IJNS
International Journal of Network Security

ISSN 1816-353X (Print)
ISSN 1816-3548 (Online)

Vol. 17, No. 5 (Sept. 2015)

INTERNATIONAL JOURNAL OF NETWORK SECURITY

PUBLISHING OFFICE

Min-Shiang Hwang
Department of Computer Science & Information Engineering, Asia University, Taichung 41354, Taiwan, R.O.C.
Email: mshwang@asia.edu.tw

International Journal of Network Security is published both in traditional paper form (ISSN 1816-353X) and in Internet (ISSN 1816-3548) at http://ijns.jalaxy.com.tw

PUBLISHER: Candy C. H. Lin

International Journal of Network Security

Vol. 17, No. 5 (Sept. 1, 2015)

A Survey of Digital Evidences Forensic and Cybercrime Investigation Procedure

Jia-Rong Sun[1], Mao-Lin Shih[2], Min-Shiang Hwang[1,3]
(Corresponding author: Min-Shiang Hwang)

Department of Computer Science and Information Engineering, Asia University[1]
(Email: mshwang@asia.edu.tw)
Department of Financial and Economic Law, Asia University[2]
No. 500, Lioufeng Rd., Wufeng, Taichung 41354, Taiwan
Department of Medical Research, China Medical University Hospital, China Medical University[3]
No. 91, Hsueh-Shih Road, Taichung 40402, Taiwan
(Received Jan. 5, 2015; revised and accepted Apr. 10 & May 2, 2015)

Abstract

Due to the development of networks, cybercrime has many crime types, including network attack, mail fraud, intimidation, copyright infringement, and so on. For network attacks, many approaches have been proposed and used to detect and defense. However, after the network attack is confirmed or other crime exists, it still need to execute the investigation procedure by the investigators, collect the evidences related to the crime, find the perpetrators, and prosecute them. Therefore, in this paper, we collect the researches of investigation procedure of cybercrime in the recent years. By introducing the research investigation procedure of these papers, we will discover the features of every procedure. Then we compare these investigation procedures via the traditional investigative procedures compatibility, cybercrime behavior analysis, evidence forensic procedures, case analysis and verification, the methods of evidence collection and analysis, and the area of judicial jurisdiction. Finally, we will propose the viewpoints of cybercrime investigation and forensic procedures, and we wish this paper will help the research of investigation and forensic procedures.

Keywords: Cybercrime, digital evidence, forensic procedure, investigation procedure

1 Introduction

In the recent years, many approaches used to detect the network attacks have been proposed [9, 11, 14, 20, 21, 22, 28, 29, 30]. By using these approaches, we can detect the network attack occurring, and defense the attacks. However, after the network attacks occurred, these attack events will be called cybercrime. Investigating these cybercrimes not only pursue the liability of criminal, and also combine the detection approaches to become an investigation strategy of cybercrime, reducing the damage from same criminal behavior.

In the cybercrime, the investigation procedures can be divided into two main parts, digital evidence forensics process, as well as cybercrime investigation procedure. In the cybercrime cases, since the properties of evidence unnecessarily exist at the entity type, perhaps they are digital data and stored in the data storage devices. The existence locations of digital evidence will be different because of the type of crime. For example, in wireless networks of cybercrime, digital evidences will exist in the record of a computer and network equipment in the offenders and the victims [35]; in the net-work attacks, digital evidences will exist in the ISP server and the computers of offender [16]. The digital evidence collecting aims to find any evidences related to cybercrime, and preserve these evidences to avoiding the digital evidences were forged, altered, deleted or destroyed. The purpose of digital evidence collected is to investigate the process of cybercrime occurred. Therefore, the process how to find the digital evidences and the perpetrators is called a criminal investigation procedure. And the criminal investigation procedure includes the procedure of forensics evidence. When a cybercrime is occurred, collecting the digital evidences, proving the existence of criminal behavior, finding identify of suspects, and proving the causation are called the cybercrime investigation procedure. In the following, we will define the cybercrime, investigation procedure, and the nature of digital evidence.

