the distribution of users' consumptions in the frequency domain. We denote by $M = (M_1, \dots, M_k)$ the maximum magnitude (by ignoring outliers) of the k first coefficients.

For example, using the Irish consumption database [39], the distribution of the individual consumption of the first half customers (from 1 to 1818) in the frequency domain is given in Figure 5. In Figure 5, the maximum magnitudes (rounded) of the 5 first coefficients are $M = (M_1, M_2, M_3, M_4, M_5) = (9, 4, 3, 2, 2)$.

Inputs: $S = (S_1, \dots, S_T)$, k , the maximum consumption M of the domain, and the privacy budget ϵ .

- (1) Compute the normalized DFT coefficients of $S: F = \text{DFT}(S)$.
- (2) Keep only the first k coefficients of F , denoted by F^k .
- (3) Generate the noisy version of F^k , denoted by \tilde{F}^k by adding a Laplace noise $\mathcal{L}(M\sqrt{2Tk}/\epsilon)$ to each coefficient in F^k .
- (4) Pad \tilde{F}^k to a T -dimensional vector, denoted by $\text{PAD}^T(\tilde{F}^k)$ by appending $T - k$ zeroes.
- (5) Apply the inverse DFT to $\text{PAD}^T(\tilde{F}^k)$ to obtain a noisy version of S denoted by \hat{S} .

ALGORITHM 3: Fourier perturbation algorithm (FPA) revisited.

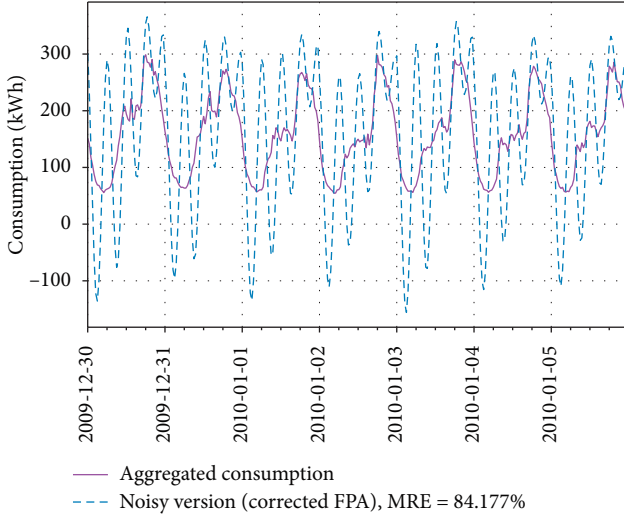


FIGURE 4: Aggregated consumption of 250 homes from 30th December 2009 to 5th January 2010 of dataset from CER [39] and its noisy version using the corrected FPA (Algorithm 3), with $\epsilon = 1$ per day and $k = 5$.

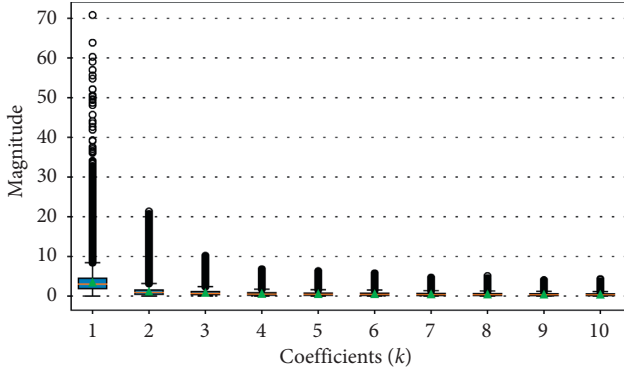


FIGURE 5: Distribution of users' consumptions in the frequency domain using the discrete Fourier transform (DFT).

The database D_2 is used for testing our methodology. Let $X = (X^1, \dots, X^n)$ with $X^j = (x_1^j, \dots, x_T^j)$ for all $j = 1, \dots, n$ be the users' individual consumptions. To publish the sum of consumptions $S = (S_1, \dots, S_T) = (\sum_{j=1}^n x_1^j, \dots, \sum_{j=1}^n x_T^j)$, our methodology, which can be applied to either the Fourier transform or to wavelet transforms, is described as follows:

- (1) For all individual consumptions X^j ($j = 1, \dots, n$), compute the corresponding magnitude in the

domain of the transform and keep the first k coefficients denoted by $C^j = (C_1^j, \dots, C_k^j)$.

- (2) If the modulus of coefficient C_ℓ^j is greater than M_ℓ ($1 \leq \ell \leq k$), replace C_ℓ^j with $C_\ell^j \cdot M_\ell / |C_\ell^j|$ so that all coefficients have a modulus smaller than M_ℓ and their phase, if the coefficient is complex, is unchanged.
- (3) Compute the sum of coefficients $C = (\sum_{j=1}^n C_1^j, \dots, \sum_{j=1}^n C_k^j)$.
- (4) Add a noise following the distribution of Laplace $\mathcal{L}(\cdot)$, depending on the sensitivity of the transform, to each coefficient C_ℓ ($1 \leq \ell \leq k$) of C . The result is denoted by \hat{C} . We note that the Laplace noise is added to the real and imaginary parts of each coefficient when the DFT is used.
- (5) Pad the vector \hat{C} by $n - k$ zeroes and compute the inverse transform to obtain the noisy version of the consumption \hat{S} .

Section 5.1 presents an adaptation of this methodology using the discrete Fourier transform.

5.1. Clamping Fourier Perturbation Algorithm. This section describes the clamping Fourier perturbation algorithm (CFPA) detailed in Algorithm 4. This algorithm allows an aggregator to compute and publish an aggregate guaranteeing ϵ -differential privacy.

CFPA takes as inputs the individual time-series consumptions of n consumers, the maximum magnitudes of DFT coefficients of individual consumptions M (computed over database D_1), the number k of DFT coefficients to be considered, and the privacy budget ϵ , and it returns the noisy time-series sum of consumptions of n consumers.

Step 1, called clamping, computes the first k DFT coefficients of each individual time-series consumption. If the magnitude of a coefficient F_ℓ^j is greater than the maximum magnitude M_ℓ , then this coefficient is clamped and replaced by $F_\ell^j \cdot M_\ell / |F_\ell^j|$, in which magnitude is $|F_\ell^j| \cdot M_\ell / |F_\ell^j| = M_\ell$. Thus, for all individual consumptions X^j , the maximum magnitude of the k first DFT coefficients $F^j = (F_1^j, \dots, F_k^j)$ is $M = (M_1, \dots, M_k)$, i.e., the final values of the coefficients have the same phase as the initial values, but their magnitudes are bounded by (M_1, \dots, M_k) .

After computing the first k DFT coefficients $F^j = (F_1^j, \dots, F_k^j)$ of each individual time-series consumption of consumers ($j = 1, \dots, n$), Step 2 consists in computing the sum $(F_1, \dots, F_k) = (\sum_{j=1}^n F_1^j, \dots, \sum_{j=1}^n F_k^j)$ of

Inputs:

- (i) Consumptions: $X = (X^1, \dots, X^n)$ with $X^i = (x_i^1, \dots, x_i^T)$ for all $i = 1, \dots, n$
- (ii) k
- (iii) The maximum magnitudes of k first DFT coefficients: $M = (M_1, \dots, M_k) \in \mathbb{R}_+^k$
- (iv) Privacy budget: ε
- (1) *Clamping*: for each individual time-series consumption X^j ,
 - (i) compute the k first DFT coefficients of X^j : $F^j = (\text{DFT}(X^j)_1, \dots, \text{DFT}(X^j)_k) = (F_1^j, \dots, F_k^j)$
 - (ii) if $|F_\ell^j| > M_\ell$, then replace F_ℓ^j with $F_\ell^j \cdot M_\ell / |F_\ell^j|$ for all $\ell = 1, \dots, k$
- (2) *Laplacian mechanism*: compute the sum of noisy consumptions of each DWT coefficient: $\widehat{F}_\ell = \sum_{j=1}^n F_\ell^j + \mathcal{L}(M_\ell \sqrt{2}/\varepsilon/k)$ for all $\ell = 1, \dots, k$. We denote $\widehat{F}^k = (\widehat{F}_1, \dots, \widehat{F}_k)$. We note that the noise is added to the real and imaginary parts of the sum of coefficients.
- (3) Pad \widehat{F}^k with $T - k$ zeros; the result is denoted by $\text{PAD}^T(\widehat{F}^k)$
- (4) Compute the inverse DFT of $\text{PAD}^T(\widehat{F}^k)$ to get the noisy sum of consumptions denoted by $\widehat{S} = (\widehat{S}_1, \dots, \widehat{S}_T)$ of the initial sum $S = (\sum_{j=1}^n x_1^j, \dots, \sum_{j=1}^n x_T^j)$.

ALGORITHM 4: Clamping Fourier perturbation algorithm (CFPA).

these coefficients using the Laplacian mechanism. The result is denoted by $\widehat{F}^k = (\widehat{F}_1, \dots, \widehat{F}_k)$.

Finally, the noisy sum of consumptions is equal to the inverse of the noisy DFT coefficients padded with $T - k$ zeros.

Theorem 3. *Algorithm CFPA is ε -differentially private.*

Proof. To prove that Algorithm 4 is ε -differentially private, we need to prove that the sensitivity of the sum of DFT coefficients of users' individual consumptions F_1 (resp. F_2, \dots, F_k) is $\sqrt{2} \cdot M_1$ (resp. $\sqrt{2} \cdot M_2, \dots, \sqrt{2} \cdot M_k$). This is done in Lemma 2.

Then, as a Laplacian noise $\mathcal{L}(M_\ell \sqrt{2}/\varepsilon/k)$ is added to each component F_ℓ ($\ell = 1, \dots, k$), the resulting \widehat{F}_1 (resp. $\widehat{F}_2, \dots, \widehat{F}_k$) is ε/k -differentially private. Finally, the composition theorem [14] guarantees that any computation on the k components of $(\widehat{F}_1, \dots, \widehat{F}_k)$ is ε -differentially private; thus, the inverse DFT of those coefficients is ε -DP. \square

Lemma 2. *Let $F^j = (\text{DFT}(X^j)_1, \dots, \text{DFT}(X^j)_k) = (F_1^j, \dots, F_k^j)$ be the first k DFT coefficients of the individual consumption of consumer j ($j = 1, \dots, n$), obtained after the clamping mechanism. The sensitivity of the sum of each DFT coefficient F_ℓ^j ($\ell = 1, \dots, k$) of n consumers' individual consumptions is $M_\ell \cdot \sqrt{2}$.*

Proof. Let $F^j = (\text{DFT}(X^j)_1, \dots, \text{DFT}(X^j)_k) = (F_1^j, \dots, F_k^j)$. After the clamping, the magnitude of each DFT coefficient F_ℓ^j is smaller than M_ℓ for $\ell = 1, \dots, k$, and the sensitivity of the function $f_\ell: \mathcal{D}^n \rightarrow \mathbb{C} \equiv \mathbb{R}^2$ defined by $f_\ell: (X^1, \dots, X^n) \mapsto \sum_{j=1}^n F_\ell^j \equiv (\sum_{j=1}^n a_\ell^j, \sum_{j=1}^n b_\ell^j)$, with $F_\ell^j = a_\ell^j + ib_\ell^j$ being equal to

$$\begin{aligned}
 \Delta_1(f_\ell) &= \max_{|a_\ell^j|, |b_\ell^j|} \left\| \left(\sum_{j=1}^n a_\ell^j, \sum_{j=1}^n b_\ell^j \right) - \left(\sum_{j=2}^n a_\ell^j, \sum_{j=2}^n b_\ell^j \right) \right\|_1 \\
 &= \max \left\| (a_\ell^1, b_\ell^1) \right\|_1 \\
 &\leq \max \sqrt{2} \left\| (a_\ell^1, b_\ell^1) \right\|_2 \quad (\text{Minkowski inequality}) \\
 &= \max \sqrt{2} \sqrt{(a_\ell^1)^2 + (b_\ell^1)^2} \\
 &= \max \sqrt{2} |F_\ell^1| \\
 &= M_\ell \sqrt{2}.
 \end{aligned} \tag{9}$$

Thus, Lemma 2 proves Theorem 3, and algorithm CFPA guarantees ε -differential privacy.

For example, Figure 6 shows the same aggregated consumption presented in Example 1 and its noisy version using CFPA (Algorithm 4) with $\varepsilon = 1$ per day and $k = 5$. Figure 6 shows that CFPA obtains a good utility with an MRE equal to 9.7%. This good utility of CFPA can be explained by the fact that Laplace noise added in CFPA depends on the amplitude of each coefficient, while in FPA, the same noise $\mathcal{L}(M \cdot \sqrt{2Tk}/\varepsilon)$ is added to every DFT coefficients, where M is the maximum consumption in the dataset.

5.2. Clamping Wavelet Perturbation Algorithm. The clamping wavelet perturbation algorithm (CWPA), as presented in Algorithm 5, is obtained by replacing DFT with DWT in Algorithm 4. The computation of DWT is based on multiresolution analysis which determines the number of

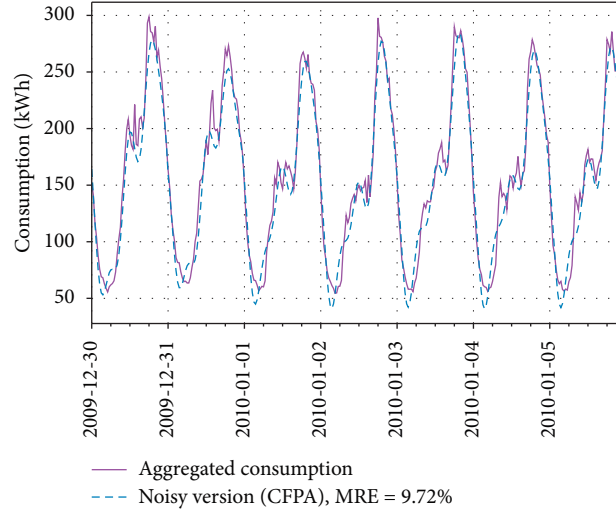


FIGURE 6: Aggregated consumption of 250 homes from 30th December 2009 to 5th January 2010 of dataset from CER [39] and its noisy version using CFPA (Algorithm 4), with $\varepsilon = 1$ per day and $k = 5$.

approximation coefficients (scaling functions) and detail coefficients (wavelet functions) [40]. DWT takes as input a time-series of length a power of 2. If the input's length is not a power of 2, we can pad it with zeroes [41].

Algorithm 5 takes as inputs the maximum magnitudes of the first k DWT coefficients which are obtained in the training process on D_1 , by computing the distribution of DWT coefficients of individual consumptions.

We note that there are multiple DWTs, such as Haar, Daubechies, Symlets, and Coiflets. In this paper, we use Haar and Daubechies wavelets as shown in Section 6, because they give a low reconstruction error, as will be discussed in Section 6.1.

Theorem 4. *The clamping wavelet perturbation algorithm (CWPA), Algorithm 5, is ε -differentially private.*

Proof. The proof is similar to the one for Theorem 3. We need to prove that the sensitivity of the sum of DWT coefficients of users' individual consumptions W_1 (resp. W_2, \dots, W_k) is M_1 (resp. M_2, \dots, M_k). This is done in Lemma 3.

Then, as a Laplacian noise $\mathcal{L}(M_\ell/\varepsilon/k)$ is added to each component W_ℓ ($\ell = 1, \dots, k$), the resulting \widehat{W}_1 (resp. $\widehat{W}_2, \dots, \widehat{W}_k$) is ε/k -differentially private. Finally, the composition theorem [14] guarantees that any computation on the k components $(\widehat{W}_1, \dots, \widehat{W}_k)$ is ε -differentially private; thus, the inverse DWT of those coefficients is ε -DP. \square

Lemma 3. *Let $W^j = (DWT(X^j)_1, \dots, DWT(X^j)_k) = (W^j_1, \dots, W^j_k)$ be the first k DWT coefficients of the individual consumption of consumer j ($j = 1, \dots, n$), obtained after the clamping mechanism. The sensitivity of the sum of each DWT coefficient W^j_ℓ ($\ell = 1, \dots, k$) of n consumers' individual consumptions is M_ℓ .*

Proof. Let $W^j = (DWT(X^j)_1, \dots, DWT(X^j)_k) = (W^j_1, \dots, W^j_k)$. After the clamping, the magnitude of each DWT

coefficient W^j_ℓ is smaller than M_ℓ for $\ell = 1, \dots, k$, and the sensitivity of the function $w_\ell: \mathcal{D}^n \rightarrow \mathbb{R}$ defined by $w_\ell: (X^1, \dots, X^n) \mapsto \sum_{j=1}^n W^j_\ell$ is equal to

$$\begin{aligned} \Delta_1(w_\ell) &= \max_{|W^j_\ell|} \left| \sum_{j=1}^n W^j_\ell - \sum_{j=2}^n W^j_\ell \right| \\ &= \max |W^1_\ell| \\ &= M_\ell. \end{aligned} \quad (10)$$

\square

For example, Figure 7 shows the same aggregated consumption presented in Example 1 and its noisy version using CWPA (Algorithm 5) with Haar wavelet, $\varepsilon = 1$ per day and $k = 5$. However, Figure 3 shows that the MRE of CWPA is still higher than 10%. We explain this result in Section 6.

6. Experimental Results

This section compares FPA, CFPA, WPA, and CWPA and explains through experimentations why CFPA achieves a better utility than other publication techniques. After presenting the raw results, we explain them by decomposing the mean relative error into a perturbation error, caused by the clamping mechanism and the Laplace mechanism, and a reconstruction error, due to ignoring $T - k$ coefficients of the transform. The analysis of the error is thus conducted in the next two Subsections 6.1 and 6.2. Section 6.1 analyzes the reconstruction error, while Section 6.2 analyzes the perturbation one.

Conditions: the experiments rely on data originating from the Irish Commission for Energy Regulation (CER) [39]. This dataset contains real time-series consumptions. The achieved results are valid for this very specific case, for Irish consumptions with an Irish weather being never too hot or too cold. The results show that the approach is good, but will probably have to be adapted for other datasets, i.e., by computing the maximum magnitudes of the k first coefficients of the

Inputs:

- (i) Consumptions: $X = (X^1, \dots, X^n)$ with $X^j = (x_1^j, \dots, x_T^j)$ for all $j = 1, \dots, n$
- (ii) k
- (iii) The maximum magnitudes of k first DWT coefficients: $M = (M_1, \dots, M_k) \in \mathbb{R}_+^k$
- (iv) Privacy budget: ϵ
- (1) *Clamping*: for each individual time-series consumption X^j ,
 - (i) compute the first k DWT coefficients of X^j : $W^j = (\text{DWT}(X^j)_1, \dots, \text{DWT}(X^j)_k) = (W_1^j, \dots, W_k^j)$
 - (ii) if $|W_\ell^j| > M_\ell$, then replace W_ℓ^j with $W_\ell^j \cdot M_\ell / |W_\ell^j|$ for $\ell = 1, \dots, k$
- (2) Laplacian Mechanism: compute the sum of noisy consumptions of each DWT coefficient: $\widehat{W}_\ell = \sum_{j=1}^n W_\ell^j + \mathcal{L}(M_\ell/\epsilon/k)$ for all $\ell = 1, \dots, k$. We denote $\widehat{W}^k = (\widehat{W}_1, \dots, \widehat{W}_k)$
- (3) Pad \widehat{W}^k with $T - k$ zeroes; the result is denoted by $\text{PAD}^T(\widehat{W}^k)$
- (4) Compute the inverse DWT of $\text{PAD}^T(\widehat{W}^k)$ to get the noisy sum of consumptions denoted by $\widehat{S} = (\widehat{S}_1, \dots, \widehat{S}_T)$ of the initial sum $S = (\sum_{j=1}^n x_1^j, \dots, \sum_{j=1}^n x_T^j)$

ALGORITHM 5: Clamping wavelet perturbation algorithm (CWPA).

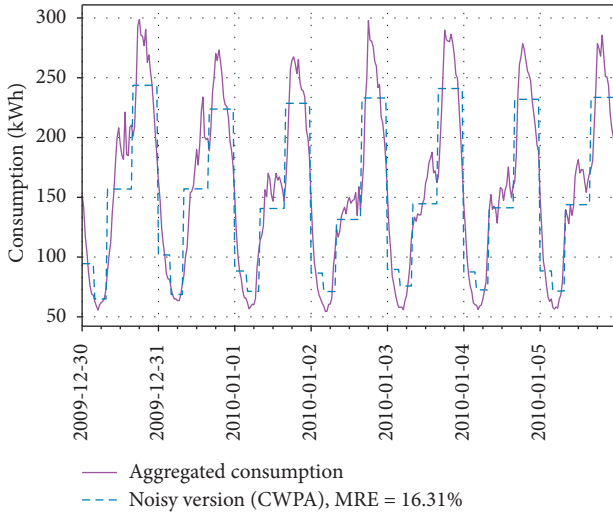


FIGURE 7: Aggregated consumption of 250 homes from 30th December 2009 to 5th January 2010 of dataset from CER [39] and its noisy version using CWPA (Algorithm 5), with Haar wavelet, $\epsilon = 1$ per day and $k = 5$.

considered transform over a subpart of the dataset. Consumption data from the CER were collected every 30 minutes from 2009 to 2010 with the participation of more than 5,000 Irish homes and businesses. This experiment only considers homes. We divided the database in two parts: D_1 , corresponding to the first half of consumers (1 to 1818), and D_2 , corresponding to the second half (1819 to 3639). D_1 is used to calibrate the algorithms by computing the maximum magnitudes $M = (M_1, \dots, M_k)$ of the first k coefficients in the frequency domain, and D_2 is used to test the publication techniques FPA, CFPA, WPA, and CWPA.

Notations: we note N as the number of homes or smart meters considered in the district to compute the time-series of the sum of consumptions. For each day (48 time slots), we compute the sum of consumptions of 50 different districts of N random homes, and we execute FPA, WPA, CFPA, and CWPA with privacy budget $\epsilon \in \{1, 3\}$ for each day and $k \in \{5, 8, 12\}$. The discrete wavelet transforms used here are Haar transform (which

represents the same wavelet as Daubechies with order 1, noted db1), Daubechies with order 2, and Daubechies with order 3, respectively, noted db2 and db3.