1.1 The Definition of Cybercrime

The cybercrime is a social problem derived from the social development. In [12], the cybercrime is defined to a 'digital' or 'hi-tech' crime type, or uses network technology as the primary or secondary tools of crime [3, 23, 27, 31, 34]. In [33], the authors consider the difference between tradi-

tional crimes and cybercrimes is the evidences of cybercrime scene belonging to an electronic format. In Taiwan, the cybercrime is also defined in the Criminal Code definition of a computer crime in Chapter 36 of the legislative purpose. In the broad sense, the computer crime refers the crime tool or process to involve the computer or Internet; in the narrow sense, the signification of computer crimes referring to the criminal objects of attack are the computers or Internet. In summary, we consider the cybercrime must use some tools to connect Internet, and carry out the illegal behaviors of offense. The evidences of this cybercrime produced has a part belonging to the digital evidence, and no fixed location of the crime, and the offender and victim does not need to face each other directly.

1.2 The Property of Digital Evidence

The type of evidence can be divided into witnesses, physical evidence and documentary evidence. The witnesses are an evidence of personal experience, but does not include speculation. The witnesses includes witnesses, victims, defend-ants or expert testimony; the physical evidence refers an object or state which can be used to prove facts of the crime, such as the tools of crime; documentary evidence refers to the content of a file which can be used as evidence, such as written report of victims. Furthermore, there is some evidences including both characteristics of documentary evidence and physical evidence, which is the evidence of cybercrime. The evidence of cybercrime belongs to a new type of evidence, called Digital evidence [4, 5, 6]. The witnesses may be changed with time or interfered by other factors, and the physical evidence and documentary evidence is relatively easy to leave the traces of modification. Therefore, under the normal circumstances, the probative force (i.e. credibility) of physical and documentary evidence are higher than witness evidence. Digital evidence is stored in data storage devices generally [33] via the electromagnetic record type, and the content of digital evidence can be understood through printing, playing, and execution, etc. From the foregoing, the digital evidence has both characteristics of physical evidence and documentary evidence. In addition, since the digital evidence exists by the electromagnetic record, it has the following features: easy to modify and copy [1, 4, 33], hard to understand the content directly without the conversion process [4, 7], and not easy to retains the original state [1, 4, 33].

1.3 The Definitions of Investigation Procedure

The difference countries have their own judicial investigation procedures based on the law of themselves [13, 26]. In Taiwan, the crime investigative procedures are prescribed in the Criminal Procedure Law. The purpose of these procedures are to investigate the facts of crime, collect evidence, find the suspects, and arrest the suspects. In addition, the types of criminal cases are divided into public prosecution and private prosecution in Taiwan, and this classification will affect the start of investigation procedure. The public prosecution event needs the victims to report the crime event to police or the judiciary to accept this criminal case; private prosecution event refers that the crime does not need to wait the report of victim, and the judicial investigators can investigate this types of crime case actively. These two types will affect the investigation procedure is started actively or passively by the judicial investigators. The start of investigation must be a legal process, otherwise this case will not be accepted by the court after the prosecution. When the investigation procedure is initiated legal, the suspects will be found via the evidences of legal collect. After summoning and asking the suspects, the innocent people will be released and the criminal will be arrested. Finally, the criminal will be prosecuted.

In this paper, we collect and survey the papers of cybercrime investigation procedures from different countries in re-cent years. First, we will introduce the architecture, processes, and forensics procedures of these investigations. Then we will compare these investigative procedures, including the traditional investigative procedures compatibility, cybercrime behavior analysis, evidence forensic procedures, case analysis and verification, the methods of evidence collection and analysis, and the area of judicial jurisdiction. Finally, we will propose the viewpoints of cybercrime investigation and forensic procedures, including the digital evidence forensic and the investigation procedure. This paper is organized as follows. In Section 2, we will introduce the proposed approaches of investigation procedures and evidence forensic in cybercrime; in Section 3, we will compare each investigation procedures, and propose our viewpoints of cybercrime investigation procedures; finally, we will draw our conclusions in Section 5.