Raw results and analysis: Figures 8 and 9 show the distribution of the mean relative estimation error (MRE) according to the number of homes in the district (N from 50 to 450) and k from 5 to 12 for the budget of privacy $\epsilon = 1$ and $\epsilon = 3$, respectively. The boxes extend from the lower to upper quartile values of the MRE, with a line at the median and a triangle representing the mean. The whiskers extend from the box to show the range of the MRE. In order to make consumption forecasts, an MRE lower than 10% is required in practice by experts in the energy sector. In this section, an MRE of less than 10% is therefore considered useful.

In Figures 8 and 9, the first column corresponds to the comparison between the FPA and the CFPA. The other columns correspond to the comparison between the WPA and the CWPA using Haar wavelet with 2 approximation coefficients, Daubechies 2 (db2) with 5 approximation coefficients, and Daubechies 3 (db3) with 10 approximation coefficients, respectively.

Figures 8 and 9 show that CFPA has a better utility than FPA. For example, for $\epsilon = 1$ (Figure 8), when $k = 5$ and the number of homes $N = 350$, the MRE of CFPA is 12%, while the MRE of FPA is 75%. In that configuration, the MRE of CFPA is 6.25 times lower than that of FPA. Similarly, the CWPA obtains a better utility than the WPA. For example, for $k = 5$ and the number of homes $N = 350$, the MRE of CWPA using Haar wavelet is 15%, while the MRE of the WPA is 30%. In that configuration, the MRE of CWPA is 2 times lower than that of WPA.

Generally, the larger the size of the district N , the smaller the MRE is. Similarly, the larger the budget of privacy ϵ , the smaller the MRE is; Figure 9 ($\epsilon = 3$) shows a better utility than Figure 8 ($\epsilon = 1$). However, Figures 8 and 9 show that WPA and CWPA using db3 are not useful for $k = 5$ because, as shown in Section 6.1, the reconstruction error is high (between 70% and 80%).

Figures 8 and 9 show that for larger k , the MRE of WPA and CWPA using db3 is smaller. Moreover, in Figure 9, for

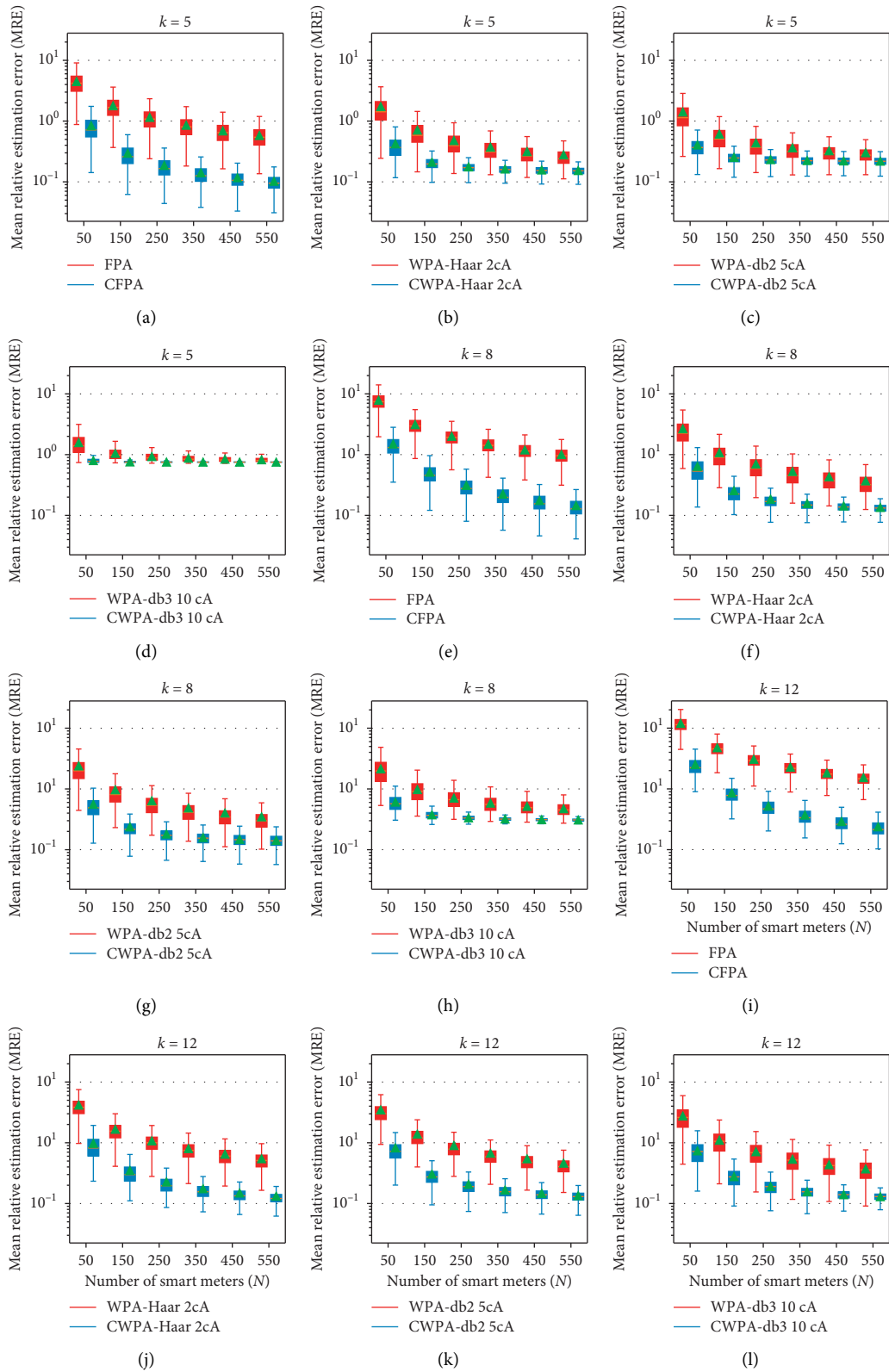


FIGURE 8: Mean relative estimation error (MRE) of FPA vs CFPA vs WPA vs CWPA, using DFT and DWT with Haar, Daubechies 2 (db2), and Daubechies 3 (db3) wavelets, according to k , and the number of smart meters (N) in the district, with $\varepsilon = 1$.

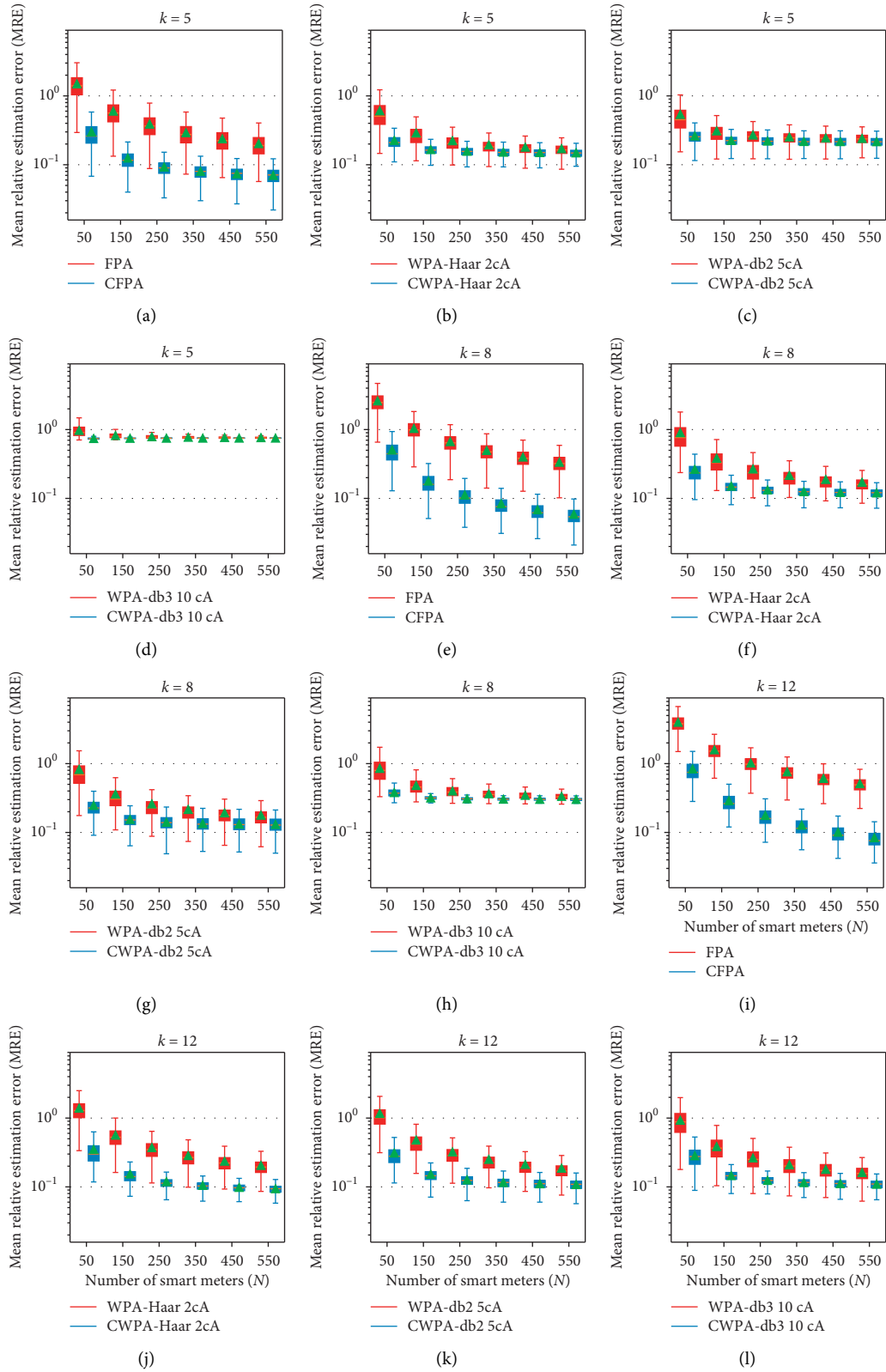


FIGURE 9: Mean relative estimation error (MRE) of FPA vs CWPA vs WPA vs CWPA with Haar, Daubechies 2 (db2), and Daubechies 3 (db3) wavelets, according to k , and the number of smart meters (N) in the district, with $\epsilon = 3$.

$k = 12$ and when the number of homes is higher than 250 and $\varepsilon = 3$, CWPA using db3 has the median of MRE smaller than 11%. The CWPA using Haar wavelet obtains the second best utility, with the median of MRE smaller than 10% when N is greater than 250, while the CFPA gets the best utility, with the median of MRE decreasing to 5% when $N = 550$ and $k = 8$.

However, the utility of FPA and WPA decreases when k increases. This is caused by the perturbation error; indeed, the greater the k , the greater the Laplacian noise added to each coefficient is. This noise is attenuated by the clamping as shown by the CFPA. Indeed, when k goes from 5 to 8, the reconstruction error decreases and the clamping also decreases the perturbation error leading to the total error reduction. However, when k goes from 8 to 12, although the reconstruction error decreases, clamping does not reduce the perturbation error sufficiently. This explains why the MRE of CFPA is a little bigger when $k = 12$ compared to $k = 8$.

In Figures 8 and 9, we notice that the median of MRE of WPA and CWPA converge to a threshold and never goes below it. For example, for $\varepsilon = 3$ and $k = 5$, the median of MRE of WPA and CWPA using db2 converges to 23%. This is caused by the reconstruction error.

6.1. Reconstruction Error. The reconstruction error is due to considering only the k first transform coefficients, thus removing the precision brought by coefficients $(k + 1, k + 2, \dots)$. To measure this error, a first solution consists in computing the cumulative distribution function (CDF) of the coefficients as a first assessment of the impact of the transform coefficients and, then, to get confirmation through some experimental reconstruction error measurements. Intuitively, if the CDF of some coefficients k is close to 1, it means that the coefficients after k ($k + 1, k + 2, \dots$) have less impact on the reconstruction, and thus, when set to zero, lead to a smaller reconstruction error.

The CDF is computed for a district of 50 homes of several transformations: discrete Fourier transform (DFT), discrete wavelet transform (DWT) using Haar, and Daubechies 2 and Daubechies 3 wavelets. The closer to 1 the cumulative distribution function at k is, the smaller the reconstruction error is. Figure 10 compares the cumulative distribution function of DFT and DWT with different wavelet transforms. This figure shows that DFT has a higher cumulative distribution than DWT for the considered range value of k ($k \leq 10$).

In order to analyze this error more precisely, we define formally the reconstruction error below, and we then compute it experimentally.

Definition 4 (reconstruction error). Let $S = (S_1, S_2, \dots, S_T)$ be a sum of time-series consumptions and $C = (C_1, C_2, \dots, C_T)$ be the coefficients in the frequency domain of this time-series. We denote $\text{PAD}^T(C^k) = (C_1, \dots, C_k, 0, \dots, 0)$ as the first k coefficients padded with zeros and $\tilde{S} = (\tilde{S}_1, \tilde{S}_2, \dots, \tilde{S}_T)$ as the inverse of $\text{PAD}^T(C^k)$ (in the time domain). The reconstruction error

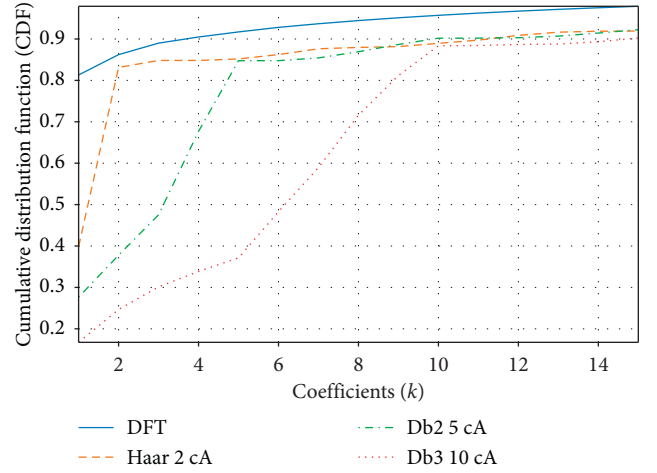


FIGURE 10: Comparison of cumulative distribution function of DFT and DWT with Haar, and Daubechies 2 and 3 wavelets for a district of 50 homes.

(RE) of $\text{PAD}^T(C^k)$ is equal to the mean relative estimation error between S and \tilde{S} given by (we add 1 to the denominator in order to avoid the division by zero)

$$\text{RE}(\text{PAD}^T(C^k)) = \frac{1}{T} \cdot \sum_{j=1}^T \frac{|S_j - \tilde{S}_j|}{S_j + 1}. \quad (11)$$

Figure 11 shows the reconstruction error for DFT and DWT with different wavelet transforms for a district of 50 and 450 homes. This figure shows that the DFT obtains the smallest relative error (lower than 10% when k is greater than 5) followed by Haar and Daubechies. We note that the reconstruction error of Daubechies 2 is higher than 23% when $k = 5$, which leads to a total error higher than 23% and justifies the relative error obtained in Figures 8 and 9.

Moreover, when $k = 5$, the reconstruction error of Daubechies 3 is higher than 70%, which justifies why its total error is higher than 70% when $k = 5$, according to Figures 8 and 9.

According to the database from the Irish Commission for Energy Regulation (CER) [39], the discrete Fourier transform gets the smaller reconstruction error, followed respectively by Haar (which is the same as Daubechies 1) and Daubechies 2 and Daubechies 3 wavelets.

6.2. Perturbation Error. The perturbation error is caused by the Laplace mechanism, applied on the first k transform coefficients. The higher the transform coefficients, the lower the impact of this perturbation in terms of relative error, and thus the lower the perturbation error.

We note that the amplitude of the Laplace noise introduced by the Laplace mechanism is different for CFPA and CWPA; it is $\sqrt{2}$ times greater for CFPA than for CWPA. Indeed, for all $\ell = 1, \dots, k$, the parameter for the Laplace noise is $\mathcal{L}(M_\ell \sqrt{2}/\varepsilon/k)$ for CFPA and $\mathcal{L}(M_\ell/\varepsilon/k)$ for CWPA. Moreover, in the CFPA, $2k$ coefficients (the real and imaginary parts of the k DFT coefficients) are noisy while only k coefficients are noisy in the CWPA.

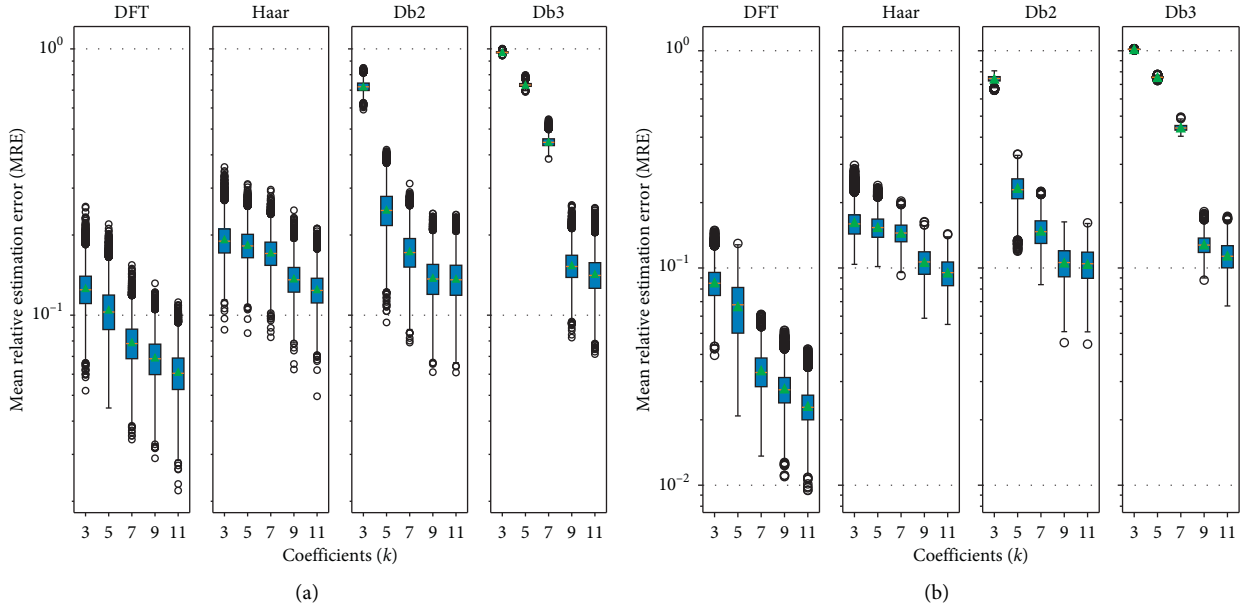


FIGURE 11: Comparison of reconstruction error of DFT and DWT with Haar and Daubechies 2 and Daubechies 3 wavelets for a district of 50 and 450 homes. (a) $N = 50$. (b) $N = 450$.

For a district of 50 homes, we compute the distribution of the magnitude of DFT and DWT with Haar and Daubechies 2 and 3 wavelets, and we compare their coefficient distribution median in Figure 12.

Figure 12 shows that the coefficient values vary according to the values of k and the considered transforms. For instance, when k is in the interval $[7, 10]$, Daubechies 3 obtains the highest magnitudes of coefficients, followed by Daubechies 2 and DFT.

In clamping perturbation algorithms (CFPA, CWPA), the clamping mechanism allows to add a noise proportional to the modulus of the coefficients of the considered transform (DFT, DWT). This reduces the impact of noise compared to perturbation algorithms (FPA, WPA); however, at the price of a perturbation error induced by the clamping of the coefficients. Formally, the perturbation error of clamping perturbation algorithms (CFPA, CWPA) is defined as follows:

Definition 5. Perturbation error for clamping perturbation algorithms (CFPA, CWPA).

Let X^1, \dots, X^N be the individual time-series of energy consumptions of N homes, with $X^i = (x_1^i, \dots, x_T^i)$ for $i = 1, \dots, N$. The sum of time-series consumptions is noted as $S = (S_1, \dots, S_T) = (\sum_{i=1}^N x_1^i, \dots, \sum_{i=1}^N x_T^i)$. For all $i = 1, \dots, N$, we note $\bar{C}^i = (\bar{c}_1^i, \dots, \bar{c}_k^i, c_{k+1}^i, \dots, c_T^i)$ as the result of the considered transform of the time-series consumption X^i whose first k coefficients $(\bar{c}_1^i, \dots, \bar{c}_k^i)$ have been clamped. We note $M = (M_1, \dots, M_k)$ as the maximum magnitude of the first k coefficients of the considered transform. Let $\bar{C} = (\sum_{i=1}^N \bar{c}_1^i + \mathcal{L}(\delta_1/\varepsilon/k), \dots, \sum_{i=1}^N \bar{c}_k^i + \mathcal{L}(\delta_k/\varepsilon/k), \sum_{i=1}^N c_{k+1}^i, \dots, \sum_{i=1}^N c_T^i)$ be the sum of coefficients of the considered transform by perturbing only the first k coefficients, with $\delta_j = M_j \sqrt{2}$ (respectively

$\delta_j = M_j$) for CFPA (respectively for CWPA), for $j = 1, \dots, k$.

Let $\bar{S} = (\bar{S}_1, \dots, \bar{S}_T)$ be the inverse transform of \bar{C} . The perturbation error of \bar{C} equals to the mean relative estimation error (MRE) between S and \bar{S} , given by (we add 1 to the denominator in order to avoid the division by zero). For CWPA, the Laplace noise $\mathcal{L}(M_j \sqrt{2}/\varepsilon/k)$ must be replaced by $\mathcal{L}(M_j/\varepsilon/k)$ for $j = 1, \dots, k$ and the DFT by the DWT,

$$\text{PE}(\bar{C}) = \frac{1}{T} \sum_{\ell=1}^T \frac{|S_\ell - \bar{S}_\ell|}{S_\ell + 1}. \quad (12)$$

The perturbation error depends on the following parameters, k , M_j , ε , and N for $j = 1, \dots, k$. k , M_j ($j = 1, \dots, k$) and ε are parameters of the Laplace distribution, so they have a direct impact on the amplitude of the added noise. Let ε and M_j be fixed; the bigger the k , the smaller the Laplace distribution parameter $\delta_k/\varepsilon/k$ is, and thus, the bigger the noise added on the k first coefficients is. This makes the perturbation error more significant. The choice of M_j is important to define the clamping threshold and it directly impacts the perturbation of the Laplace mechanism. The greater the M_j , the bigger the Laplace noise is, and thus, the more the perturbation error is. The smaller the M_j (close to zero), the less the Laplace noise is, but the more the coefficients are clamped, and thus, the more the perturbation error is. The number of homes N indirectly plays a role in the perturbation error; the larger the N , the more diluted the added noise is. This leads to decrease the perturbation error.