2 The Survey of Cybercrime Investigation and Forensics Procedure

Cybercrime is a crime type produced from the development of Internet. According to the definition of cybercrime, the evidences of cybercrime include digital evidences, cybercrime has no fixed location of the crime, and the offender and the victim of cybercrime do not need to face each other directly. Therefore, the content of cybercrime investigation procedure must contain the methods including to find the real perpetrators, digital evidence forensic, and analysis of crime. In addition, the investigators is not limited to use only one method in the cybercrime investigation, and they will use many methods to collect evidences and identify the perpetrators as long as the methods is not illegal. Therefore, if these are proposed cybercrime investigation procedures, they can be used to

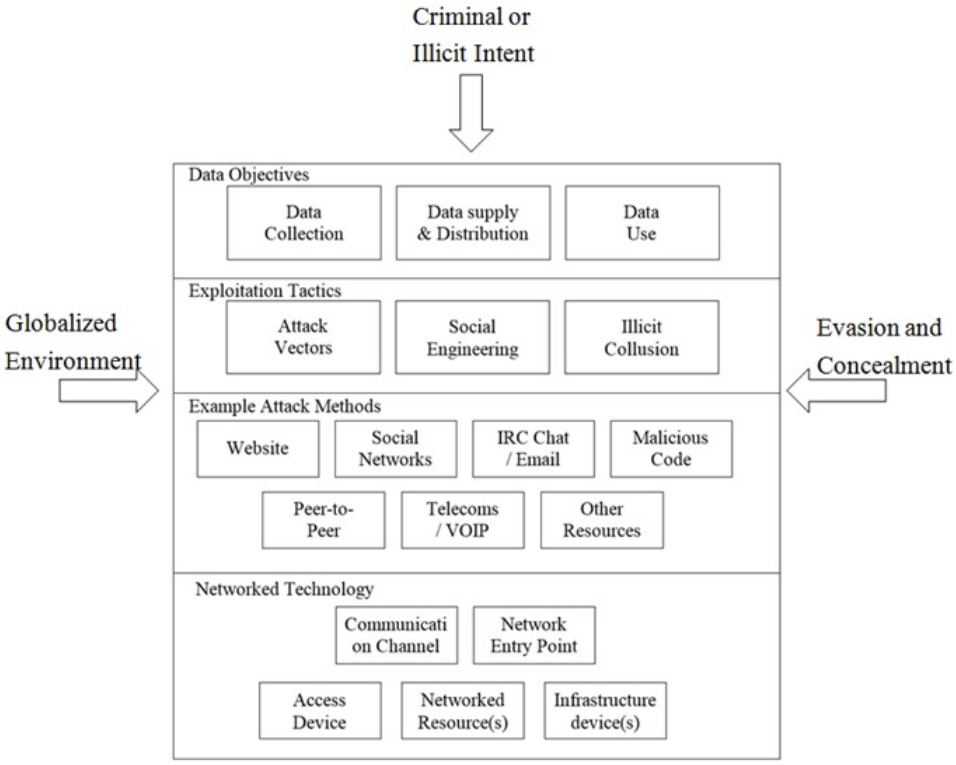


Figure 1: The cybercrime execution stack

find the real perpetrators, collect evidences, and analyze the method of cybercrime, so this procedure will be referenced and used by the investigators. In the following, we will describe the proposed cybercrime investigation procedures.

2.1 The Growing Phenomenon of Crime and Internet

In this paper [12], the authors proposed and defined a cybercrime execution and analysis model. The purpose of this paper is making the conventional policing models more easy use to investigate cybercrime, and help the investigators plan investigations. The investigation of cybercrime model is defined to a Cybercrime Execution Stack in this paper. This model is affected by three factors, including Criminal or illicit intent, Globalized Environment, and Evasion and Concealment [12]. In the different countries, the Criminal or illicit intent of cybercrime is stipulated in their own criminal law, and it will affect whether the offense is founded or not. The factor of Globalized Environment will affect the extent of offense in different countries. If a cybercrime crosses several area of judicial jurisdiction, the extent of offense may be different, or violate the different codes of law. Since the Internet has anonymity, the behavior of evasion and concealment in the crime will increase the difficulty of crime investigation and information collection. Therefore, the evasion and concealment of cybercrime also are the one of affection factors in cybercrime investigation. In the Cybercrime Execution Stack, as the Figure 1, it has 4 main stacks, including Data Objectives, Exploitation Tactics, Example Attack Methods, and Networked Technology [12]. According to the basic function of network technology, Data Objectives can be divided into groups: data collection, data supply and distribution, and data use [12]. The cybercrime tactics will be found out from the target type of attacks and the criminal behavior. Therefore, in the Exploitation Tactics it includes three groups: Attack Vectors, Social Engineering and Illicit Collusion. In the above Exploitation Tactics, it can produce lots of different attack methods, and the Attack Vectors include malware, Trojans, spyware, worms or viruses; Social Engineering includes impersonation, email, phishing, blogs or social networking; Illicit Collusion includes private websites, email, Internet Relay Chat (IRC), Peer-to-Peer data sharing. Finally, the Networked Technology is used to find and collect the evidences and information of cybercrime. These technical characteristics is communication channel, network entry point, access device, network resources, and the infrastructure devices.

2.2 The Stages of Cybercrime Investigations

In [13], the authors combine the Cybercrime Execution Stack [12] and the investigations stages from the investigation process of law enforcement to a compound procedure of cybercrime investigation (See Figure 2) [13]. The purposes of this investigation procedure are to establish

the connection of Cybercrime Execution Stack and law enforcement investigation, and bridge the gap between technical and non-technical investigation. In the technology side, the authors refer the Cybercrime Execution Stack, and use this stack as the technology of investigate cybercrime. This investigation procedure has four phases: Initiation, Outcome, Cybercrime Execution Stack, and Law enforcement investigation process. The Cybercrime Execution Stack includes four stages: Data Objective, Exploitation Tactics, Attack Methods, and Networked Technology [13] (See Figure 1). The purpose of Cybercrime Execution Stack [12] is used to make the investigator analyze and divide the technology as well as the feature objectively, and assist every stage of the Law enforcement investigation process. The Law enforcement investigation includes six stages: Modelling, Assessment, Impact/Risk, Planning, Tools, and Action. Modelling stage used to assess, evaluate, plan and communicate the content of a crime event, and assist the assessment stage in the investigation process. The results of Modelling stage is used to analyze the knowledge and technology related to the cybercrime in the Assessment stage. In the Impact/Risk stage, the potential threat, offences, evidence, and victims will be analyzed in this stage. According to the results of Modelling stage, Assessment stage, and Impact and risk stage, the investigation actions will be planed and confirmed in the Planning stage. The Tools stage is used to find and consider the adequate skills, tools and equipment. The Tools stage is used to find the adequate skills, tools and equipment, and it will help the potential digital evidence. In the Action stage, the action plan will be confirmed, managed, and coordinated to include the skilled resources and jurisdictions.

2.3 New Model for Cyber Crime Investigation Procedure

In this paper of [26], the authors proposed a new procedure model of cybercrime investigation. It improves the digital investigation process of Brian Carrier [8], and increases several phases used to investigate the cybercrime, coursing this investigation procedure is more suitable to investigate the cybercrime event. In the digital investigation process of Brian Carrier [8] there are five phases, including readiness phase, deployment phase, physical crime scene investigation phase, cybercrime scene investigation phase, and review phase. In [26], the phases of investigation procedure include readiness phase, consulting with profiler, cybercrime classification and investigation priority decision, damaged cybercrime scene investigation, analysis by crime profiler, suspects tracking, injurer cybercrime scene investigation, suspect summon, cybercrime logical reconstruction, and writing report. The readiness phase is used to ensure the executing of investigation will be succeed, and reduce the waste time and error of investigation. The Crime profiling is used to find the information of the suspects from the crime scene. It will help to investigate same type crime in future, and