Figure 13 (respectively, Figure 14) shows the distribution of the perturbation error of the clamping perturbation algorithms (CFPA and CWPA) according to k , N , with $\varepsilon = 1$ (respectively, $\varepsilon = 3$).

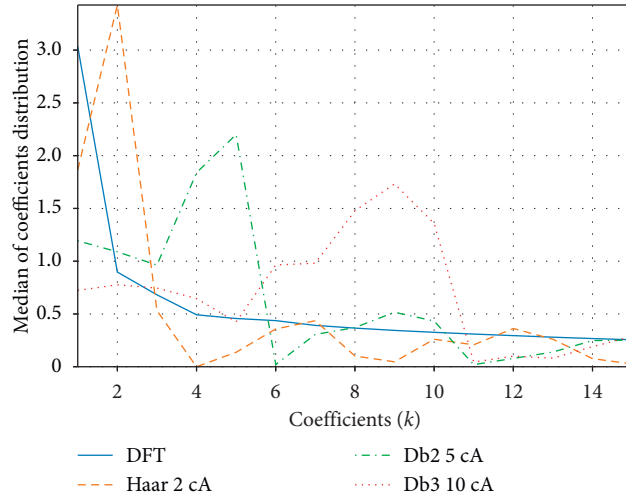


FIGURE 12: Comparison of the median of coefficients distribution of DFT and DWT with Haar and Daubechies 2 and Daubechies 3 wavelets for a district of 50 homes.

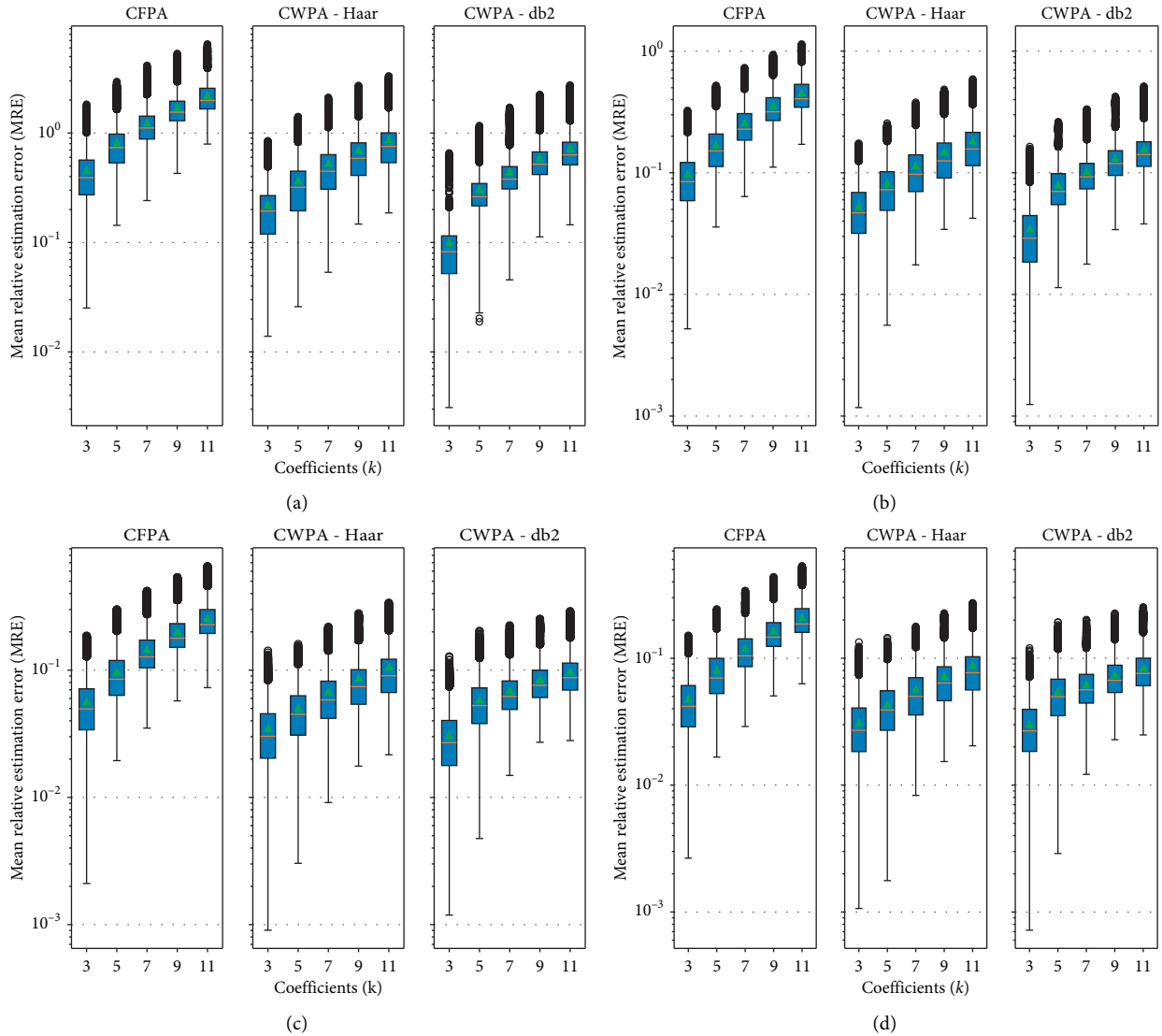


FIGURE 13: Distribution of perturbation error according to clamping perturbation algorithms CFPA and CWPA, with Haar, Daubechies 2, and Daubechies 3, and according to k and the number of homes N , for a fixed privacy budget, $\epsilon = 1$. Note that the scales in (a)–(d) are different. (a) $N = 50$. (b) $N = 250$. (c) $N = 450$. (d). $N = 550$.

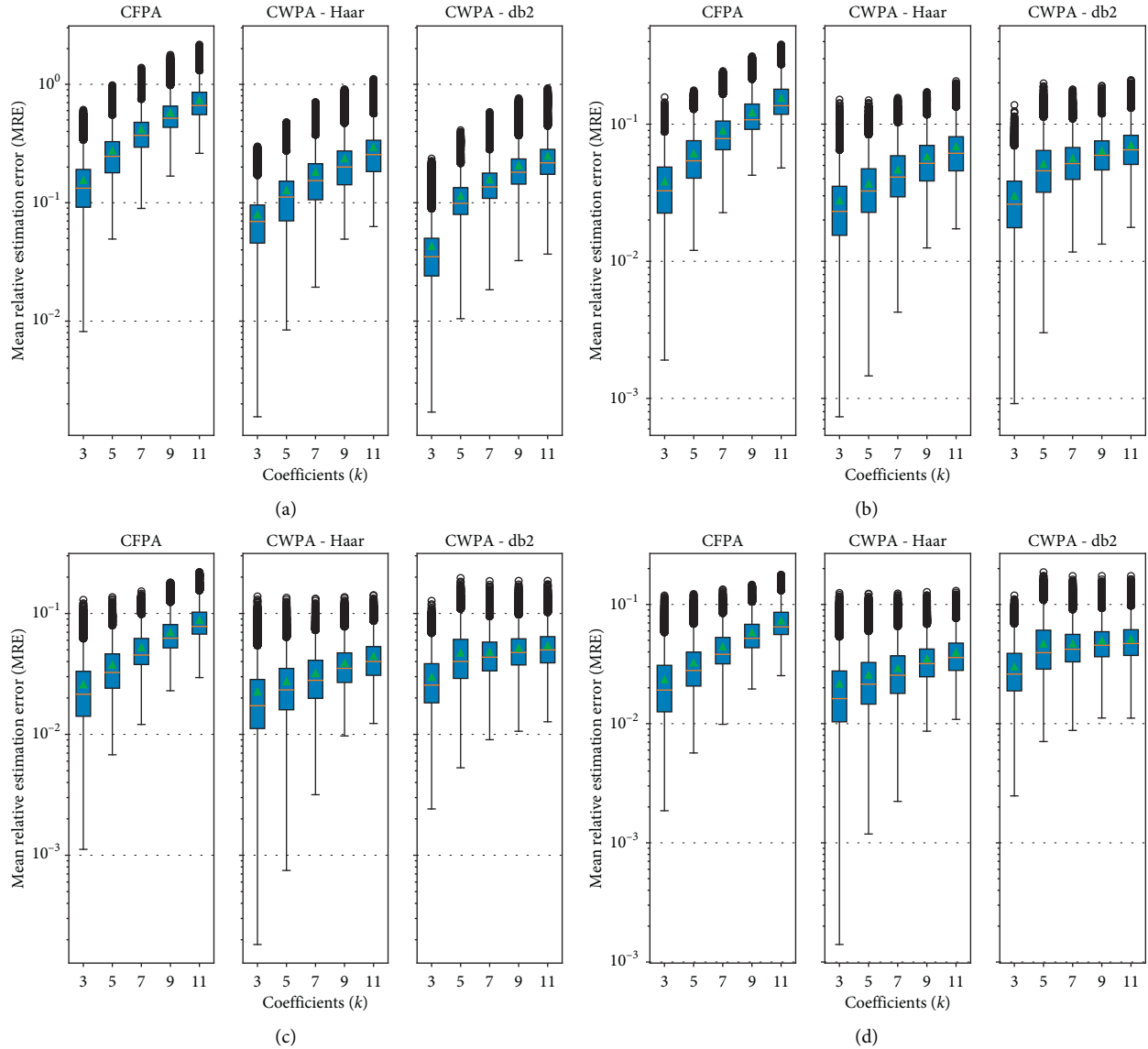


FIGURE 14: Distribution of perturbation error according to clamping perturbation algorithms CFPA and CWPA, with Haar, Daubechies 2, and Daubechies 3, and according to k and the number of homes N , for a fixed privacy budget, $\epsilon = 3$. Note that the scales in 14(a)–14(d) are different. (a) $N = 50$ (b) $N = 250$. (c) $N = 450$. (d). $N = 550$.

Figures 13 and 14 show that the perturbation error of CFPA is higher than that of CWPA. This result is explained by

- (1) The number of coefficients to be noised in CFPA is twice as many as the number of coefficients to be noised in CWPA. Indeed, in CFPA, the DFT coefficients are complex numbers, so both real and imaginary parts must be noised.
- (2) The absolute value of the noise added in the CFPA is $\sqrt{2}$ times greater than that in the CWPA.

In addition, the greater the N , the more the added noise is diluted in the aggregate, causing the perturbation error to decrease. E.g., for $\epsilon = 1$ (Figure 13), when $k = 5$, the median of the perturbation error of CFPA (respectively, CWPA with

Haar) goes from 70% to 7% (respectively from 32% to 4%) when N goes from 50 to 550. Likewise, for $\epsilon = 3$ (Figure 14), when $k = 5$, the median of perturbation error of CFPA (respectively, CWPA with Haar) goes from 25% to 2.7% (respectively from 11% to 2.1%) when N goes from 50 to 550. We notice that, the greater the N , the smaller the difference of the perturbation error between CFPA and CWPA is. This result is also true when ϵ increases. This can be explained by the decrease of the noise introduced on the coefficients of the transforms.

Figures 13 and 14 show that the perturbation error increases when k increases. The larger the k , the smaller the budget ϵ/k allocated to each coefficient is. This leads to a noise increase on each coefficient and thus on the perturbation error.

TABLE 3: Publishing algorithm with the smallest MRE according to the budget of privacy ϵ and the number of homes in the district (N).

Number of homes (N)	Budget of privacy, ϵ	Best algorithm	Coefficients, k	Median of MRE (%)
50	1	CWPA-Haar, db2	5	35
	3	CWPA-Haar		21
150	1	CWPA-Haar	5	19
	3	CFPA		11
250	1	CFPA	5	16
	3			8
350	1	CFPA	5	12
	3		8	7
450	1	CFPA	5	10
	3		8	6
550	1	CFPA	5	9
	3		8	5

6.3. *Summary of the Experimental Results.* The combination of the reconstruction error (Figure 11) and the perturbation error (Figures 13 and 14) enables to determine which transform is appropriate according to the number of homes N and the budget of privacy ϵ , for getting a total error as small as possible.

Lemma 4. *The mean relative error (MRE) of CFPA (respectively, CWPA) is lower than or equal to the sum of the reconstruction error and the perturbation error of CFPA (respectively, CWPA).*

Proof. The proof of the above lemma deferred to the appendix. \square

Section 6.1 shows that the reconstruction error of DFT is lower than that of the considered DWT. For example, when $N = 450$ and $k = 5$, the median of the reconstruction error is 6% for DFT, while it is 13% for the Haar and Daubechies 2 transforms.

However, Section 6.2 shows that algorithms based on DFT (e.g., CFPA) have a higher perturbation error than those based on DWT (e.g., CWPA).

According to Lemma 4, the total error (MRE) is less than or equal to the sum of reconstruction error and perturbation error. Thus, if the reconstruction error or perturbation error is greater than 10%, there is a high probability that the final error will not be less than this threshold.

As the reconstruction error of the DWT is greater than 9%, there is a high probability that the final error of CWPA will not be less than this threshold, even if the Laplace noise decreases, i.e., when the number of homes N or the privacy budget ϵ increases. However, as the reconstruction error of the DFT is small (the median is between 2% and 3% when $k = 7, 8, 9$), then the total error of the CFPA may be lower than that of the CWPA when the impact of Laplace noise decreases. For example, the median of the perturbation error of CFPA is between 3% and 5% when $k = 7, 8, 9$, $N = 550$, and $\epsilon = 3$. This analysis explains why, for $\epsilon = 1$, the CWPA obtains a better utility than the CFPA when the number of homes N is less than 250. For example, when $N = 50$ and $k = 5$, the median of the perturbation error (respectively, the

reconstruction error) of CFPA is 70% (respectively, 10%) against 32% (respectively, 18%) for CWPA using Haar. Thus, the median of MRE of CFPA is between 70% and 80% against 32% and 50% for CWPA.

When N is higher than 250, CFPA gets a better utility than CWPA. For example, when $N = 450$ and $k = 5$, the median of the perturbation error (respectively, the reconstruction error) of CFPA is 8.5% (respectively, 6.5%) against 4.5% (respectively, 16%) for CWPA using Haar. Thus, the median of MRE of CFPA is between 8.5% and 15% against 16% and 20.5% for CWPA.

In this use case, by comparing the different techniques for publishing time-series consumption, it appears that clamping perturbation algorithms (CFPA, CWPA) get a better utility than unbounded algorithms (FPA, WPA), which shows that the clamping mechanism reduces the total error. Furthermore, when the number of homes is greater than 250, CFPA obtains the best utility, with a mean relative error of less than 10% when $\epsilon = 3$. When the budget of privacy $\epsilon = 1$, the mean relative error of CFPA is less than 10% for $N = 450$ homes.

The CWPA gets the best utility when the number of homes N is smaller than 150 and the budget of privacy ϵ is 1. This is justified by its low perturbation error.

Table 3 summarizes the publishing algorithm with the smallest MRE according to the budget of privacy ϵ and the number of homes in the district (N). Based on the dataset from the Irish Commission for Energy Regulation (CER) [39], Table 3 shows that the clamping Fourier perturbation algorithm (CFPA) achieves a lower MRE than the clamping wavelet perturbation algorithm (CWPA) for $N > 150$. Hence, CFPA gets a better utility than CWPA for $N > 150$.

7. Conclusion

The large deployment of smart meters provides users and suppliers with the capacity to optimize the energy consumption through forecasting and demand-response services. This paper proposes an original and efficient approach to mitigate privacy leakages of users' consumptions. This approach uses differential privacy and time-series transformations for supporting high privacy guarantees and utility. The clamping Fourier perturbation algorithm (CFPA)

we propose achieves an error 6 times lower than the Fourier perturbation algorithm (FPA). Similarly, the clamping wavelet perturbation algorithm (CWPA) achieves an error 2 times lower than the wavelet perturbation algorithm (WPA). Thanks to our algorithm, the publication of aggregate time-series consumptions is now possible while guaranteeing that the aggregate does not reveal any individual consumptions and while achieving better utility than existing algorithms. These privacy-preserving aggregate time-series consumptions can then be used as a building block, enabling services such as forecasting and demand-response, which are suitable for improving the efficiency and reliability of the electric grid.

In the future, we plan to investigate how to decentralize our clamping transform perturbation algorithm in order to resist to malicious aggregators. We plan to examine how to combine secure multiparty computation (SMC) with differential privacy (DP). SMC enables parties to compute a joint function without learning any individual inputs. SMC combined with DP could allow homes to compute and publish their aggregated consumptions without relying on an aggregator. However, SMC incurs a communication cost, which might have an impact on the running time performance.

Appendix

Proof of Lemma 4

Lemma Appendix (Lemma 4). *The mean relative error (MRE) of CFPA (respectively, CWPA) is lower than or equal to the sum of the reconstruction error and the perturbation error of CFPA (respectively, CWPA).*

Proof. Let $S = (S_1, \dots, S_T)$ be the aggregate consumption to be published by using CFPA or CWPA. Let $C = (c_1, \dots, c_T)$ be the coefficients of the considered transform of S . For simplicity, we consider that we use the CFPA5; we have $S = \text{IDFT}(c_1, \dots, c_T)$, where IDFT means the inverse of the DFT transform. We note \bar{c}_j as the clamped coefficient of c_j for $j = 1, \dots, k$. Let $\hat{S} = \text{IDFT}(\bar{c}_1 + \mathcal{L}(M_1\sqrt{2}/\epsilon/k), \dots, \bar{c}_k + \mathcal{L}(M_k\sqrt{2}/\epsilon/k), 0, \dots, 0)$ be the result of the aggregate consumption, where $M = (M_1, \dots, M_k)$ is the maximum magnitude of the first k DFT coefficients. Let $d_j = \bar{c}_j - c_j$ for $j = 1, \dots, k$:

$$\begin{aligned} \hat{S} &= \text{IDFT}\left(\bar{c}_1 + \mathcal{L}\left(\frac{M_1\sqrt{2}}{\epsilon/k}\right), \dots, \bar{c}_k + \mathcal{L}\left(\frac{M_k\sqrt{2}}{\epsilon/k}\right), 0, \dots, 0\right) \\ &= \text{IDFT}\left(d_1 + c_1 + \mathcal{L}\left(\frac{M_1\sqrt{2}}{\epsilon/k}\right), \dots, d_k + c_k + \mathcal{L}\left(\frac{M_k\sqrt{2}}{\epsilon/k}\right), 0, \dots, 0\right) \\ &= \text{IDFT}(c_1, \dots, c_T) + \text{IDFT}\left(d_1 + \mathcal{L}\left(\frac{M_1\sqrt{2}}{\epsilon/k}\right), \dots, d_k + \mathcal{L}\left(\frac{M_k\sqrt{2}}{\epsilon/k}\right), 0, \dots, 0\right) \\ &\quad - \text{IDFT}(0, \dots, 0, c_{k+1}, \dots, c_T). \end{aligned} \tag{A.1}$$

Let $\tilde{S} = (\tilde{s}_1, \dots, \tilde{s}_T) = \text{IDFT}(c_1, \dots, c_k, 0, \dots, 0) - \text{IDFT}(c_1, \dots, c_T) = -\text{IDFT}(0, \dots, 0, c_{k+1}, \dots, c_T)$ corresponding to the difference between the aggregate consumption where the last $T - k$ DFT coefficients are replaced by zeros and the initial aggregate consumption (corresponding to the reconstruction error). Let $\bar{S} = (\bar{s}_1, \dots, \bar{s}_T) = \text{IDFT}(\bar{c}_1 + \mathcal{L}(M_1\sqrt{2}/\epsilon/k), \dots, \bar{c}_k + \mathcal{L}(M_k\sqrt{2}/\epsilon/k), c_{k+1}, \dots, c_T) - \text{IDFT}(c_1, \dots, c_T) = \text{IDFT}(d_1 + \mathcal{L}(M_1\sqrt{2}/\epsilon/k), \dots, d_k + \mathcal{L}(M_k\sqrt{2}/\epsilon/k), 0, \dots, 0)$ corresponding to the difference between the aggregate consumption where the first k DFT coefficients are clamped and noisy and the initial aggregate consumption (corresponding to the perturbation error). Then, we obtain $\hat{S} = S + \bar{S} + \tilde{S}$. Let $A = (a_1, \dots, a_n)$ and $B = (b_1, \dots, b_n)$ be two vectors of the same size; we note $A/B = (a_1/b_1, \dots, a_n/b_n)$. Let $S + 1 = (s_1 + 1, \dots, s_T + 1)$, $\hat{S} - S/S + 1 = \bar{S}/S + 1 + \tilde{S}/S + 1$. Then,

$$\begin{aligned} \left\| \frac{\hat{S} - S}{S + 1} \right\|_1 &= \left\| \frac{\bar{S}}{S + 1} + \frac{\tilde{S}}{S + 1} \right\|_1 \\ &\leq \left\| \frac{\bar{S}}{S + 1} \right\|_1 + \left\| \frac{\tilde{S}}{S + 1} \right\|_1. \end{aligned} \tag{A.2}$$

Thus, the MRE of CFPA (respectively, CWPA) is lower than or equal to the sum of the reconstruction error and the perturbation error of CFPA (respectively, CWPA). \square

Data Availability

The dataset used in this paper is from the Irish Commission for Energy Regulation available at <https://www.ucd.ie/issda/data/commissionforenergyregulationcer/>.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

- [1] V. Mishra, *An approach to recovery of critical data of smart cities using blockchain*, Ph.D. thesis, Arizona State University, Tempe, AZ, USA, 2017.
- [2] C. S. Lai, Y. Jia, Z. Dong et al., "A review of technical standards for smart cities," *Clean Technologies*, vol. 2, no. 3, pp. 290–310, 2020.
- [3] C. S. Lai, L. L. Lai, and Q. H. Lai, "Smart grids and big data analytics for smart cities," 2020.