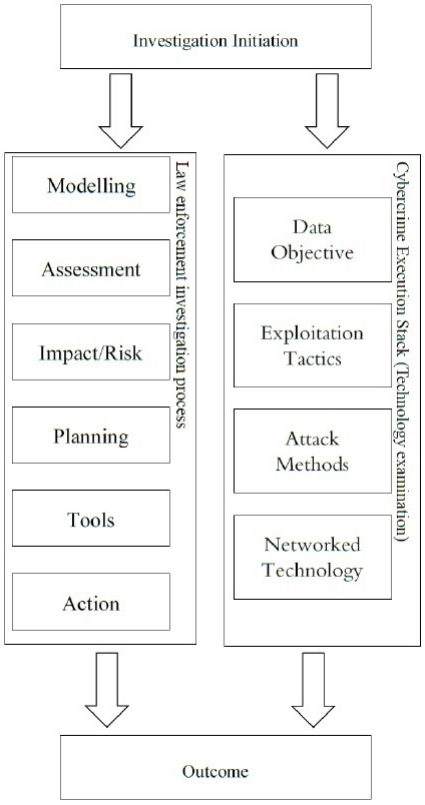


Figure 2: The stages of cybercrime investigations

reduce the time of investigation. The Cybercrime classification and investigation priority decision are used to decide the priority of investigation based on crime profiling data and classifying. In the Damaged (victim) cybercrime scene investigation phase, it's used to collect digital evidences, and the collection method is listed as below.

1) Establish "police line" on Internet;
2) Set the collection equipment to collect evidences of cybercrime events;
3) Photo evidences by digital or video camera;
4) Use tools to collect and analyze the volatile evidences [2, 19];
5) Use the storage imaging method to prevent the evidence from be modified or deleted [18, 19];
6) Obtain the evidences of network by using network forensic systems [24, 25].

In the Crime profiling phase, the investigator analyzes the nature of suspects by using the information collected from the crime scene. It will help to reduce the scope of investigation. After then, the investigator trace the suspects based on the digital evidences and cyber information in the Suspects tracking phase. In the Injurer cybercrime scene investigation phase, the investigation points are same with the Damaged (victim) cybercrime scene investigation phase, and increase a step to collect

the evidences from the printers of injurer. In the Suspects summon phase, the suspects will be summoned according to the collected digital evidences and the information of crime scene. In the cybercrime logical reconstruction phase, the investigators use the information and evidences that are collected from above investigation procedure to re-construct the cybercrime process, and use this reconstruct result to check the investigation result. At last phase, Writing report, the investigators write the report of criminal case about the evidence collect, preserve, and analyze. The Investigation Procedure of [26] is shown in Figure 3.

2.4 SoTE: Strategy of Triple-E on Solving Trojan Defense in Cyber-crime Cases

In this paper [16], it presented a strategy of Triple-E based on [16, 17], and used to investigate the cases of internet intrusion in the cybercrime, like Trojan. By using the strategy of Triple-E, the authors wish to identify the suspects of cybercrime, find the facts of cybercrime, and collect the evidences. In the Triple-E, it has three viewpoints, including Education, Enforcement, and Engineering. The Education viewpoint focuses to reduce the cybercrime amount of hackers and recidivism rate before cybercrime occurring. And the Education will establish a safe internet habits of people, which is used to increase public awareness by distributing a safe internet behavior, implementing a public awareness campaign, and observing the feeling of shame [16]. Furthermore, the investigators use the 6W1H (What, Which, When, Where, Who, Why, and How) Questions to find the motivation and purpose of hackers, and to establish a complete view of cybercrime events, avoiding being deceived by the suspects. The Enforcement focuses of investigation are the investigation field, philosophy role, the purpose of fact finding, and constructing the criminal fact. And the Enforcement based on MDFA (Multi-faceted Digital Forensics Analysis) Strategy can be used against the cybercrime events. Furthermore, the enforcement procedure can be examined from diverse viewpoints, such as exploring aggressive attacks, Comparing illegal offenses, and constructing a holistic view [16]. In the Engineering approach, it focuses on the forensics field, science role, the purpose of target authentication, and the method of arresting the criminals [16] based on the process of Ideal Log and M-N Model. In this viewpoint, it focuses on the importance of evidential records and comparison with other logs, and the measures such as to enable some elementary data for scientific consideration, synchronize the timestamp issues, and conduct an audit examination or cross examination [16]. The utilization of SoTE is shown in Figure 4.

This three viewpoints are related to four layers, including 6W1H questions policy, MDFA strategy procedure, Ideal Logs and M-N Model process, and Evidence record. The 6W1H questions policy is related to Education viewpoint, and used to define a direction of investigation procedure, including What, Which, When, Where, Who, Why, and How.

In the MDFA strategy procedure, it's related to Enforcement viewpoint, and used to analyze the information of cybercrime events. The MDFA strategy has four phases, including Evidential Phase (Evidence), Forensic Phase (Scene), Suffering Phase (Victim), and Behavior Phase (Suspect). In the Evidential Phase, it's used to collect and preserve evidences until the cybercrime case into court proceedings. The Evidential Phase has 5 steps: Identification, Preservation, Examination, Interpretation, and Presentation. In the Forensic Phase, it's used to collect and examine evidences from the crime scene, and discover the criminal process and facts through the crime scene reconstruction. The Forensic Phase has 5 steps: Qualified Expert, Chain of Custody, Admissibility Consideration, Forensic Conclusion, and Crime Scene Reconstruction. In the Suffering Phase, the investigators find and discover the clues of crime case by using the information from victims provided. The steps of Suffering Phase include Variety of Victim, Everyday Process, Victim Himself, Victim Reaction, and Societal Response. In the Behavior Phase, the information of the suspect will be evaluated and analyzed, such as the criminal psychology, personality, criminal actions, and voluntary or not. The steps of Behavior Phase are Background Understanding, Environmental Influence, Linkage Analysis, Logic Reasoning, and Criminal Profiling.

In the Ideal Logs and M-N Model process is used to identify the users behind the computer, and discriminate the in-formation of evidence is real or forged. The Ideal Logs fall into two categories, explicit and tacit knowledge. The explicit knowledge is used to find the location of the suspect by using the clues from digital evidences, such as IP address and timestamp. The tacit knowledge is used to find the clues of digital action and response message, such as data up-load/download, program execute, and abnormal behavior. The M-N Model process is a method used to check the log-in/logout process. M is the path traces from client to server, N is a parts including login and logout in a period of time. When a user wants to login server, the client will produce a login time record TLogin_1. The Login message will be through ISP (Internet Service Provider), and produce a login time record TLogin_2. The Login message will arrive to a server, and produce a login time record TLogin_3. When the user wants to logout a server, the logout message will follow the path of login, and produce the logout time record TLogout_3, TLogout_2 , TLogout_1 on the server, ISP, and client. Further, the M-N model provides a proposition analysis consisting of Sequential Inequality and Period Inequality. This methodology will help clarify the issues that the evidences are reliable or not, and the suspect is guilty or not. The M-N model is shown in Figure 5.