- [4] R. Weron, *Modeling and Forecasting Electricity Loads and Prices: A Statistical Approach*, vol. 403, John Wiley & Sons, New York, NY, USA, 2007.
- [5] T. De Souza, J. Wright, P. O'Hanlon, and I. Brown, "Set difference attacks in wireless sensor networks," in *Proceedings of the International Conference on Security and Privacy in Communication Systems*, Padua, Italy, September 2012.
- [6] S. S. Clark, H. Mustafa, B. Ransford, J. Sorber, K. Fu, and W. Xu, "Current events: identifying webpages by tapping the electrical outlet," in *Proceedings of the ESORICS*, Egham, UK, September 2013.
- [7] U. Greveler, P. Glösekötterz, B. Justusy, and D. Loehr, "Multimedia content identification through smart meter power usage profiles," in *Proceedings of the International Conference on Information and Knowledge Engineering (IKE)*, Las Vegas, NV, USA, July 2012.
- [8] M. Jawurek, F. Kerschbaum, G. Danezis, and SoK, *Privacy Technologies for Smart Grids - A Survey of Options*, Microsoft Res., Cambridge, UK, 2012.
- [9] G. Bauer, K. Stockinger, and P. Lukowicz, "Recognizing the use-mode of kitchen appliances from their current consumption," *Lecture Notes in Computer Science*, vol. 9, pp. 163–176, 2009.
- [10] A. Prudenzi, "A neuron nets based procedure for identifying domestic appliances pattern-of-use from energy recordings at meter panel," *Power Engineering Society Winter Meeting*, vol. 2, 2002.
- [11] G. W. Hart, "Nonintrusive appliance load data acquisition," in *Proceedings: International Load Management Conference*, Bonn, Germany, May 1985.
- [12] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Proceedings of the Theory of Cryptography Conference*, pp. 265–284, Springer, New York, NY, USA, March 2006.
- [13] G. Eibl and D. Engel, "Differential privacy for real smart metering data," *Computer Science - Research and Development*, vol. 32, no. 1-2, pp. 173–182, 2017.
- [14] C. Dwork and A. Roth, "The algorithmic foundations of differential privacy," *Foundations and Trends in Theoretical Computer Science*, vol. 9, pp. 211–407, 2014.
- [15] P. Barbosa, A. Brito, and H. Almeida, "A technique to provide differential privacy for appliance usage in smart metering," *Information Sciences*, vol. 371, pp. 355–367, 2016.
- [16] H. Wang and Z. Xu, "Cts-dp: publishing correlated time-series data via differential privacy," *Knowledge-Based Systems*, vol. 122, pp. 167–179, 2017.
- [17] V. Rastogi and S. Nath, "Differentially private aggregation of distributed time-series with transformation and encryption," in *Proceedings of the 2010 ACM SIGMOD International Conference on Management of Data*, pp. 735–746, Athens, Greece, November 2010.
- [18] F. McSherry and I. Mironov, "Differentially private recommender systems: building privacy into the netflix prize contenders," in *Proceedings of the 15th ACM SIGKDD international conference on Knowledge Discovery and Data Mining*, pp. 627–636, Paris, France, July 2009.
- [19] L. Lyu, Y. W. Law, J. Jin, and M. Palaniswami, "Privacy-preserving aggregation of smart metering via transformation and encryption," in *Proceedings of the 2017 IEEE Trustcom/BigDataSE/ICSS*, pp. 472–479, IEEE, Sydney, Australia, August 2017.
- [20] C. Rottondi, A. Barbato, L. Chen, and G. Verticale, "Enabling privacy in a distributed game-theoretical scheduling system for domestic appliances," *IEEE Transactions on Smart Grid*, vol. 8, pp. 1220–1230, 2016.
- [21] C. Rottondi and G. Verticale, "Privacy-friendly appliance load scheduling in smart grids," in *Proceedings of the International Conference on Smart Grid Communications (SmartGridComm)*, Vancouver, Canada, October 2013.
- [22] C. Thoma, T. Cui, and F. Franchetti, "Secure multiparty computation based privacy preserving smart metering system," in *Proceedings of the 2012 North American power symposium (NAPS)*, pp. 1–6, IEEE, Champaign, IL, USA, February 2012.
- [23] F. Leukam Lako, P. Lajoie-Mazenc, and M. Laurent, "Reconciling privacy and utility for energy services—an application to demand response protocols," in *Proceedings of the 2020 IEEE European Symposium on Security and Privacy Workshops (EuroS & PW)*, pp. 348–355, IEEE, Genoa, Italy, September 2020.
- [24] M. A. Mustafa, S. Cleemput, A. Aly, and A. Abidin, "A secure and privacy-preserving protocol for smart metering operational data collection," *IEEE Transactions on Smart Grid*, vol. 10, no. 6, pp. 6481–6490, 2019.
- [25] T. Dimitriou and M. K. Awad, "Secure and scalable aggregation in the smart grid resilient against malicious entities," *Ad Hoc Networks*, vol. 50, pp. 58–67, 2016.
- [26] G. Danezis, C. Fournet, M. Kohlweiss, and S. Zanella-Béguelin, "Smart meter aggregation via secret-sharing," in *Proceedings of the Workshop on Smart Energy Grid Security*, Berlin Germany, November 2013.
- [27] G. Ács and C. Castelluccia, "I have a DREAM! (Differentially private smart Metering)," in *Proceedings of the International Workshop on Information Hiding*, Prague, Czech Republic, May 2011.
- [28] K. Kursawe, G. Danezis, and M. Kohlweiss, "Privacy-friendly aggregation for the smart-grid," in *Proceedings of the International Symposium on Privacy Enhancing Technologies Symposium*, Waterloo, Canada, July 2011.
- [29] T. Jeske, "Privacy-preserving smart metering without a trusted-third-party," in *Proceedings of the Security and Cryptography (SECRYPT)*, Seville, Spain, August 2011.
- [30] F. D. Garcia and B. Jacobs, "Privacy-friendly energy-metering via homomorphic encryption," in *Proceedings of the International Workshop on Security and Trust Management*, Athens, Greece, September 2010.
- [31] B. Pejó and D. Desfontaines, "Sok: differential privacies," 2020.
- [32] S. L. Garfinkel, J. M. Abowd, and S. Powazek, "Issues encountered deploying differential privacy," in *Proceedings of the 2018 Workshop on Privacy in the Electronic Society*, pp. 133–137, Toronto, Canada, October 2018.
- [33] Ú. Erlingsson, V. Pihur, and A. Korolova, "Rappor: randomized aggregatable privacy-preserving ordinal response," in *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, pp. 1054–1067, Scottsdale, AZ, USA, November 2014.
- [34] Differential Privacy Team, "Learning with privacy at scale," 2017.
- [35] B. Ding, J. Kulkarni, and S. Yekhanin, "Collecting telemetry data privately," in *Proceedings of the Advances in Neural Information Processing Systems*, pp. 3571–3580, Long Beach, CA, USA, December 2017.
- [36] G. Ács, C. Castelluccia, and R. Chen, "Differentially private histogram publishing through lossy compression," in *Proceedings of the 2012 IEEE 12th International Conference on*

Data Mining, pp. 1–10, IEEE, Brussels, Belgium, December 2012.

- [37] “Regulation (EU) 2016/679 of the european parliament and of the council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing directive 95/46/EC (general data protection regulation),” *Official Journal of the European Union*, vol. 119, 2016.
- [38] C. Dwork, “Differential privacy: a survey of results,” in *Proceedings of the International Conference on Theory and Applications of Models of Computation*, Xi’an, China, April 2008.
- [39] “C for Energy Regulation (CER), CER smart metering project - electricity customer behaviour trial, 2009-2010,” 2012, <http://www.ucd.ie/issda/data/commissionforenergyregulationcer/>.
- [40] R. Wang, *Continuous- and Discrete-Time Wavelet Transforms*, Cambridge University Press, Cambridge, UK, 2012.
- [41] E. J. Stollnitz, T. D. DeRose, A. D. DeRose, and D. H. Salesin, *Wavelets for Computer Graphics: Theory and Applications*, Morgan Kaufmann, Burlington, MA, USA, 1996.

Research Article

AB_SAC: Attribute-Based Access Control Model Supporting Anonymous Access for Smart Cities

Runnan Zhang ¹, Gang Liu ¹, Shancang Li ², Yongheng Wei ¹ and Quan Wang ¹

¹School of Computer Science and Technology, Xidian University, Xi'an 710071, China

²Department of Computer Science, University of the West of England, Bristol BS16 1QY, UK

Correspondence should be addressed to Gang Liu; gliu@xidian.edu.cn

Received 3 February 2021; Revised 21 February 2021; Accepted 2 March 2021; Published 22 March 2021

Academic Editor: Qi Li

Copyright © 2021 Runnan Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Smart cities require new access control models for Internet of Things (IoT) devices that preserve user privacy while guaranteeing scalability and efficiency. Researchers believe that anonymous access can protect the private information even if the private information is not stored in authorization organization. Many attribute-based access control (ABAC) models that support anonymous access expose the attributes of the subject to the authorization organization during the authorization process, which allows the authorization organization to obtain the attributes of the subject and infer the identity of the subject. The ABAC with anonymous access proposed in this paper called AB_SAC strengthens the identity-less of ABAC by combining homomorphic attribute-based signatures (HABSs) which does not send the subject attributes to the authorization organization, reducing the risk of subject identity re-identification. It is a secure anonymous access framework. Tests show that the performance of AB_SAC implementation is similar to ABAC's performance.

1. Introduction

Smart cities will involve millions of autonomous smart objects around us, monitoring, collecting, and sharing data without us many times being aware of it [1]. Such pervasive and autonomous behavior can be seen as a peril without proper security and focus on user privacy preferences. Indeed, stakeholders should provide citizens with the tools to ensure their privacy and safely participate in IoT services [2].

Attribute-based access control (ABAC) could address the issues of the fine-granularity of resource protection and the user scalability of network systems, and it can provide appropriate strategies for the access control in the open network environment in the future [3]. With the fine-grained access to data, ABAC can guarantee the security of private information in the validation process of zero trust. ABAC model is considered to be identity-less in its infancy. It is the most important privacy protection-related feature of ABAC. In open computing environments, there is an urgent need for privacy protection and anonymous authorization. In many scenarios, researchers have proposed access control

models that support anonymous authorization [4], such as cloud computing [5, 6], wireless body area networks [7], payment [8], and smart grid [9]. The methods mainly use attributes to encrypt objects to support anonymous authorization in the access control model.

Ahuja and Mohanty [10] proposed a scalable attribute-based encryption (ABE) scheme in cloud environment. The scheme generates a hierarchical attribute private key for users through hierarchical authority and hierarchical ciphertext policy attribute-based encryption (CP-ABE) algorithm. The scheme supports flexible authorization, authority delegation, and authority sharing through hierarchical structure and private key of the hierarchical attribute. Yuen et al. [11] proposed an anonymous ABAC model based on an anonymous certificate to support k-times anonymous authorization for cloud services. When the subject is authenticated, its private key is randomly generated and delivered from the attribute publishing agency. When the subject accesses the service, it needs to obtain the certificate generated by the public key of the trusted institution, the private key of the user, and the number of times of access and

prove that he has the property satisfying a certain policy through its own attribute private key. If the number of accesses in the certificate is less than the maximum number of accesses, the subject can access the service. Given the serious harm of privacy disclosure in the personal health records (PHR) system, Pussewalage and Oleshchuk [12] proposed an anonymous ABAC scheme based on ABE and proxy re-encryption to protect privacy. The scheme assumes that all subjects have public key infrastructure (PKI) certificates. Subjects obtain attribute private key from attribute authorities (AAs) (including PHR owner). Proxy re-encryption reduces the computing cost of the PHR owner. If the subject has attributes that meet the access policy of PHR, it can access the PHR through his own attribute private key.

The above method supports anonymous access by using a certificate instead of a subject identity or ABE. There are some defects in these methods. In some schemes, the certificate of the subject is unique, so the subject's access may be linked to its certificate. The attacker can re-identify the identity of the subject through the access linked to the subject. Once the re-identification is successful, it will cause unexpected privacy disclosure [13]. The encryption of objects based on attributes limits the types of objects. It is friendly to access control of objects that can be moved from server to client, such as files. It is unfriendly to access control of objects that cannot be moved to client, such as web services, which needs additional mechanisms to provide access control and increases the complexity of its implementation. And using an ABE-based algorithm to access objects often needs to download the object or generate tokens, which increases the load of the network.

In this paper, we propose an ABAC model that supports anonymous access called AB_SAC. By combining homomorphic attribute-based signatures (HABSSs) [14] and transferring some functions of ABAC to the attribute authority and audit institution, we strengthen the identity-less of ABAC, so that its authorization is no longer dependent on identity. HABS supports encrypted attribute operations, and the identity of the subject is not included in the signature. This makes it the most suitable encryption algorithm to support anonymous access in ABAC. AB_SAC does not use unique certificates and is friendly for all types of objects. It inherits the features of fine-grained access control, flexible policy, and unlimited object type of ABAC. It overcomes the defect that the ABAC framework does not support anonymous access and provides an audit function that improves security.

Our contribution:

- (1) This paper proposes an ABAC framework that supports anonymous access and fine-grained control of attributes by subjects. AB_SAC is a direct extension of the ABAC framework, which has compatibility with existing ABAC implementations. It inherits many advantages of ABAC, such as fine-grained access control, flexible policy, and unlimited resource. AB_SAC supports audit function.
- (2) AB_SAC's threat model and detailed procedures are described.

- (3) An AB_SAC implementation is shown. Tests show that the performance of AB_SAC implementation is similar to ABAC's performance.

To the best of our knowledge, this paper is the first to propose an ABAC model supporting anonymous authorization by extending the authorization framework of the ABAC. Moreover, this model solves the problems of object type restriction and authorization based on a unique identifier. This paper continues Zhang's work [15].

2. Preliminary

2.1. ABAC Model. The attribute-based access control model (ABAC) is an emerging access control model that has attracted much attention from scholars. The ABAC model has many excellent features such as fine-grained, flexible, dynamic, identity-less, and rich policy expression, which makes it have a wide range of application scenarios, such as distributed computing [16], Internet of Things (IoT) healthcare [17], IoT home devices [18], financial industry [19], and so on. While there is currently no single agreed-upon model or standardization of ABAC, there are commonly accepted high-level definitions and descriptions of its function. One such high-level description is presented in the National Institute of Standards and Technology's publication (NIST), a "Guide to Attribute Based Access Control (ABAC) Definition and Considerations" [20].

Attribute-based access control is an access control method where subject requests to perform operations on objects are granted or denied based on assigned attributes of the subject, assigned attributes of the object, environmental conditions, and a set of policies that are specified in terms of those attributes and conditions.

Many researchers use the framework which is proposed by XACML [21] shown in Figure 1, in which context handler, Policy Enforce Point (PEP), Policy Decision Point (PDP), Policy Information Point (PIP), and Policy Attribute Point (PAP) are internal services of authorization organization and are managed by an authorization organization. Context handler performs preliminary verification of access requests. PEP executes access control decisions. PDP evaluates access requests based on acquired attributes and policies. PIP manages attributes of all subjects, objects, and environment. PAP manages and maintains a policy library. In ABAC, when a subject accesses an object, it sends an access request to an authorized organization. The process of evaluating access requests by the ABAC framework is as follows:

- (1) Subject sends an access request to PEP, and PEP forwards the access request to context handler.
- (2) Context handler verifies the correctness of access request. Context handler sends the access request to PIP. The PIP performs attribute retrieval based on the unique identifier of the subject and object in the access request and returns the subject attribute set, object attribute set, and environment attribute set required for policy evaluation to the context handler.

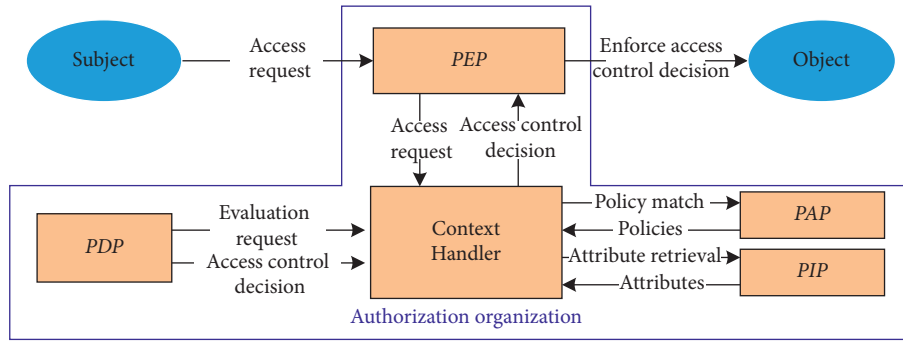


FIGURE 1: ABAC framework with context handler.

- (3) The access request and attribute set are forwarded to PAP. PAP performs policy matching according to the attribute sets provided by context handler and returns an applicable policy set to context handler.
- (4) Context handler sends the subject attribute set, object attribute set, environment attribute set, and applicable policy set to PDP. PDP evaluates access request based on the obtained attribute sets and policy set and returns the access control decision to context handler.
- (5) Context handler sends the access control decision to PEP. PEP enforces the access control decision.

Although the context handler seems to be just forwarding messages throughout the access control process, it improves the compatibility of the framework. The context handler converts the format of messages between different points to make the points compatible. Other researchers use the ABAC framework [6] as shown in Figure 2, except that there is no context handler, and the other processes are consistent with Figure 1.

In the authorization process of ABAC, the subject has certain anonymity, that is, PDP only makes access control decisions based on attributes and policy sets. In this process, the PDP does not need to use the subject identity information contained in the access request which is called identity-less. Identity-less is considered to be the basis for implementing anonymous access in ABAC.

However, the access request used by the ABAC framework shown in Figure 1 must contain the unique identifier of the subject and object. The basis for retrieving attributes is the unique identifier of the subject and object contained in the access request. And PIP is part of the authorization organization. Obviously, the authorization organization has the subject's attributes, the subject's identity (subject's unique identifier), and access records. Once the authorization organization becomes abnormal or becomes untrustworthy, the privacy information is leaked, and the consequences are disastrous.

If the access request submitted by the subject to the authorization organization is anonymous, the authorization organization no longer has access records. The AB_sAC framework proposed in this paper separates PIP subject-related functions from the authorized institution to a trusted

third party. In this way, the subject could choose the attributes for authorization without revealing his identity. In addition, the framework also supports auditing, which can reveal the identity of the subject when necessary and ensure the security of the system. Extending the ABAC framework and giving it anonymous access and audit support can expand ABAC's scope of application and improve its practicality [22].

3. HABS Algorithm

HABS is an anonymous certification scheme based on the attribute-based signatures (ABSs). The ABS is designed for the user to sign a message with fine-grained control over identifying information, and it does not support the properties required for anonymous certification [23]. HABS has a clear identification of missing properties to serve anonymous certification objectives. HABS supports a flexible selective disclosure mechanism at no extra computation cost [14], which is inherited from the expressiveness of ABS for defining access policies.

HABS relies on four procedures based on the following that involves the inspector, subject, issuer, and verifier.

System initialization procedure is shown in Figure 3. The procedure derives a global parameter containing the inspector's public key and pairs of public and private keys for the subject, inspector, and issuer. HABS.Setup and HABS.KeyGen are executed by the trusted third party.

HABS.Setup: it takes as input the security parameter ξ and outputs the global public parameter $params$. This algorithm also derives a pair of public and private keys (pk_{ins}, sk_{ins}) for the tracing authority referred to as the inspector. In the following, public parameters $params$ are assumed to include the public key of the inspector, and all the algorithms have default input $params$.

HABS.KeyGen: this algorithm takes as input the global parameters $params$ and outputs the pair of public and private keys for subjects and the issuer. The public and private keys are noted, respectively, for subjects (pk_s, sk_s) and for the issuer (pk_{is}, sk_{is}) .

Credential issuing procedure is shown in Figure 4. It issues a certified credential for the subject based on its attributes. The HABS.Issue algorithm is executed by the issuer. The HABS.Obtain algorithm is executed by the subject.

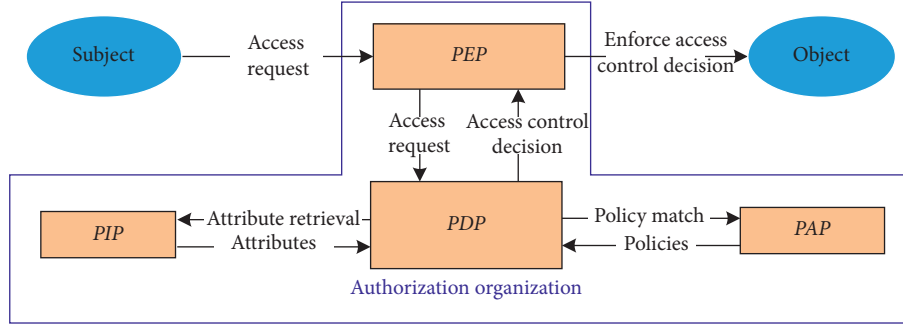


FIGURE 2: ABAC framework without context handler.

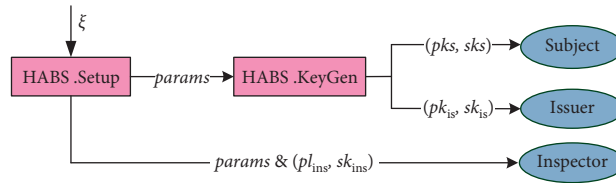


FIGURE 3: System initialization procedure.

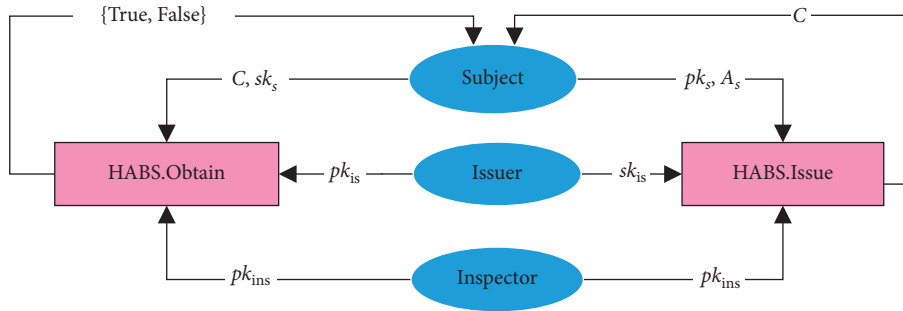


FIGURE 4: Credential issuing procedure.