In the Evidence record, since the evidences is used to discover the crime fact and the internet behavior, the collected evidence record must has the clear and objective features. At last, the investigators find the causality from the result of this four-layer, and make the details of a

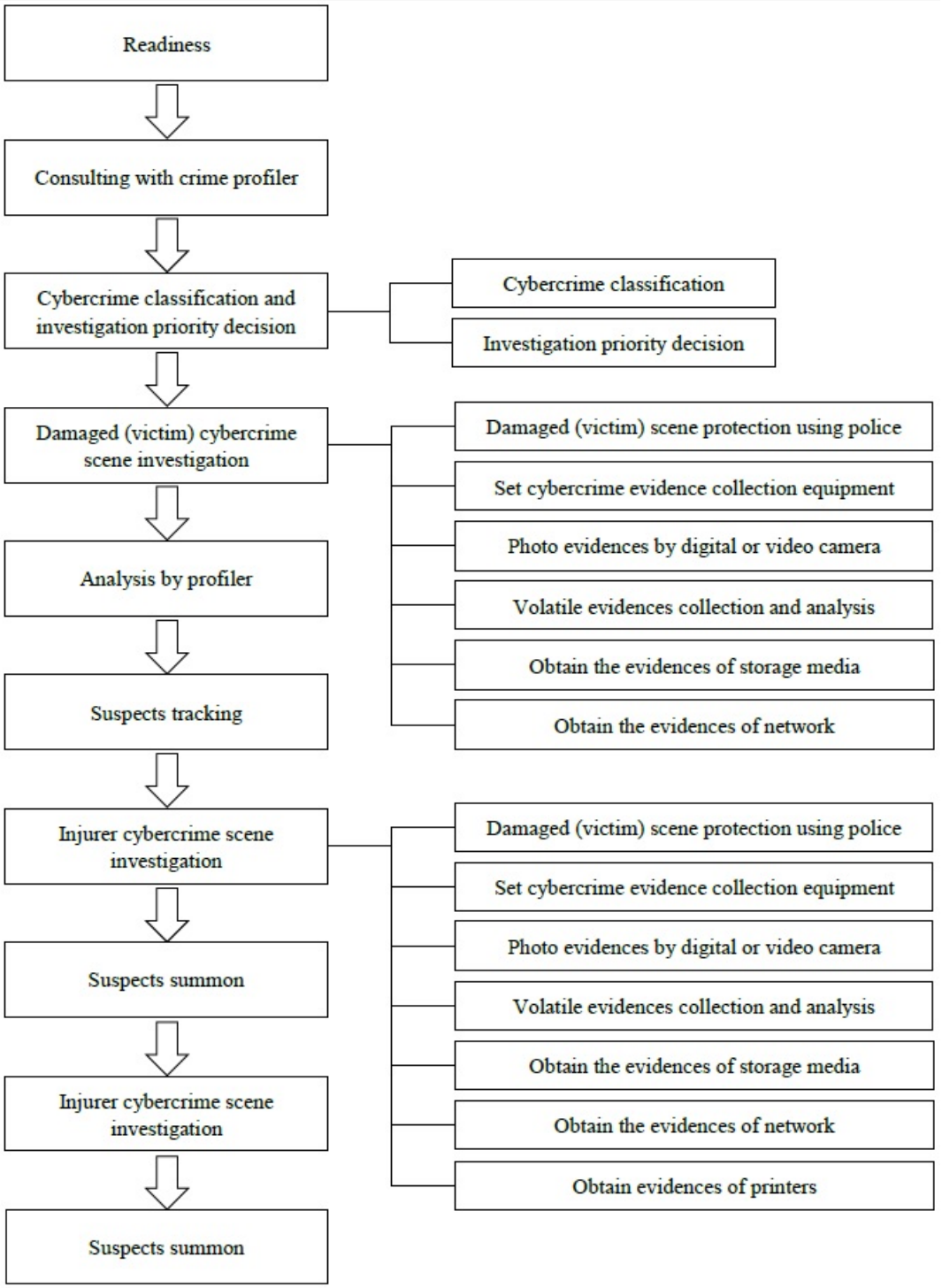


Figure 3: The investigation procedure of [12]