HABS.Issue: the algorithm takes as input the public key of the subject pk_s , the set of attributes $A_s \subset \mathcal{A}$ belonging to the subject s (where A is referred to as the attribute universe), the private key of the issuer sk_{is} , and the public key of the inspector pk_{ins} . It outputs a certified credential C over the set of attributes A_s .

HABS.Obtain: the algorithm is up to the user to verify the correctness of the received certified credential C over its attributes. The algorithm takes as input the signed commitment C , the private key of the subject sk_s , the public key of the issuer pk_{is} , and eventually the public key of the inspector pk_{ins} . It outputs a bool to point out the correctness of the credential C .

The verifying procedure shown in Figure 5 enables the verifier to check that a subject is authorized to access an object with respect to some access policy. As such, the verifier has first to send a random message to the subject. Second, the user signs the received message based on his credential. In a nutshell, the subject signs the received message based on the subset of his attributes that satisfy the

signature predicate. The user finally sends his signature to the verifier who checks the resulting signature. The HABS.Show algorithm is executed by the subject. The HABS.Verify algorithm is executed by the verifier.

The HABS.Show algorithm takes as input the randomized message m , a signing predicate Υ , the private key of the subject sk_s , the credential C , and a subset of its attributes A'_s , such as $\Upsilon(A'_s) = 1$. This algorithm outputs a signature Σ (or an error message \perp).

The HABS.Verify algorithm takes as input the received signature Σ , the public key of the issuer pk_{is} , the signing predicate Υ , and the message m . It outputs a bool.

The HABS supports the inspection procedure shown in Figure 6 performed by a separate and trusted entity referred to as the inspector. It relies on two algorithms, namely, HABS.trace and HABS.judge, needed to identify the subject and give a proof of judgment. They are executed by the inspector.

The HABS.trace algorithm takes as input the secret key of the inspector sk_{ins} , the issuer public key pk_{is} , and the

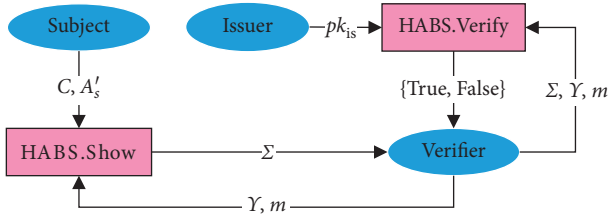


FIGURE 5: Verifying procedure.

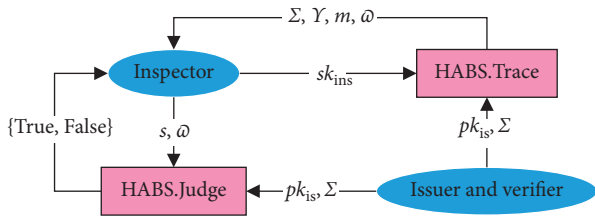


FIGURE 6: Inspection procedure.

signature Σ . It outputs the identity S of the subject that has signed the message m with respect to the predicate Y . It also outputs a proof ω .

The HABS.judge algorithm takes as input the public key of the issuer pk_{is} , the signature Σ , the identity S of the subject, and the proof ω . It outputs a bool, where True means that ω is a valid proof which proves that signature originates from subject (S) Σ .

HABS supports a flexible selective disclosure mechanism that allows the subject to sign the message m with a subset of its attributes A'_s . The verifier only obtains A'_s after successful verification and does not reveal the identity of the subject. In addition, this scheme ensures the unlinkability between sessions while maintaining the anonymity of the subject.

The flexible selective disclosure mechanism of HABS and the identity-less feature of ABAC have complementary advantages. It maintains the anonymity of subjects while providing appropriate attributes for the ABAC authorization process. At the same time, HABS adds the inspector's public key pk_{ins} when generating the anonymous attribute certificate, and when necessary, the inspector audits specific access. The HABS algorithm can be well combined with the ABAC model, giving it anonymous access and auditing capabilities.

4. AB_SAC Framework

AB_SAC is an access control framework with anonymous access and audit capabilities. The authorization agency is not malicious and curious. It tries to obtain the identity of the subject by analyzing the information carried in the subject's access request. The AB_SAC prevents the authority from acquiring the identity of the subject.

4.1. The Architecture of AB_SAC. AB_SAC shown in Figure 7 is a direct extension of the ABAC framework, which has compatibility with existing ABAC implementations. It inherits many advantages of ABAC, such as fine-grained access control, flexible policy, and unlimited object. The

functions of PDP, PEP, and PAP in the AB_SAC framework are consistent with the corresponding modules in the framework shown in Figure 2. The main functions of other modules are as follows:

- (i) PIP: in order to achieve anonymous access in AB_SAC, the subject attributes are managed by the attribute authority (AA) and the subject itself. PIP is only responsible for collecting and managing object attributes and environmental attributes. And PIP sends the corresponding attributes to the context handler according to the attribute requests.
- (ii) Context handler: repackages and redirects requests based on context.
- (iii) Key distribution center (KDC): a trusted third party responsible for generating and distributing keys.
- (iv) Attribute authority (AA): in AB_SAC, AA is responsible for collecting and managing all subject attributes. During the subject authentication process, the AA generates an attribute certificate for the subject based on the attributes owned by the subject. And AA helps the audit authority to complete the audit when the audit is needed.
- (v) Audit authority (Au): in AB_SAC, Au is responsible for auditing subject access. When the user conducts illegal operations or under other situations that require auditing, the context handler sends an audit request to Au. With the help of AA, Au can track the signature and reveal the identity of the subject.

KDC, AA, and Au should be trusted third parties. If KDC and Au are not trusted third parties, then the authorization organization can directly initiate an audit to obtain the identity of the subject. If AA is controlled by an authorization organization, it can identify the subject by issuing attributes with unique values and creating corresponding policies.

4.2. Workflow of AB_SAC. The ABAC model has four workflows, which are initialization, registration, anonymous access, and audit. Before the system starts running, the initialization workflow is executed:

- (i) (i.1) KDC runs HABS.Setup and HABS.KeyGen to generate public and private key pairs for AA and Au and distributes secret keys.

After the system is initialized, the system is started. Registration, anonymous access, and audit are all performed at runtime. When the subject enters the system for the first time, it executes the registration workflow:

- (i) (r.1) The subject sends a registration request to AA and KDC.
- (ii) (r.2) The AA assigns attributes to the subject according to the registration request and returns the attribute set. KDC generates a public and private key pair for the subject and sends it to the subject.
- (iii) (r.3) The subject sends a certificate request to the AA, and the request contains the subject's public key.

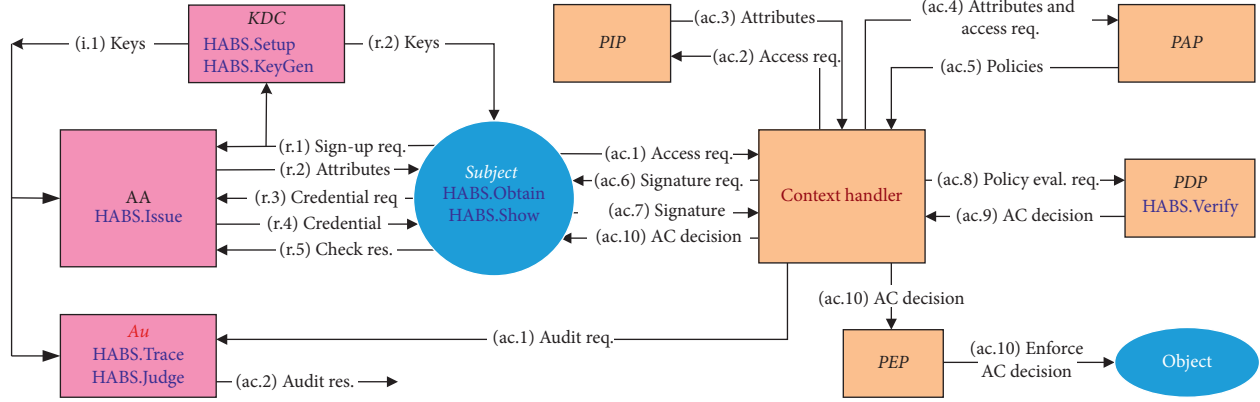


FIGURE 7: AB_SAC framework; the pink module is provided by a trusted third party; the yellow module is provided by the authorization organization; the blue module is the subject and object; req. means request; res. means result; eval. means evaluation.

- (iv) (r.4) After receiving the certificate request, AA issues the certificate for the subject and returns it to the subject.
- (v) (r.5) After receiving the attribute certificate, the subject verifies whether the attribute certificate is consistent with its own attribute set and returns the verification result to the AA.

The subject can obtain a new certificate by re-registering, and the old certificate will be revoked.

When the system is running, the subject anonymously accesses the object:

- (i) (ac.1) When the subject accesses the object, it sends an access request to the context processor. The access request contains only the object's unique identifier and operation.
- (ii) (ac.2) After receiving the access request, the context handler forwards it to the PIP.
- (iii) (ac.3) The PIP returns the relevant object attributes and environment attributes to the context handler according to the unique identifier of the object in the access request.
- (iv) (ac.4) The context handler forwards the object attributes, environment attributes, and access request to PAP.
- (v) (ac.5) PAP performs policy matching based on these attributes. PAP returns the policies set suitable for these attributes to the context handler.
- (vi) (ac.6) Due to the lack of subject attributes in the access request, policy evaluation cannot be performed. The context handler generates the signature predicate Υ and the random message m for each policy in the policy set and packages them into signature requests and send signature requests to the subject.
- (vii) (ac.7) After receiving the signature request, the subject selects the appropriate subset of attributes A'_s and runs HABS. Show to sign the random message m in the signature request and sends the signature Σ to the context handler.

- (viii) (ac.8) After receiving the signature, the context handler packages the signature with the corresponding policy, object attributes, and environment attributes into a policy evaluation request and sends it to the PDP.
- (ix) (ac.9) After receiving the policy evaluation requests, the PDP evaluates the policies based on the attributes in the requests. Then, PDP combines the evaluation results of these policies to form an access control (AC) decision and returns it to the context handler.
- (x) (ac.10) The context handler sends the access control decision to the subject. If the decision is denied, the workflow ends. If the decision is permitted, it is forwarded to PEP and PEP enforces the access control decision.

When abnormal access of the subject is discovered, the audit workflow can be started:

- (i) (au.1) The context sends an audit request to Au. The audit request contains the signature Σ and access request.
- (ii) (au.2) Au evaluates the audit request. If the audit decision is denied, the audit decision will be output directly. If the audit decision is permitted, the HABS. Trace and HABS. Judge algorithm is executed based on the information in the audit request to expose the corresponding subject identity. Then, Au packages the subject identity and the evidence as to the audit decision and outputs it.

Obviously, the subject identity is not involved in AB_SAC's access workflow. The subject-related information obtained by the context handler or PDP in the access workflow is the signature Σ . The signature Σ only involves the predicate Υ , the message m , and a subset of subject attributes A'_s . Thus, AB_SAC's access workflow is identity independent, and AB_SAC's access is anonymous. The message m in the signature request is randomly generated, making it impossible for the authorization organization to link multiple accesses to one credential. This allows AB_SAC to overcome the drawback of using unique

credentials. The operations on objects in AB_SAC are performed by PEP, and there is no restriction on the type of objects. AB_SAC's improvement of ABAC does not involve the policies, so it inherits many advantages of ABAC, such as the fine-grained access control and flexible policy.

5. Analysis

Kaaniche and Laurent [14] proved the correctness, unforgeability, anonymity, and the anonymity removal of the HABS algorithm. The security feature of AB_SAC is the same as the HABS algorithm. AB_SAC has the characteristics of HABS and ABAC.

There are many literatures comparing multiple characteristics of different models, which can help researchers quickly understand these models. Servos et al. [22] divided the existing ABAC works into ABAC implementation, ABAC model, policy, attribute, and other categories and compared the granularity, flexibility, and dynamic characteristics of these works. Aftab et al. [24] analyzed and compared ABAC and RBAC. The evaluation in this article involves 12 issues such as granularity, flexibility, dynamics, role explosion, and interpretability. Ouaddah et al. [25] evaluated multiple models based on six characteristics: scalability, availability, flexibility, context, real-time, granularity, and delegation. Qi et al. [26] analyzed the advantages and disadvantages of RBAC, ABAC, and hybrid models and proposed objective evaluation methods for model granularity, flexibility, and decision efficiency.

Our analysis involves six characteristics, features of the model anonymous access, auditing, restricted object type, fine-grained access control, and policy flexibility. Anonymous access and auditing involve security issues; restricted object type, fine-grained access control, and policy flexibility involve availability. The characteristics included in the security issue are related to anonymous access; the characteristics included in the availability issue are always used to evaluate access control models.

The comparison between AB_SAC and the existing solution is shown in Table 1.

6. Implementation

To verify the availability of AB_SAC and to test the performance difference between AB_SAC and ABAC, AB_SAC is implemented. ABAC implementation [27] contains all the modules and steps in Figure 1. The AB_SAC implementation which contains all the modules and steps in Figure 7 is transformed from the ABAC implementation. The sample attributes and policies used in the implementation are part of the student homework management system. In this implementation, the attributes owned by the student include student number, gender, identity, grade, class, and course. The object is the homework file. Operations include read, update, create, and delete. The functionality and performance of the implementation are tested by randomly generating access requests.

6.1. Functionality Testing. When the homework management system is initializing, execute (i.1) to generate the key for the corresponding module. After the initialization of the system, the user could register. After execution of (r.1) to (r.5), the user obtains the corresponding attribute certificate and could access objects. The user access to the resource process is (ac.1) to (ac.10), resulting in between granted or denied. Users only need to initiate requests; property signatures and other works are automatically processed by the implementation. As shown in Figure 8(a), if a student with the student number 090854 wants to view their C++ homework, the user sends an access request: {object: C++ homework grade 2 class 3.pdf; operation: read}. After the system receives the user's access request, it returns the corresponding signature predicate Υ to the user: {job occupation = student or teacher, grade = 2, class = 3}. The client automatically completes the signature and returns the results to the authorization authority. The authorization authority verifies the signature and gives access control results. Figures 8(a) and 8(b) are two access requests, and Figures 8(c) and 8(d) are their corresponding signature predicate and access control results. AB_SAC has achieved the desired function.

6.2. Performance Testing. In the performance test, the factors affecting AB_SAC performance are explored. The performance of the ABAC implementation and the AB_SAC implementation is compared.

Table 2 shows the environment configuration of this experiment. Table 3 shows the policy configuration of this experiment. The attributes are randomly assigned to each user. The test was conducted in 10 rounds for each scheme. Each round of testing consists of system startup, user registration, and 1000 accesses. Access requests are randomly generated. Figure 8 shows the time for the ABAC implementation and the AB_SAC implementation to complete 1000 accesses.

In scheme 1, the AB_SAC implementation takes an average of 2.3 ms to complete a policy evaluation and the evaluation time is stable. As shown in Figure 9, the ABAC implementation and the AB_SAC implementation are not much different in execution efficiency, and AB_SAC is available.

Comparing schemes 1, 2, and 3, they have no significant difference in execution time which is shown in Figure 10. This is because in ABAC implementation, the execution time of finding attributes based on the identity of the subject is constant. The execution time of the HABS increases with the number of attributes, but the impact is very small.

Comparing schemes 1, 4, and 5, the execution time increases as the number of policies increases (Figure 11). The ABAC implementation needs to traverse the policy library when searching for a suitable policy. The execution time of HABS is not affected by the size of the policy library.

In summary, there is no significant difference between the execution time of AB_SAC implementation and ABAC

TABLE 1: Solution comparison.

Literature	Feature	Anonymous access	Fine-grained AC	Policy flexibility	Audit	Restricted object type
[11]	Attribute certificate, k-times anonymity	Y	Y	N	N	N
[10]	Hierarchical CP-ABE, delegation	Y	Y	N	N	Y
[12]	CP-ABE, proxy re-encryption	Y	Y	N	N	Y
[4]	Attribute certificate	Y	Y	N	Y	N
[5]	Attribute certificate, hierarchical structure	Y	N	N	Y	N
[6]	CP-ABPRE	Y	Y	N	N	Y
AB ₅ AC	Nonunique identifier	Y	Y	Y	Y	N

User request

Student number	190203
Attributes	job occupation=student , grade=2 , class=3
Object	C++ homework grade 2 class 3.pdf
<input checked="" type="checkbox"/> Read <input type="checkbox"/> Write <input type="checkbox"/> Delete <input type="checkbox"/> Update	
<input type="button" value="Submit"/> <input type="button" value="Reset"/>	

(a)

User request

Student number	190203
Attributes	job occupation=student , grade=2 , class=3
Object	C++ homework grade 2 class 1.pdf
<input checked="" type="checkbox"/> Read <input checked="" type="checkbox"/> Write <input type="checkbox"/> Delete <input type="checkbox"/> Update	
<input type="button" value="Submit"/> <input type="button" value="Reset"/>	

(b)

Request evaluation result

Evaluation result	Grant
Access control log	
Original request:	
Object : C++ homework grade 2 class 3.pdf	
Operation : Read	
Signature predicate:	
job occupation=student or teacher , grade=2 , class=3	
Access control result :	
Grant	
Time :	
2.2ms	

(c)

Request evaluation result

Evaluation result	Deny
Access control log	
Original request:	
Object : C++ homework grade 2 class 1.pdf	
Operation : Read, Write	
Signature predicate:	
job occupation=student or teacher , grade=2 , class=1	
Access control result :	
Deny	
Time :	
2.5ms	

(d)

FIGURE 8: Sample access requests and evaluation of the requests.

TABLE 2: Environment configuration.

Operating system	Windows 7
CPU	Intel(R) Xeon(R) CPU E5530 @2.40 GHz
Memory	8 GB

TABLE 3: Policy configuration.

Scheme	Number of attributes	Number of policies
Scheme 1	5	500
Scheme 2	3	500
Scheme 3	7	500
Scheme 4	5	750
Scheme 5	5	1000

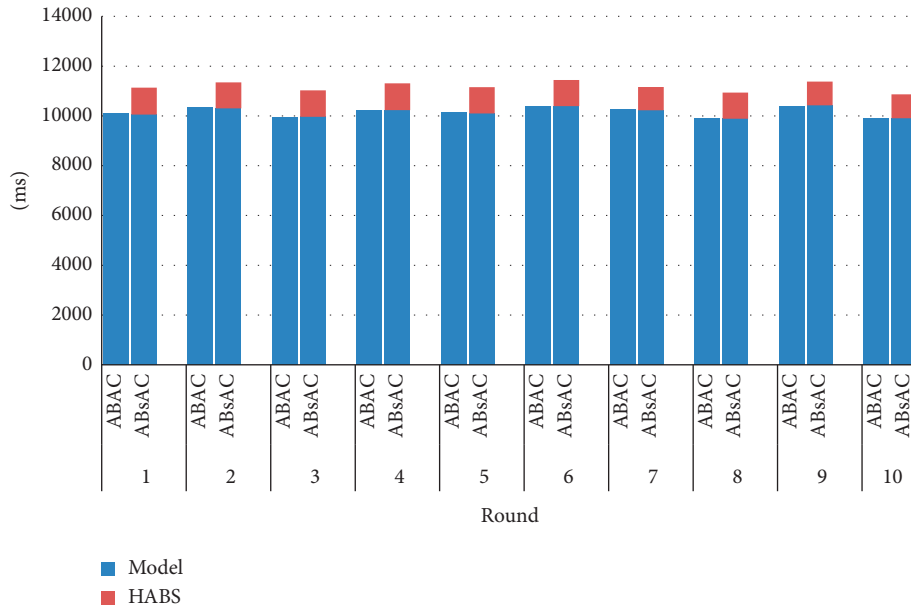


FIGURE 9: Performance of ABAC and AB_sAC; ms is milliseconds; blue bar is time consumption of native ABAC implementation; red bar is time consumption of HABS.

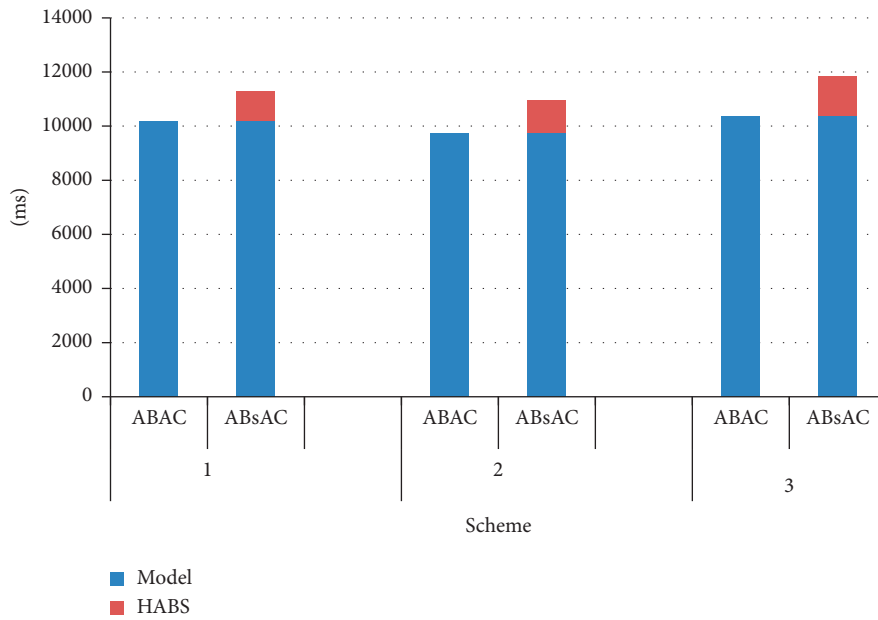


FIGURE 10: Performance of schemes 1, 2, and 3.

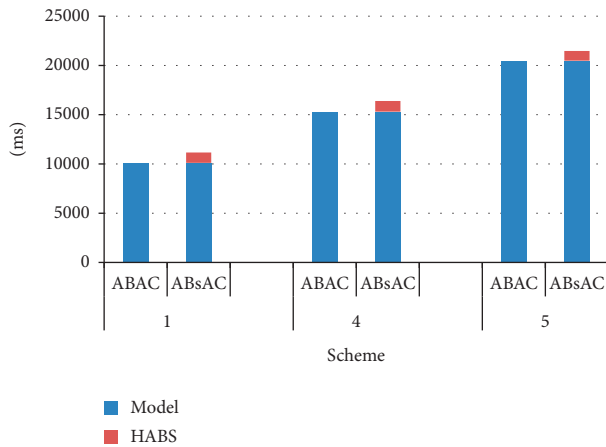


FIGURE 11: Performance of schemes 1, 4, and 5.

implementation. From a performance perspective, AB_sAC is available. The main factor affecting the execution time of AB_sAC is the number of policies.

7. Conclusion

Existing anonymous access solutions have the problems of subject re-identification and constraints on the types of objects. The ABAC with anonymous access proposed in this paper called AB_sAC inherits the features of the ABAC model, such as fine-grained authorization, policy flexibility, and unlimited object types. By combining HABS, it strengthens the identity-less of ABAC, so that the access does not involve a unique identification, reducing the risk of subject identity re-identification.

In ABAC, the authorization organization has the subject's attributes, subject's identity (subject's unique identifier), and access records. Once the authorization organization becomes abnormal or becomes untrustworthy, the privacy information is leaked, and the consequences are disastrous. The AB_sAC framework proposed in this paper separates PIP subject-related functions from the authorized institution to a trusted third party. The identity of the subject no longer participates in the process of access, which reduces the pressure on the privacy protection of authorized organizations. In the future work, we will build a variety of different IoT application environments to test the performance of AB_sAC in different environments.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This study was supported in part by the Shaanxi Key R&D Program under grant no. 2019ZDLGY13-01 and the

National Natural Science Foundation of China under grant no. 61972302.

References

- [1] A. Al-Fuqaha, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of things: a survey on enabling technologies, protocols, and applications," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015.
- [2] V. Guizani, Juan Antonio Martinez, and A. F. Skarmeta, "User-centric access control for efficient security in smart cities," in *Proceedings of the 2017 Global Internet of Things Summit (GloTS)*, IEEE, Geneva, Switzerland, 2017.
- [3] X. Wang, H. Fu, and L. Zhang, "Research progress on attribute-based access control," *Acta Electronica Sinica*, vol. 7, pp. 1660–1667, 2010.
- [4] M. Backes, C. Jan, and D. Sommer, "Anonymous yet accountable access control," in *Proceedings of the 2005 ACM Workshop on Privacy in the Electronic Society*, Alexandria, VA, USA, 2005.
- [5] X. Yao, H. Ning, L. T. Yang, and Y. Xiang, "Anonymous credential-based access control scheme for clouds," *IEEE Cloud Computing*, vol. 2, no. 4, pp. 34–43, 2015.
- [6] Y. Liu, X. Chen, and H. Li, "Anonymous attribute-based proxy re-encryption for access control in cloud computing," *Security and Communication Networks*, vol. 9, no. 14, pp. 2397–2411, 2016.
- [7] A. Li, "Context-aware authorization and anonymous authentication in wireless body area networks," in *Proceedings of the 2018 IEEE International Conference on Communications (ICC)*, pp. 1–7, IEEE, Kansas City, MO, USA, 2018.
- [8] K. Shuaib, "Secure charging and payment protocol (SCPP) for roaming plug-in electric vehicles," in *Proceedings of the 2017 4th International Conference on Control, Decision and Information Technologies (CoDIT)*, pp. 0173–0178, IEEE, Barcelona, Spain, 2017.
- [9] X. Li, S. Kumari, L. Xu, A. K. Sangaiah, and K.-K. R. Choo, "A provably secure and anonymous message authentication scheme for smart grids," *Journal of Parallel and Distributed Computing*, vol. 132, pp. 242–249, 2019.
- [10] R. Wu and S. Mohanty, "A scalable attribute-based access control scheme with flexible delegation cum sharing of access privileges for cloud storage," *IEEE Transactions on Cloud Computing*, vol. 8, no. 1, pp. 32–44, 2017.
- [11] T. H. Yuen, J. K. Liu, M. H. Au, X. Huang, and W. Susilo, "\$K \$-times attribute-based anonymous access control for cloud computing," *IEEE Transactions on Computers*, vol. 64, no. 9, pp. 2595–2608, 2014.
- [12] H. S. Pusewalage and V. Oleshchuk, "A patient-centric attribute based access control scheme for secure sharing of personal health records using cloud computing," in *Proceedings of the 2016 IEEE 2nd International Conference on Collaboration and Internet Computing (CIC)*, pp. 46–53, Pittsburgh, PA, USA, 2016.
- [13] L. Sweeney, "k-anonymity: a model for protecting privacy," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 5, pp. 557–570, 2002.
- [14] N. Kaaniche and M. Laurent, "Attribute-based signatures for supporting anonymous certification," in *Proceedings of the European Symposium on Research in Computer Security*, pp. 279–300, Springer, Cham, Switzerland, 2016.
- [15] R. Zhang, "A³BAC: attribute-based access control model with anonymous access," vol. 344, pp. 441–447, in *Proceedings of the International Conference on Security and Privacy in New*

- Computing Environments*, vol. 344, pp. 441–447, Springer, Cham, Switzerland, 2020.
- [16] G. Fedrechski, “Attribute-based access control for the swarm with distributed policy management,” *IEEE Transactions on Consumer Electronics*, vol. 65, no. 1, pp. 90–98, 2018.
 - [17] Y. Yang, X. Liu, and R. H. Deng, “Lightweight break-glass access control system for healthcare Internet-of-Things,” *IEEE Transactions on Industrial Informatics*, vol. 14, no. 8, p. 3610, 2018.
 - [18] A. L. M. Neto, “Attributed-based authentication and access control for IoT home devices: demo abstract,” in *Proceedings of the 17th ACM/IEEE International Conference on Information Processing in Sensor Networks*, pp. 112–113, Porto, Portugal, 2018.
 - [19] M. Qiu, B. Thuraisingham, L. Tao, and H. Zhao, “Proactive user-centric secure data scheme using attribute-based semantic access controls for mobile clouds in financial industry,” *Future Generation Computer Systems*, vol. 80, pp. 421–429, 2018.
 - [20] D. F. Hu, R. Ferraiolo, and D. F. Kuhn, “Guide to attribute based access control (ABAC) definition and considerations (draft),” *NIST Special Publication*, vol. 800, no. 162, 2013.
 - [21] Standard OASIS, “Extensible access control markup language (xacml) version 3.0,” 2013, http://docs.oasis-open.org/xacml/2.0/access_control-xacml-2.0-core-spec-os.Pdf.
 - [22] D. Servos and S. L. Osborn, “Current research and open problems in attribute-based access control,” *ACM Computing Surveys*, vol. 49, no. 4, pp. 1–45, 2017.
 - [23] H. K. Maji, M. Prabhakaran, and M. Rosulek, “Attribute-based signatures,” in *Proceedings of the Cryptographers’ Track at the RSA Conference*, pp. 376–392, Springer, Berlin, Heidelberg, 2011.
 - [24] U. Muhammad, “The evaluation and comparative analysis of role based access control and attribute based access control model,” in *Proceedings of the 2018 15th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP)*, pp. 35–39, IEEE, Chengdu, China, 2018.
 - [25] A. Ouaddah, A. Abou Elkalam, and A. Ait Ouahman, “Access control in the Internet of Things: , big challenges and new opportunities,” *Computer Networks*, vol. 112, pp. 237–262, 2017.
 - [26] H. Mousannif, X. Di, and J. Li, “Formal definition and analysis of access control model based on role and attribute,” *Journal of Information Security and Applications*, vol. 43, pp. 53–60, 2018.
 - [27] <https://github.com/JimmyKang13/TestDemoSet>.

Research Article

Enhancing Digital Certificate Usability in Long Lifespan IoT Devices by Utilizing Private CA

Daiki Yamakawa ^{1,2} **Takashi Okimoto** ^{1,3} **Songpon Teerakanok** ^{1,4}
Atsuo Inomata ^{1,4,5} and **Tetsutaro Uehara** ^{1,3}

¹Cyber Security Laboratory, Ritsumeikan University, Kusatsu, Shiga 525-8577, Japan

²Graduate School of Information Science and Engineering, Ritsumeikan University, Kusatsu, Shiga 525-8577, Japan

³College of Information Science and Engineering, Ritsumeikan University, Kusatsu, Shiga 525-8577, Japan

⁴Research Organization of Science and Technology, Ritsumeikan University, Kusatsu, Shiga 525-8577, Japan

⁵Osaka University, Suita, Japan

Correspondence should be addressed to Daiki Yamakawa; yamakawa@cysec.cs.ritsumei.ac.jp

Received 4 December 2020; Revised 22 January 2021; Accepted 7 February 2021; Published 16 February 2021

Academic Editor: Chalee Vorakulpipat

Copyright © 2021 Daiki Yamakawa et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Today, smart devices and services have become a part of our daily life. These devices and services offer a richer user experience with a much higher quality of services than before. Many of them utilize sensing functions via cloud architecture to perform remote device controls and monitoring. Generally, the security of the communication between these devices and the service provider (e.g., cloud server) is achieved by using the TLS protocol via PKI standard. In this study, we investigate the risk associating with the use of public certificate authorities (CAs) in a PKI-based IoT system. An experiment is conducted to demonstrate existing vulnerabilities in real IoT devices available in the market. Next, the use of a private CA in the cloud-centric IoT architecture is proposed to achieve better control over the certificate issuing process and the validity period of the certificate. Lastly, the security analysis pointing out the strengths and drawbacks of the proposed method is discussed in detail.

1. Introduction

Emerging of the Internet of Things (IoT) brings a big leap in technological advancements to today's information system. IoT technologies introduce a new way of connecting IoT devices (so-called "things") to the Internet. It allows many devices to form networks for various purposes. By collecting a large amount of data from these devices, IoT provides users with a richer experience and higher quality of services. On the other hand, these new advancements in technology come with new challenges regarding security and privacy. In this paper, the network security aspects of cloud-based IoT devices are studied and discussed.

Unlike the communication between a web browser and the webserver, in the cloud-centric IoT architecture, IoT devices require predetermined server information to successfully initiate communication with the cloud server.

During communication with external servers, transport layer security (TLS) is often used as a primary method for authenticating and providing confidentiality of data via encryption and digital signatures. Generally, TLS requires server certificates issued by public certification authorities (CAs) during the handshake period. However, there often are reports about vulnerabilities due to the lack of appropriate and consistent verification methods of the certificate chain [1, 2].

Furthermore, the TLS end-entity certificate has a maximum validity period of 398 days, according to Baseline Requirement version 1.7.3 [3]. With each update of the Baseline Requirement, the validity period of the TLS certificate is getting shorter every time, while the cost of certificate renewal is getting higher. This poses threats to some IoT systems in which IoT devices may not be connected to the network for a long period.

To achieve TLS security, many manufacturers currently use public CA to issue certificates for their systems. In this work, we point out that the use of public CAs, however, may not be an optimal choice for IoT systems due to the long lifespan of IoT devices, certificate cost, and inflexible requirement of the issuing process. In contrast, using a company's owned private CA may be an attractive way to tackle these problems in IoT. By using private CA, it allows the company to flexibly issue certificates for all their servers and devices with customizable details, such as the validity period. This will help to reduce the risk of certificate verification failure for some long lifespan IoT devices. Furthermore, since the private CA can issue any number of certificates for the company without additional costs, the use of private CA also provides scalability to the business. Lastly, without relying on trusted third parties such as public CAs, some unnecessary risks regarding external factors can be removed.

There are 2 main contributions in this research. First, we introduce a risk assessment method to analyze the security aspects regarding the use of TLS in various scenarios, especially when the IoT devices are not connected to the network for a long time. Second, a PKI-based IoT architecture utilizing a private certificate authority is presented. The proposed method is designed to solve the security problems according to the analyzed results obtained from the risk assessment process.

The rest of this paper is organized as follows. Section 2 gives some background information regarding the pinning process and also some examples of past security incidents. Sections 3 and 4 present a risk analysis of applying TLS to the IoT and a demonstration of an attack showing vulnerabilities in today's IoT devices, respectively. Next, the proposed method of utilizing private CA over the public one is introduced in Section 5. In Section 6, we discuss the security aspects of the proposed method and compare it against the traditional PKI-based architecture. Finally, we briefly conclude this paper in the last section.

2. Background

In the following subsections, some background knowledge and information are provided. First, the concept of cloud-centric IoT architecture is described. We then discuss the severity of the attack against TLS communication by addressing past incidents caused by operational errors in TLS. Next, a brief introduction to pinning techniques is presented. Finally, some related work and past research are presented and discussed.

2.1. Cloud-Centric IoT System Architecture. According to the 2019 White Paper on Information and Communications [4] published by the Ministry of Internal Affairs and Communications, Japan, the number of IoT devices is expected to be approximately 44.8 billion worldwide by 2021, and this number is expected to increase in the future. Generally, in cloud-centric architecture, a user uses his/her control device (typically a smartphone) to send remote commands or receive the operational status of an IoT device (e.g., home

appliances) from the cloud. Regarding user security and privacy, PKI-based SSL/TLS communication is often used to establish secure communication in cloud-centric IoT architecture by authenticating the sender and encrypting transmitted data to ensure data confidentiality. Figure 1 shows an example of a communication between a smartphone application with an IoT device via a cloud server.

2.2. Past Incidents. Regarding the use of SSL/TLS, there is also a possibility of system failure due to server certificate expiration. The cause of this problem may come from human error or technical problem in system infrastructure. In this section, we present examples of past incidents involving inappropriate certificate update handling which ultimately resulted in system failures.

The first incident is the case study of a communication disturbance and services disruption of SoftBank, a large Japanese telecommunications company, on December 6, 2018 [1], which involved 36 million customers rendering them unable to make a phone call or accessing the Internet. As a result, SoftBank had to emergently downgrade the mobility management entity (MME) to the non-TLS version to cope with the situation.

Second, there is a malfunction in a device called "Unko Button" [2], an IoT device made by 144Lab [5] designed to keep track of new-born babies' stool information. In this incident, the device simply could not connect to the server due to the expired certificate. Generally, we can solve such kind of problem by updating the server certificate as usual. However, in the case of the mentioned device, the fingerprint of the server certificate was hard-coded inside the machine and could not be recovered in a typical way. Furthermore, when executing the Over the Air (OTA) update, the device was configured to forcefully use TLS without any alternative. As a result, the company had to recall all of its devices.

Lastly, there are two security reports, in 2018 and 2019, on an android and IOS application regarding lacking proper verification of server certificates. First, a MITM vulnerability was found on an older version (before version 3.0.0) of the Android application "NTV News24" [6] caused by lacking X.509 certificate verification from SSL servers. In addition, another report about MITM vulnerability was found on a LINE application (version 7.1.3 to 7.1.5) [7] on the IOS platform due to the same reason, i.e., no X.509 certificate verification.

2.3. Pinning. Pinning is a technique of embedding a server's certificate-related information to the applications or devices. This technique allows clients to determine the authenticity of the servers they are talking to without the risk of a MITM attack. There are several types of pinning depending on what information being stored (pinned) inside the client application/machine (e.g., CA certificates, end-entity certificates, and public keys). In this subsection, we discuss two pinning technologies: HTTP Public Key Pinning (HPKP, rfc7469 [8]) and DNS-Based Authentication of Named Entities (DANE, rfc6698 [9]).

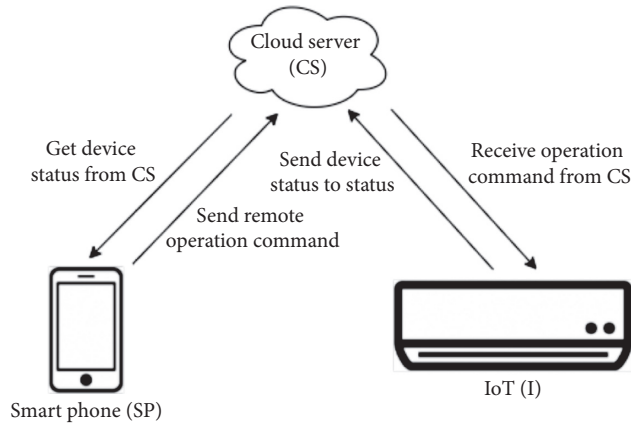


FIGURE 1: An overview of the cloud-centric IoT system architecture.

HPKP [8], started by Google and first developed on Chrome, is a pinning technique which pins servers' public keys information to clients using the following procedure. First, when the client connects to the server for the first time, the server sends a response back to the client with a special header. This special response header contains the public key (identical to the public key in the server certificate) and the information regarding how long the client should keep the pinned information. Therefore, after the first contact with the server, the client can now verify the certificate chain sent by the server by itself. This helps the client from being tricked by a fake/malicious server in the MITM attack. However, during the updating of the certificate information, if the updating of pinning information results in failure or problems, the client may no longer be able to connect to the server until the end of the mentioned period. Therefore, HPKP comes with such risk and operational difficulties. As a result, Google decided to abolish the support for HPKP from Chrome72 [10].

Next, DNS-Based Authentication of Named Entities (DANE) [9] is a technique that enables the utilizing of both domain name and certificates which are published for each domain. Utilizing DANE, the clients use DNS entry and TLSA Resource Record. The TLSA record contains four of the following fields: certificate usage, selector, matching type, and certificate association. The certificate usage field contains information regarding how to verify the certificate. It specifies the location where the clients will pin the information in the certificate chain and also the certificate verification method. A selector field refers to the entire certificate or "SubjectPublicKeyInfo" field in the certificate. Next, the matching type specifies whether the entire certificate association information or only a hashed version of it will be used. Lastly, the certificate association data field contains the raw data for verification; the data stored here are changed according to the information contained in the other fields.

Using DANE, clients verify the certificate with TLSA Resource Record (RR) received from the DNS server. In case that the verification process succeeds, the client will continue with the TLS handshake. Should the verification results in failure, clients will abort the process. To prevent the pinned

information from being tampered, DNSSEC [11] is usually implemented to overcome this problem. However, there is still a controversy about whether DNSSEC can truly solve the problems of DNS security or not. Also, there is a problem regarding the insufficient number of end-user making DNSSEC difficult to become widespread.

2.4. Related Work. Sánchez et al. [12] provides a comprehensive technical review of several security aspects of TLS and PKI focusing on the certificate pinning mechanism. In this paper, several attacks against SSL/TLS protocol such as MITM attack using insecure renegotiation (discovered by Rescorla, E. in 2009) and BEAST attack (discovered by Duong, T. and Rizzo, J. in 2011) are discussed. The paper points out that the length of the authentication key tends to become longer due to advancements in today's computing technology. To enhance the credibility of the PKI, the use of the pinning technique is introduced and analyzed in various aspects against multiple scenarios. As a result, the author concludes that DANE is currently the best choice for the pinning technique both in terms of its functions and the cost of management.

In [12], the paper presents a way to enhance the security of the system by utilizing public CA. On contrary, in this paper, we proposed the use of private CA to gain full control of the certificate issuing process and to strengthen the security of PKI-based IoT systems. In Sections 3 and 4, we discuss the risk associated with today's PKI-based IoT system in detail.

3. Risk Analysis for IoT Devices Using TLS

There are possibilities that some problems may arise if an IoT device tries to establish TLS communication with the cloud server after disconnecting from the network/Internet for a long time. IoT devices are generally produced in factories and then sold to customers via retailers. An example of a problem (so-called "dead stock") is given in which some currently unused/unsold IoT devices being kept in the inventory/warehouse of retailers for a long time.

The causes of the dead stock problem may vary from country to country. Also, it depends on the size of the company and its inventory management policies. For example, large retail enterprises and manufacturers, which are well-established and have existed for a long time, usually have a significantly lower risk of going bankrupt comparing to small venture companies due to differences in management and cash flow.

Business models of small manufacturers or retail companies are usually limited by their smaller cash flow encouraging them to lower the number of products in the inventory and urge them to sell their products more quickly. Although “dead stock” is not something desirable, big enterprises still have more options available. They can wait and keep their products inside the inventory for some time if they need to. Hence, the inventory problems are more likely to occur to these enterprises rather than small venture companies that are urged to sell things to generate cash flow very quickly.

When the IoT products are kept in the inventory or remained unused for a long time, these devices become old and can potentially cause several security issues due to their outdated security configurations.

An excellent example of a very recent vulnerability reported on Cisco devices in 2021 is given [13]. On January 2021, a vulnerability report was founded on several Cisco devices, i.e., Cisco RV110W Wireless-N VPN (2011), RV215W Wireless-N VPN Router (2012), and RV130 VPN (2014). A vulnerability found on these old Cisco devices allows an unauthenticated, remote attacker to execute arbitrary code on the target Cisco device due to improper validation of incoming UPnP traffic. Regarding this report, an attacker can exploit the vulnerability by sending crafted UPnP request to the affected target to execute arbitrary code as the root which can end up causing the affected device to reload, finally resulting in denial of service (DoS) condition [14]. In this example, the affected devices are old Cisco equipments which mostly are 7 to 10-year-old. Even though the information of the end-of-support date displayed on the Cisco website indicates that Cisco intends to support these devices for 3 more years (until 2024) [15], there are possibilities that these devices may be prone to future attacks after the end-of-support date.

Furthermore, in 2020, a vulnerability has been reported against an older model of Nintendo game console (i.e., Nintendo64). Exploiting such vulnerability allows attackers to execute malicious code on the mentioned game console. This vulnerability report is also a good example showing that even though the device is old, it still has a possibility of becoming a target of the attack.

Since the current generation of IoT device has relatively higher capabilities than electronic devices in the past, IoT devices have now become an attractive target for attackers. Concerning previously mentioned examples, we cannot deny the possibility that a similar situation may happen to these IoT devices in the future when they become old.

In this research, we studied problems which might occur when IoT devices did not connect to the network or the Internet for a long time, focusing on problems associated

with TLS certificates and their validity period. We first discuss the peculiar situation of IoT devices compared with the traditional devices, such as PC and smartphone, which usually are preinstalled with web browsers having TLS capability. Finally, we point out a problem that occurred in this situation and discuss the solutions to the problem.

First, during the TLS handshake, a web browser utilizes trust anchors stored in the device to perform a certificate path (so-called “certificate chain”) validation from an end entity to the root entity. Generally, these traditional devices (laptops, for example) that come with built-in browsers can connect to the network. These devices are usually connected to the network or Internet to utilize various services including obtaining new patches and updates. Hence, it is relatively safe to assume that these devices are usually connected to the network, and there is a slim chance that these devices are disconnected from the network for a very long time, e.g., 5–10 years.

Let us consider the case of an electrical shop where the traditional devices such as PCs and laptops are kept in boxes and stored in the inventory for a long period. Generally, OS supports for these devices are usually no longer than 10 years. Also, the average lifespan of PCs and laptops is approximately 3–5 years. Therefore, there is a relatively low possibility that the root certificate stored in these devices will be expired before the end of their lifespan. Therefore, we can assume that typical devices like laptops, smartphones, and PCs are not likely to face the problems of digital certificate expiration.

On the contrary, IoT devices are different. IoT devices usually consist of two primary functions: base functions and auxiliary functions. Base functions represent the main functions of the devices. Without these main functions, the IoT device can be considered useless or unusable. On the other hand, auxiliary functions are optional functions that are designed to deliver smarter and higher quality of services. For example, an IoT air conditioner’s primary function is to adjust (increase or decrease) temperature via cooling and heating process, while it may have an auxiliary feature allowing users to turn it on/off through the Internet.

Auxiliary features of IoT devices usually come in the form of network/internet-related services which allows users to control and monitor their own devices via the Internet. However, as we will see, many IoT devices can be used properly only with their base functions without using any auxiliary features. Therefore, some unused/unsold IoT devices that are kept in storage for a very long time may encounter security problems caused by expired root certificates, even though the devices can still properly function.

Typically, an IoT device communicates with a cloud server for two major reasons: (1) performing services and (2) conducting maintenance (e.g., firmware update). The firmware update is a way for enterprises to continue supports for their IoT products. The update is performed to provide the device with new or improved features or eliminate some existing problems including known vulnerabilities. In terms of security, a firmware update usually involves updating the trust anchor information stored inside an IoT device.

After some time, companies usually decide to stop providing supports to some of their older products resulting in the older model of IoT devices will no longer be able to receive any further updates (including the certificate update). At this point, the user has generally two main options: replace IoT devices with the newer version or keep using them.

Replacing all IoT devices in the system is theoretically a best practice in terms of security, especially for a home user since changing one or two IoT devices is relatively easy. However, the same may or may not hold true for enterprises looking from the business perspective where the cost of implementation and disruption of continuing services do matter. Furthermore, in the case of the Industrial IoT (IIoT) system in which a large number of IoT devices and sensors are deployed across the site/factory, replacing all devices and sensors just because the maker decided to stop providing the firmware support may not be an optimal choice since the devices are still practically usable. Although no further firmware update may lead to these devices being exposed to future vulnerabilities, however, after performing risk prioritization and business impact analysis, these companies may consider accepting the risk or decide to find another way to mitigate the risk without replacing all IoT devices.

Since the certificate issued by public CAs is relatively short-lived compared with certificates issued by private CA, this can cause service disruptions in some IoT devices rendering them unable to use their auxiliary features, not working properly, or even stop working completely after the certificate expires. On the other hand, utilizing a TLS certificate issued by private CAs with customizable expiration periods can help mitigate such problems. Figure 2 shows the comparison of the operational lifetime of an IoT device between the use of public and private CA's certificates. As we will see, in case of a certificate issued by the public CA (Figure 2(a)), the device is guaranteed to work properly until the root certificate expires. On the other hand, the certificate issued by a private CA offers more flexibility to this problem. Using private CA can help to significantly extend the operational lifetime for an IoT device (Figure 2(b)). Note that during this extended period, users can also decide for themselves whether to replace or not to replace their devices with the new ones.

In the following subsections, we perform a risk analysis by first discussing the problem and attack scenario against the previously mentioned scenario where an IoT device has not been connected to the network for a long period. Then, the details of risk identification and countermeasures are finally presented.

3.1. Attack Scenario. In this section, we discuss attack surfaces and problems that might occur in the scenario in which an IoT device has been disconnected from the network for a very long period. Figure 3 shows an overview of the IoT system utilizing public CA that we used in risk identification.

In this scenario, the IoT device is connected to the network after being unused for a very long time. The very first thing an IoT device does is to send a request to the NTP

server to acquire the current time information. The device will then try to contact the cloud server to receive the certificate chain information. Next, it verifies the certificate chain to ensure that the certificate signature is valid (correct and not expired) and also checks whether the Certificate Policy (CP) is matched and satisfies constraint conditions.

Since we focused on the scenario that an IoT device is not connected to the network for a long time, therefore, we classified the term "long time" into three cases: *A* (less than a year), *B* (longer than 1 year but less than 10 years), and *C* (more than 10 years). Table 1 shows the risks associated with each case.

According to Table 1, in case *A*, the period of disconnection is shorter than most software supporting periods, device lifespan, and root certificate lifespan. Therefore, there are no particular risks during this period. However, in the case of *B*, since the certificate validity period is approximately 1 year (according to Baseline Requirement), there is a risk that the end-entity certificate is already expired. Lastly, in period *C*, because the period is considerably long, there are chances that the TLS cipher suites become insecure or deprecated or the root CA's certificates are expired. For example, TLS 1.3 [16] was introduced after the elapse of ten years from the TLS 1.2 [17] release which comes with various changes in underlying handshake protocol and cipher suites. Furthermore, Cryptography Research and Evaluation Committees (CRYPTREC) initiated by the Japanese government also published a list of recommended cryptographic techniques called "e-Government Recommended Ciphers List" [18] which are expected to be secure within ten years.

3.2. Risk Identification and Countermeasure. In this section, risks and their countermeasure are explained and discussed. First, we identify risks and measures to be considered in case there are problems with the CA's root certificates rendering them to become invalid or unusable. These problems may be caused by the public CAs become bankrupt, go out of business, getting hacked, or the root private key is leaked or compromised.

An excellent example of security problems caused by an attack against public CA is the case of DigiNotar. DigiNotar is a Dutch certificate authority owned by VASCO Data Security International, Inc. founded in 1998. The company was responsible for issuing certificates to the private sectors and handled the PKI part of the Dutch government's e-government program called "PKIoverheid" (<https://cryptosense.com/wp-content/uploads/2014/11/black-tulip-update.pdf>). In 2011, an attacker performed unauthorized access to DigiNotar's CA server which allows the attacker to unlawfully get his/her hands on the private certificate information. As a result, all DigiNotar's certificates were revoked and the company finally went bankrupt in September 2011 [19].

To deal with this incident, the countermeasure to this problem is to update the trust anchor in every machine. Before updating a trust anchor, each device needs to perform a firmware update. However, the device also needs to

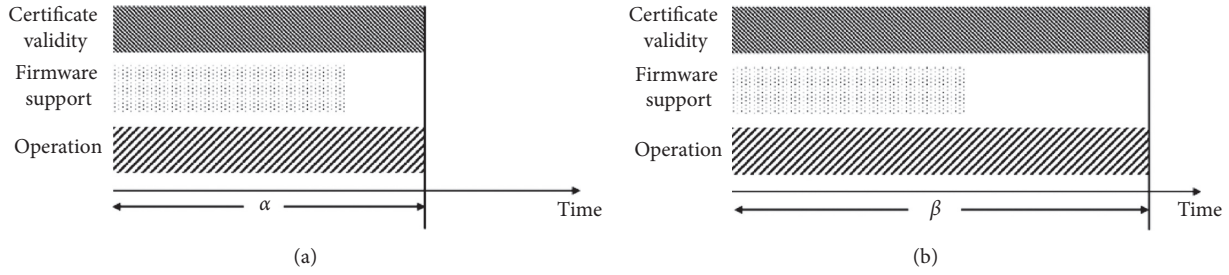


FIGURE 2: Operational lifetime of an IoT device.

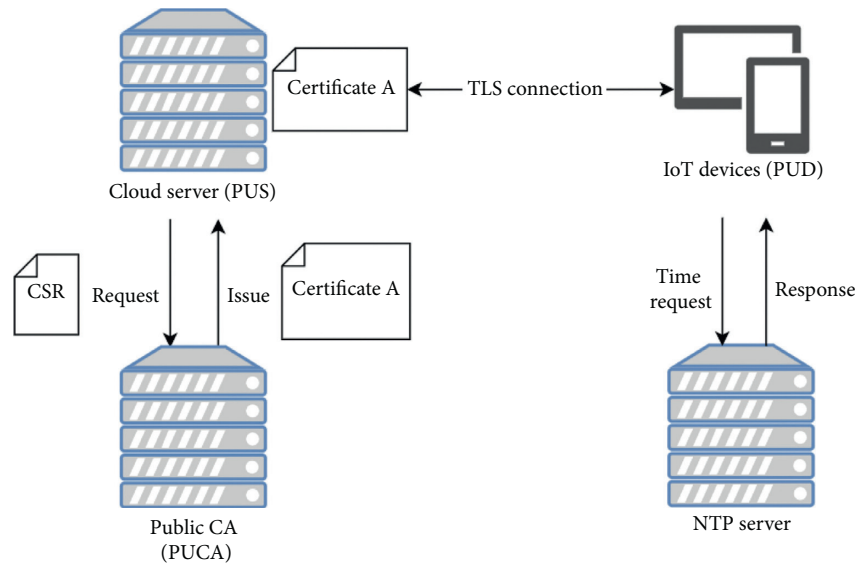


FIGURE 3: The system constitution using public CA.

TABLE 1: Risks associating with each period.

Period	A (less than a year)	B (less than 10 years)	C (more than 10 years)
Risk	None	The end-entity certificate is expired	TLS and its cipher suites are compromised Root certificate is expired

confirm the integrity of the firmware before the update is installed. In case that the integrity of the firmware is not guaranteed, there is a risk that the device might be running a tampered firmware making it vulnerable to some attacks. Therefore, the server is also required to perform code signing on the firmware to ensure its integrity. Next, there is a possibility the cipher suites become insecure. For example, some vulnerabilities may be found in some cryptographic algorithms rendering many systems and protocols utilizing such algorithms become vulnerable against particular types of attacks. A good example of vulnerability found in cipher suites is the case of an attack called the “Lucky 13” attack. Lucky 13 attack is a cryptographic timing attack against TLS protocol targeting the cipher block chaining (CBC) mode of operation, discovered by AlFardan, N.J. and Paterson, K. in 2013 [20]. Lucky 13 attack utilizes the padding that is not protected in CBC mode in the construction of integrity verification of TLS. Should an attacker succeed in executing the Lucky 13 attack, he/she would be able to decrypt the

message and obtain the confidential information inside. Fortunately, the researchers who found this attack exercised the vulnerability disclosure policy and worked with software vendors to the creating updates and patches to mitigate this problem and also made them available at the time of publication.

To deal with this type of problem (i.e., vulnerabilities found within cipher suites), the countermeasure in this situation is to update the firmware of each IoT device as soon as it connects to the network. When an IoT device tries to connect to the server using vulnerable cipher suites, the server should turn down the request/connection and allow the IoT device to download and install a new firmware to prevent any known vulnerabilities.

Lastly, there is also a risk associating with the use of network time protocol (NTP) [21]. Generally, an IoT device will attempt to query the time information from the NTP server as soon as it connects to the network. However, the authenticity of this time information becomes a critical issue

because it involves the verification of the certificate validity period. An attacker can execute a MITM attack by creating a fake NTP server and sending the manipulated time information to the clients (i.e., IoT devices).

To prevent the MITM attack, we need to consider the use of any reliable information other than the time information to prevent the IoT device from being tricked by the fake NTP server. GPS [22] information is considered a reliable choice for any IoT device equipped with GPS-related functions. If such functions are implemented, IoT devices also have an option to receive the time information from the satellites. Another way to overcome the mentioned problem is to use the pinning technique to store the public key information of the service server in advance. In this case, the IoT device can establish a secure connection with the cloud server using the pinned public key. After TLS handshake, the secure communication channel is established; the IoT device can then ask the cloud server for the correct time information.

Unfortunately, if the devices have already been affected by the MITM attack, the IoT device has no choice but to repeatedly try to reconnect to the real server. In case that, after many attempts, the client is still unable to establish secure communication with the real server, the IoT device is suggested to disconnect from the network and operate in an offline mode. If the IoT device somehow cannot establish a secure connection with the server and also cannot request the firmware update from the server, it is also advised to operate the device in offline mode.

4. Vulnerabilities in Today's PKI-Based IoT Security

In this section, the experiment and security analysis of today's PKI-based IoT security are tested and discussed. In this study, an experiment to show the vulnerability in today's IoT system was conducted. During the experiment, a man-in-the-middle (MITM) attack was carried out in the testing environment to demonstrate how a real IoT device with the improper implementation of security controls (i.e., PKI-related modules) may fall prey to such attacks. The following subsections present details of the testing environment and the results of the MITM attack against various IoT devices.

4.1. Testing Environment. There are four main components in the test system: target IoT devices (I), cloud server (CS), a fake (malicious) cloud server (FS), and a fake DHCP and DNS Server (FD). Figure 4 shows an overview of the test system.

According to Figure 4, the DHCP and DNS server was deployed using Raspberry Pi 3 (Model B, v1.2) [23] running dnsmasq [24] (v2.76) module with Raspbian 9 OS. The fake cloud server for executing a MITM attack is implemented using a PC running Nginx v1.18.0 [25] on Ubuntu 20.04.1 OS [26]. Lastly, the IoT devices component represents IoT devices from different vendors. Table 2 shows the details of IoT devices used for evaluation. There are five devices used during the test: a smart air conditioner, an air purifier, an IoT control device, a camera, and a smart plug. Each device

comes with a different TLS certificate validity period. Note that the specific details of each device (e.g., device model and manufacturer) in Table 2 cannot be disclosed and were intentionally omitted due to security reasons.

In this section, we demonstrate how to exploit weaknesses in IoT devices focusing on the MITM attack. All devices being tested are typical IoT home appliances. However, the same exploit can be found and can also be applied to the industrial field (i.e., IIoT) as well.

Under normal circumstances when the network connection between I and CS is active, the IoT devices can securely communicate and exchange information with the cloud server without a problem. On the other hand, the target IoT device is, however, considered prone to attack when the connection between itself and the cloud server is lost. In the following subsection, we demonstrate how a MITM attack can be executed under this condition.

4.2. MITM Attack. In this work, we assess the security of today's IoT system against a MITM attack. Generally, the IoT devices are expected to properly verify the certificate chain from an end entity to the root entity using the public key of each CA. However, there are also possibilities that some IoT devices available in the market are not practically doing this or do not verify each certificate within the chain properly. In this experiment, we found that some IoT devices verify the TLS certificate using only subject or issuer fields. This allows us to bypass the certificate verification of these IoT devices and finally complete the TLS handshake.

To demonstrate such an attack, first, the risk assessment against a MITM attack was conducted. In this assessment, an IoT device is considered to have a MITM-related vulnerability if the attacker can bypass the certificate verification and successfully complete the handshake process. There are two phases in performing a MITM attack: data collection and experimentation.

4.3. Data Collection. First, we begin the data collection phase by collecting cloud server (CS) related information (e.g., IP address) by observing network traffic between target IoT devices (I) and the server. To achieve this goal, the IoT devices are connected to the local access point (AP) which is also connected to the local network via a layer-2 switch (S1). Utilizing the port mirroring function of the network switch, the malicious server (PC) is connected to the mirror port on S1 to gather information. This allows the PC to observe the communication between I and CS. Using the obtained information with the client function of the TLS connection via OpenSSL [27], we can successfully retrieve the server certificates. Figure 5 shows the network topology used in the testing against the MITM attack.

4.4. Experimentation. Next, we set up the attack environment by first removing C2 from the layer-2 switch to simulate a situation where the connection to the cloud server C2 is not available. Next, we plug a fake DHCP and DNS server FD into the network. Then, a fake TLS certificate (FC)

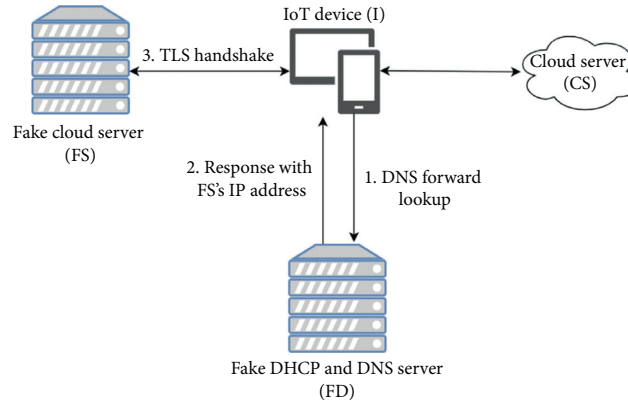


FIGURE 4: Overview of the test system.

TABLE 2: Details of IoT devices used during the test.

Corporate	A	B	C	D	E
IoT devices	Air conditioner	Air purifier	Controller	Camera	Plug
Issuer type	Private	Public	Public	Public	Public
Period of validity	24 years	2 years	2 years	1 year	1 year

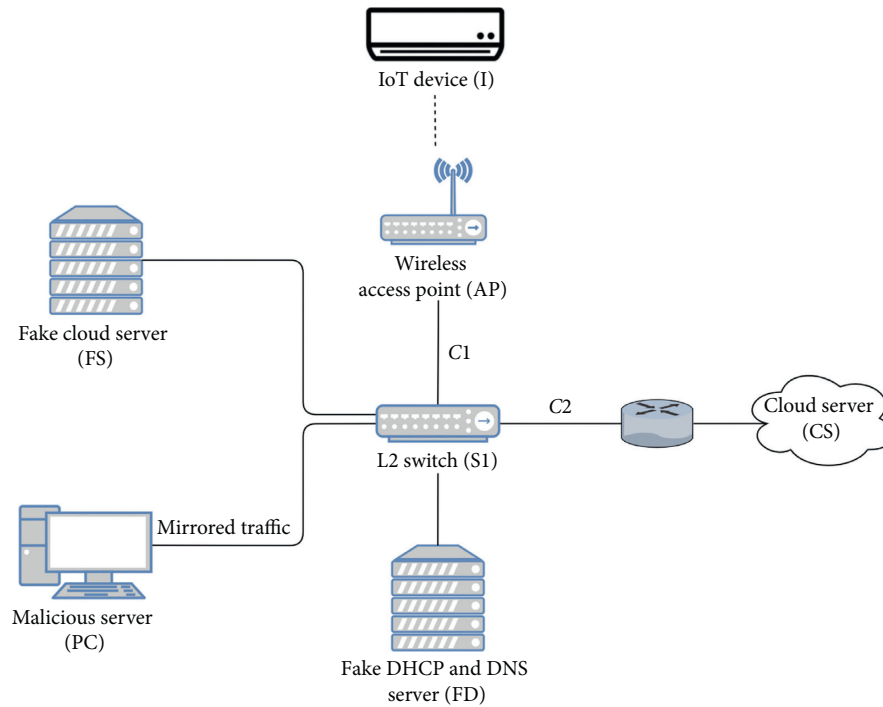


FIGURE 5: Network topology used for a MITM attack.

imitating the one retrieved from the cloud server is forged. This certificate FC will be used later to deceive the target IoT device.

The MITM attack begins with an IoT device sending a forward DNS lookup request to FD. The fake DNS server FD then replies with the IP address of the fake cloud server FS. Next after resolving the server IP, the IoT device will try to initiate a TLS handshake with the server (which is the fake one) by sending a “Client Hello” message. FS then replies to

the IoT device with a “Server Hello” message using the previously forged TLS certificate. This marks the completion of the MITM attack.

5. Results

During the data collection stage, we obtained the certificate information from the cloud server by observing the traffic coming in and out of the ethernet switch S1. Depending on

the type of certificate authority (CA) used to issue the certificate, the collected certificate information can be classified into two types: certificates signed by public CA and certificates signed by the private CA. Table 2 shows the type of certificates associated with each device. In this work, we generally focus on the difference between a certificate issued by public and private CA. According to the TLS Baseline Requirements document [3], the TLS certificate validity period is approximately 2 years. On the other hand, there is no such rule/requirement in the case of certificates signed by the private CA. In our experiment, the validity period of the certificate issued by private CA is set to roughly 24 years. Regarding the forged TLS certificate, Table 3 shows information that appeared on the certificate and also information of fields that can and cannot be imitated/manipulated. Table 4 shows the supported X509v3 [28] extensions on each device. As shown in Table 4, we can imitate generally most of the extensions with the exceptions of the one which requires CA information or the secret key information from the cloud server. Although we can not create a perfect copy of the certificate since it requires CA's private key to perform cryptographic operations, the information that appeared in the forged certificate is enough to deceive some IoT devices to believe that the certificate is real since some devices inadequately verify the authenticity of a certificate using only subject and issuer fields. Hence, as a result, we found that 2 devices (i.e., B and C) were prone to MITM because they allowed us to bypass the certificate verification and complete the TLS handshake, while the other 3 devices (A, D, and E) did not.

6. Proposed Mechanism

In this study, we proposed the use of private CA in the cloud-centric IoT architecture, which is shown in Figure 6. Using private CA allows companies to issue digital certificates themselves without using any trusted third party (i.e., public CA). This approach allows manufacturers to have full control over the digital certificates pinned to their devices.

There are three primary components in the proposed architecture. First, the IoT device (PRD) represents a cloud-connected IoT device. A certificate issued by the private CA and verification key (VKF) was pinned to this device at the manufacturing stage. The cloud server (PRS) and the private CA (PRCA) are operated by the company (i.e., IoT device manufacturer) that makes PRD. Using the certificate issued by PRCA, a secure communication channel between PRS and PRD can be established. Furthermore, the cloud server (PRS) uses the predefined signature key (SKF) during the firmware updating process. Regarding the private CA, PRCA is the private certificate authority owned and managed by the manufacturing company to provide security (via digital certificate) to the systems. Having a self-signed certificate, PRCA is responsible for issuing the certificate to PRS. Since PRCA does not have to operate according to the Baseline Requirement, PRCA can flexibly decide the validity period of the issued certificates.

In this work, first, PRS makes and sends Certificate Signing Request (CSR) to the private CA. PRCA then issues a

certificate B based on the obtained CSR. The manufacturer then pins PRCA's self-signed certificate and electronics signature verification key (VKF) to their products (e.g., IoT devices). When a user purchases these devices and turns them on, the IoT devices (PRD) will try to initiate a TLS secure communication with the cloud server (PRS) (i.e., sending a "Client Hello" message). The server then accepts the request and sends the certificate chain including certificate B back to the client. Next, PRD verifies the certificate chain received from the server using the pinned certificate (PRCA's self-signed certificate). In case that the device receives any certificate that is not signed by PRCA, the TLS connection is terminated.

Regarding firmware update, a verification key (VKF) is another key pinned to an IoT device at the time of manufacturing to be used for the firmware updating purpose. The use of a separate key pair for firmware updating operation allows the IoT devices to perform an update (including security updates) even another key pair is compromised.

6.1. Benefits. There are three main benefits of using a private CA in issuing certificates in the IoT system. First, there is a low possibility of errors caused by the expiration of the certificate during the firmware updates because the validity period of the certificate issued by the private CA can be customized. Second, there is no need to change the certificate chain in case of intermediate or root CAs having technical problems or being attacks. Lastly, the cost of issuing and updating is almost free, excluding operational and labor costs.

As mentioned in Section 2, there are incidents caused by a failure in updating certificate information which is mostly due to certificate expiration. To tackle this problem, we should make the validity period of the certificate in IoT-related system longer. This will reduce the number of times an IoT device needed to update the certificate during its lifetime which will also reduce the chance of errors that occurred during the updating process. Using the proposed method, the validity period of the certificate can be customized. Since the company owns the CA, therefore, it has full control over the certificate issuing process and the certificate specification.

On contrary, using the public CAs increases the attack surface that adversaries can utilize. For example, public root and intermediate CAs can now become the target of attacks. On the other hand, if certificates are signed by the private CA and are kept as trust anchors, the range of attack targets can be reduced.

OpenSSL allows the manufacture to build the private CA for free because the software is open-source software (OSS). The firmware update server can prevent the attackers from tampering with the firmware updating process by using the client verification. In case the attackers can get their hand on the firmware, they can freely analyze and discover vulnerabilities within the firmware. There is a report of an OS command injection attack on the firmware of some web cameras in CVE-2013-1599 [29]. Thus, to prevent the system

TABLE 3: Contents of the certificate which can be imitated.

Field of certificates	Can be imitated/manipulated?
Version	Yes
Serial number	Yes
Signature algorithm	Yes
Issuer	Yes
Validity	Yes
Subject	Yes
Subject public key info	Partially (except the public key information)
X.509v3 extensions	Almost everything (except the fields that require a secret key or CA certificate information)
CT precertificate timestamp	No

TABLE 4: X.509v3 extensions.

X.509v3 extensions	A	B	C	D	E
Authority key identifier	×	×	×	×	×
Authority information access	—	○	○	○	○
Subject alternative name	—	○	○	○	○
Certificate policies	—	○	○	○	○
Extended key usage	—	○	○	○	○
CRL distribution points	—	○	○	○	○
Key usage	○	○	○	○	○
Subject key identifier	×	×	×	×	×
Basic constraints	—	—	○	○	○

“○” and “×” indicate fields that can and cannot be imitated, respectively, while “—” represents fields that are not being used/presented in certificates of some particular devices.

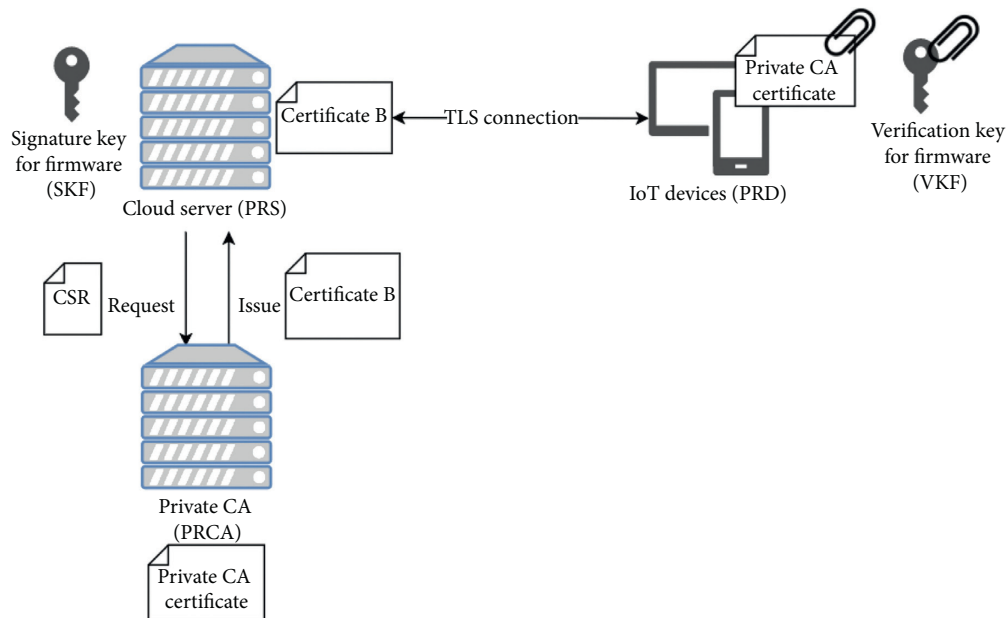


FIGURE 6: An overview of the proposed framework.

from such risk, the firmware must be downloaded only by the authorized/intended IoT devices. To achieve this goal, TLS client verification is considered an efficient way to overcome the problem. TLS client verification allows the server to verify whether the client is actually who it is claimed to be. In this case, each client (i.e., IoT device) must have client certificates. However, the cost of implementing such an approach is considered very expensive (or almost practically impossible in the IoT industry), if the

manufacturer decides to use public CA for issuing client certificates. However, issuing client certificates can be done for free (or much less cost) by using the private CA owned by the company.

6.2. *Drawbacks and Limitations.* On the downside, the proposed method of using a private CA in the IoT system suffers from two main drawbacks. First, the cipher suites or

cryptographic algorithms used in the system may become deprecated or vulnerable to attacks. In the proposed system, there are possibilities that the vulnerabilities of the underlying cryptographic methods will be discovered during the lifetime of an IoT device since the certificate validity period is long.

Second, from the user perspective, there are also reliability issues when using private CA. Generally, the public root CAs gain many trusts because the trust anchor at the top of the certificate chain is operated under the root CA certificate program. For instance, Mozilla Root Store Policy [30] requires the CAs to get audited by third-party organizations such as Web of Trust for CA [31] and ETSI [32]. However, regarding the proposed system, since the private CA is used, the companies are not required to get audited. Therefore, the proposed method still requires good and secure implementation of the entire system to make it practical.

7. Security Analysis and Discussion

In this section, the security aspects of the proposed method are analyzed and discussed. Table 5 shows a comparison between the proposed method and traditional approaches. We compare our proposed method with two different architectures: A1 and A2. Table 5 shows the comparison between the proposed architecture and the existing techniques.

In the first architecture A1, public CA is employed. The public CA issues the server certificate conforming to the Baseline Requirement [3]. When the cloud server and the IoT device try to create a secure connection, the IoT device verifies the certificate chain received from the server. Similar to the first architecture, the second architecture A2 utilizes public CAs to issue digital certificates. However, in this architecture, the manufacturer pins verification key information inside the IoT device at the time of manufacture. This verification key information is used to verify code signing to ensure the integrity of firmware during the firmware update process.

The proposed method, on the other hand, uses the private CA to issue server certificates. These certificates have a longer validity period while they are also not required to follow the Baseline Requirement. Moreover, the self-signed certificate of the private CA is also pinned to the products (i.e., IoT devices) to help to prevent a MITM attack. Lastly, in the proposed architecture, a verification key for code signing is also stored inside each IoT device, similar to the A2 case. The following subsections discuss each criterion used in the comparison in detail.

7.1. Issue Cost. Generally, a certificate is required as proof of identity when communicating with TLS. Companies usually have to pay trusted third-party companies to issue their certificates. Therefore, A1 and A2 cost a lot more money than the proposed method to successfully implement the system. Besides, companies will have to pay more money if they need more certificates in the future. In contrast, the proposed method offers a cheaper way to deal with the

certification fee problem using private CA. Also, the private CA can be implemented using free open-source software, e.g., OpenSSL. Since the company has a CA of its own, therefore, the company can issue any number of certificates cheaply without additional cost.

7.2. External Factors Risk. There are possibilities of external factor risk in which the companies may not be able to directly control, for example, disclosure of CA's secret key and so on. In this case, since A1 and A2 both rely on external organizations (i.e., public CA) to handle all their certification-related issues, therefore, there is a higher external factor risk in A1 and A2 than the proposed method in which private CA is employed.

7.3. Attack Surfaces (Number of CAs That Can be Targeted). A certificate issued by a public CA usually involves the issuance of certificates by many trusted organizations. These trusted parties can be attacked and compromised. Therefore, the case of A1 and A2 has a higher risk of being attacked due to a larger number of CAs that can be targeted. On the other hand, the proposed method has a very small number of CAs in the entire system, typically only one CA for a small company. Hence, the attack surface, in terms of the number of CAs that can be attacked, is relatively small compared with A1 and A2.

7.4. Risk of Certificate Update Failure. There are some incidents in which a client cannot connect to the server using TLS because the administrator of the server failed to update or renew the server certificate. Countermeasures of such incidents involve reviewing operational policies and the use of automatic updates. However, no matter what measures a manufacturer take, there is always a chance of making a mistake during an update. Therefore, the most effective measures are to reduce the number of times the manufacturer has to update the certificate by issuing the certificate with a longer validity period than the IoT device lifetime.

In this case, A1 and A2 cannot use such countermeasure since both of them use public CA to issue certificates which generally have much shorter validity than the lifespan of the IoT device. On contrary, the proposed method can issue a unique certificate with a long expiration period by using the private CA.

7.5. Risk of Firmware Tampering. Sometimes, an IoT device may be required to perform a firmware update to add some new features or for security reasons. During the firmware updating process, attackers may find a way to tamper with the firmware to install backdoors or create other vulnerabilities to the IoT device. Therefore, the integrity of the firmware used during the update is the utmost crucial factor needed to be concerned.

Code signing can help ensure the integrity of the firmware during the update process. Since the first architecture A1 does not have a verification key and does not utilize code signing; thus, A1 is considered prone to such

TABLE 5: Comparison between the proposed architecture and the traditional approaches.

Factors	A1	A2	Proposed architecture
Issue cost	High	High	Low
External factors risk	High	High	None
Attack surfaces (number of CAs that can be targeted)	High	High	Low
Risk of failure during certificate update	High	High	Low
Risk of firmware being tampered	High	Low	Low
Management cost (operate, labor cost)	Low	Low	High
CA reliability	High	High	Relatively low
Internal factor risk (insider threats)	Low	Low	Relatively high
Verification time	Slow	Slow	Fast

attack. On the other hand, both A2 and the proposed method have a lower risk against such attacks due to having a verification key pinned to each device at the time of manufacture.

7.6. Management Cost (Operational and Labor Cost). If the company decides to implement a private CA of its own, they unavoidably have to pay additional costs, i.e., operational and labor costs. Since A1 and A2 use public CAs to issue their server certificates, therefore, all operational and labor costs are already included in the amount they pay to the third-party companies from the beginning. Therefore, there is no extra cost in the case of A1 and A2. The proposed method, however, have to pay a full price of implementing a private CA including server fee and some hardware security modules (HSMs) for protecting the secret key. Hence, the proposed method is considered more expensive to manage and maintain than the other architectures.

7.7. CA Reliability and Internal Factor Risk. Many public CAs are well-known in terms of security and reliability because many of them are required to periodically perform security auditing to ensure the security of the system. For this reason, public CA is generally trusted by many companies and organizations. However, it is not likely to be the case for the private CA. Since the security of the private CA depends on each company implementing the system, therefore, it is very difficult to determine whether the private CA of which company is secure or insecure unless it performs security auditing conforming with the industrial standard.

7.8. Verification Time. Generally, many IoT devices are facing the problem of limited computational power due to energy constraints. This causes a time-lag and slower computational speed in IoT devices, comparing to traditional devices such as laptops or smartphones. Hence, one of the primary goals in designing and implementing an IoT security control is to limit resource consumption. Regarding the certificate verification process, end-entity certificates are generally issued by the intermediate CA. Also, the certificate of this intermediate CA is issued by another intermediate or the root CA. This process forms the concept of a certificate chain in which the destination server sends such hierarchical information (a.k.a. certificate chain) to the device (including

IoT device). Upon receiving the certificate chain, the device verifies the validity of this certificate chain starting from the end-entity certificate to the root CA certificate.

The verification of a certificate chain usually involves the decryption of each certificate within the chain. Therefore, the number of operations required to verify a certificate chain increases in proportion to the length of the chain. Asymmetric-key cryptographic operations (e.g., decryption) are usually computationally expensive. Thus, verifying several certificates, especially in the case of a complex certificate chain, can pose a significant challenge for the IoT. On contrary, using a private certification authority allows the end-entity certificate to be issued directly from the root CA, except when the company also utilizes its own intermediate CA. This results in a flatter hierarchical structure of the certificate chain which can significantly help to reduce computational power consumption and time for verification in the computational resource-limited IoT devices.

Furthermore, using private CA with pinning technique can help getting rid of some unnecessary processes such as checking certificate validity through OCSP or checking against the certificate revocation list (CRL). Although this reduction of unnecessary complexity is not a major factor in improving the efficiency of the system, it can still help to save some computational and network resources which can ultimately end up improving the overall performance of the IoT.

8. Conclusion

In this paper, we study the standard secure communication model of the cloud-centric IoT architecture using TLS and PKI. We identify the security flaws and vulnerabilities of the public CA-based IoT architecture due to errors in the certificate verification process. Next, risk identification and countermeasures regarding the situation when an IoT device tries to connect to the network after disconnecting for a very long period are discussed. Following with a simulation of a MITM attack, we demonstrate an attack on actual devices available in the IoT market. As result, it is shown that using public CA may not be an optimal solution for IoT. To overcome such a problem, finally, we proposed the use of private CA together with the separate verification key for firmware updating. Lastly, the benefits, drawbacks, and design limitations of the proposed method are discussed and compared with today's existing PKI-based approaches.

Data Availability

This research and all experiments contain information on vulnerabilities in real IoT devices available in the market. Publicizing this sensitive information will allow adversaries to take advantage of the mentioned vulnerabilities, which can affect a large number of users. Due to this reason, we decided to exercise a nondisclosure policy on the experimental data of this research.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

- [1] SoftBank Corp, *SoftBank Press Conference*, SoftBank Corp, Tokyo, Japan, 2018, <https://www.youtube.com/watch?v=Gu7xMClCbM>, in Japanese.
- [2] 144Lab Co., Ltd., *144Lab: Unko Button*, 144Lab Co., Ltd., Tokyo, Japan, 2020, <https://unkobtn.com/>, in Japanese.
- [3] CA/Browser Forum, “Baseline requirements for the issuance and management of publicly-trusted certificates version 1.7.3,” 2020, <https://cabforum.org/wp-content/uploads/CA-Browser-Forum-BR-1.7.3.pdf>.
- [4] Ministry of Internal Affairs and Communications Japan, “The evolving digital economy and society 5.0 beyond,” in *White Paper on Information and Communication* Ministry of Internal Affairs and Communications Japan, Tokyo, Japan, 2019, <https://www.soumu.go.jp/johotsusintokei/whitepaper/ja/r01/pdf/01honpen.pdf>, in Japanese.
- [5] 144Lab, *Light Up Your Science*, 144Lab, Tokyo, Japan, in Japanese, 2020.
- [6] National Institute of Standards and Technology, “CVE-2019-6032,” in *National Vulnerability Database* National Institute of Standards and Technology, Gaithersburg, MD, USA, 2019, <https://nvd.nist.gov/vuln/detail/CVE-2019-6032>.
- [7] National Institute of Standards and Technology, “CVE-2018-0518,” in *National Vulnerability Database* National Institute of Standards and Technology, Gaithersburg, MD, USA, 2018, <https://nvd.nist.gov/vuln/detail/CVE-2018-0518>.
- [8] C. Palmer, R. Sleevi, and C. Evans, *RFC 7469-Public Key Pinning Extension for HTTP*, Internet Engineering Task Force (IETF), Fremont, CA, USA, 2015, <https://tools.ietf.org/html/rfc7469>.
- [9] J. Schlyer and P. Hoffman, *RFC 6698-the DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS) Protocol: TLSA*, Internet Engineering Task Force (IETF), Fremont, CA, USA, 2012, <https://tools.ietf.org/html/rfc6698>.
- [10] Google LLC, “Google, remove HTTP-based public key pinning (removed),” 2020, <https://www.chromestatus.com/feature/5903385005916160>.
- [11] R. Austein, M. Larson, D. Massey, S. Rose, and R. Arends, *RFC 4033-DNS Security Introduction and Requirements*, IETF Network Working Group, Fremont, CA, USA, 2005, <https://tools.ietf.org/html/rfc4033>.
- [12] D. Sánchez, A. M. Lopez, F. A. Mendoza, P. A. Cabarcos, and R. S. Sherratt, “TLS/PKI challenges and certificate pinning techniques for IoT and M2M secure communications,” *IEEE Communications Surveys and Tutorials*, vol. 21, no. 4, pp. 3502–3531, 2019.
- [13] National Institute of Standards and Technology, “CVE-2021-1360,” in *National Vulnerability Database* National Institute of Standards and Technology, Gaithersburg, MD, USA, 2021, <https://nvd.nist.gov/vuln/detail/CVE-2021-1360>.
- [14] Cisco systems, Inc., *Cisco Small Business RV110W, RV130, RV130W, and RV215W Routers Remote Command Execution and Denial of Service Vulnerabilities*, Cisco systems, Inc., San Jose, CA, USA, 2021, <https://tools.cisco.com/security/center/content/CiscoSecurityAdvisory/cisco-sa-rv-overflow-WUnUgv4U>.
- [15] Cisco systems, Inc., *Cisco RV130W Wireless-N Multifunction VPN Router*, Cisco Systems, Inc., San Jose, CA, USA, 2021, <https://www.cisco.com/c/en/us/support/routers/rv130w-wireless-n-multifunction-vpn-router/model.html>.
- [16] E. Rescorla, *RFC 8446-the Transport Layer Security (TLS) Protocol Version 1.3*, Internet Engineering Task Force (IETF), Fremont, CA, USA, 2018, <https://tools.ietf.org/html/rfc8446>.
- [17] T. Dierks and E. Rescorla, *RFC 5246-TLS Protocol Version 1.2*, IETF Network Working Group, Fremont, CA, USA, 2020, <https://www.ipa.go.jp/security/rfc/RFC5246EN.html>.
- [18] National Institute of Information and Communications Technology and Information-Technology Promotion Agency Japan, *CRYPTREC Report 2019*, Information-Technology Promotion Agency Japan, Tokyo, Japan, 2020, <https://www.cryptrec.go.jp/report/cryptrec-rp-2000-2019.pdf>.
- [19] B. V. Fox-IT, “Black tulip report of the investigation into the DigiNotar certificate authority breach,” 2012, https://www.researchgate.net/publication/269333601_Black_Tulip_Report_of_the_investigation_into_the_DigiNotar_Certificate_Authority_breach.
- [20] N. J. AlFardan and K. G. Paterson, “Lucky thirteen: breaking the TLS and DTLS record protocols,” in *Proceedings of the IEEE Symposium on Security and Privacy*, pp. 526–540, Berkeley, CA, USA, May 2013.
- [21] D. Mills, U. Delaware, J. Martin, J. Burbank, and W. Kasch, *RFC 5905-Network Time Protocol Version 4: Protocol and Algorithms Specification*, Internet Engineering Task Force (IETF), Fremont, CA, USA, 2020, <https://tools.ietf.org/html/rfc5905>.
- [22] National Coordination Office, *GPS: the Global Positioning System*, National Coordination Office, Washington, DC, USA, 2020, <https://www.gps.gov/>.
- [23] Raspberry Pi Foundation, *Raspberry Pi 3 Model B*, Raspberry Pi Foundation, Cambridge, UK, 2020, <https://www.raspberrypi.org/products/raspberry-pi-3-model-b>.
- [24] S. Kelley, “Simon kelley: Dnsmasq,” 2020, <http://www.thekelleys.org.uk/dnsmasq/doc.html>.
- [25] Nginx, Inc., “Nginx,” Nginx, Inc., San Francisco, CA, USA, 2020, <https://nginx.org/en>.
- [26] Canonical Ltd., *Ubuntu 20.04.1 LTS (Focal Fossa)*, Canonical Ltd., London, UK, 2020, <https://releases.ubuntu.com/20.04>.
- [27] OpenSSL Software Foundation, *The OpenSSL Project: OpenSSL Cryptography and SSL/TLS Toolkit*, OpenSSL Software Foundation, Adamstown, MD, USA, 2020, <https://www.openssl.org>.
- [28] S. Santesson, S. Farrell, S. Boeyen, R. Housley, W. Polk, and D. Cooper, *RFC 5280-Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*, IETF Network Working Group, Fremont, CA, USA, 2008, <https://tools.ietf.org/html/rfc5280>.
- [29] National Institute of Standards and Technology, “CVE-2013-1599,” in *National Vulnerability Database* National Institute of Standards and Technology, Gaithersburg, MD, USA, 2013, <https://nvd.nist.gov/vuln/detail/CVE-2013-1599>.

Database National Institute of Standards and Technology, Gaithersburg, MD, USA, 2020, <https://nvd.nist.gov/vuln/detail/CVE-2013-1599>.

- [30] Mozilla, *Mozilla Root Store Policy Version 2.7*, Mozilla, Mountain View, CA, USA, 2020, <https://www.mozilla.org/en-US/about/governance/policies/security-group/certs/policy>.
- [31] CPA Canada, *Principles and Criteria and Practitioner Guidance*, CPA Canada, Toronto, Canada, 2020, <https://www.cpacanada.ca/en/business-and-accounting-resources/audit-and-assurance/overview-of-webtrust-services/principles-and-criteria>.
- [32] European Telecommunications Standards Institute (ETSI), *Welcome to the World of Standards!*, European Telecommunications Standards Institute (ETSI), Sophia Antipolis, France, 2020, <https://www.etsi.org/>.