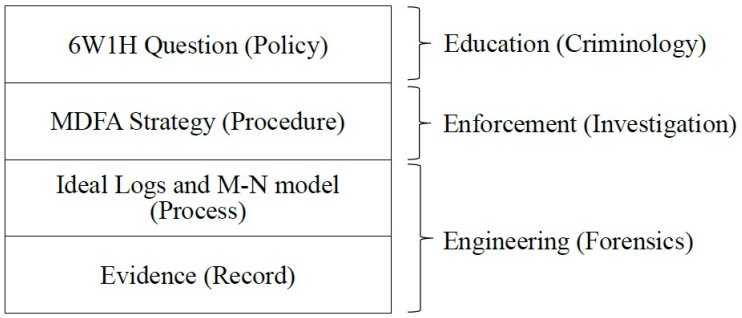


Figure 4: The utilization of SoTE

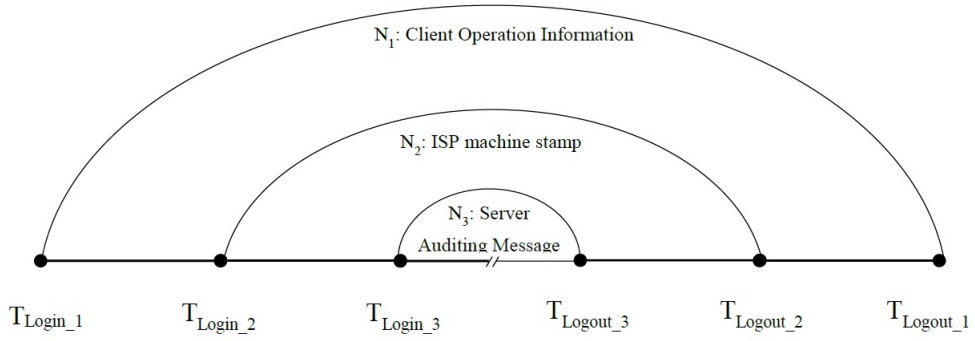


Figure 5: TM-N model

crime event clear.

2.5 A Study on Digital Forensics Standard Operation Procedure for Wireless Cybercrime

In this paper [35], the authors proposed a Standard Operation Procedure (SOP) of digital forensics for a wireless cybercrime. This procedure includes two pairs, the digital forensics and the wireless cybercrime investigation. The authors of this paper define the main behaviors of a wireless cybercrime, and use the definition to propose a wireless cybercrime investigation. Further, this paper proposed a digital forensics SOP based on the Digital Forensic Standard Operation Procedure (DFSOP).

In the wireless cybercrime investigation, the five behaviors of a wireless cybercrime were defined as follows [35]:

1) Cracking a wireless Internet access, and then connected to Internet by using the identity of another person;
2) Invading a wireless base station;
3) Intercepting packets of a wireless network; side-recording the conversations, accounts, and passwords;
4) Denial attacking the wireless base station;
5) Phishing in the wireless base station.

The wireless intrusion is the beginning in the wireless cybercrime. When the intrusion action is successful, the behaviors (2) to (5) will be also finished successfully. In order to solve the above wireless cybercrime, the investigation of this paper provide three stages, including Investigating and analyzing wireless cybercrime, Recognizing the criminal origin and behavior, and Arresting the perpetrator. In the Investigating and analyzing wireless cybercrime, the plan of investigation is to follow the description of the victim. And, then, the content of the wireless cybercrime will be identified by analyzing the modus operandi, such as checking the record from access points, the status of the network, the detection systems, and log files. In the Recognizing the criminal origin and behavior, the purpose of this stage is to find the suspects of a wireless cybercrime. The investigation methods are detecting the data of user, tracing the connection source, checking the record of communications, the firewall records, and so on. Sometime, in order to obtain the clues of a suspect, the investigation process even need to monitor and record the wireless network. In the Arresting the perpetrator, it's used to collect the evidence by using search and seize, summoning the suspects, and the forensic of wireless networks. Further, in order to facilitate the execution of investigation, this paper provides four directions to help the investigation of wireless cybercrime, including [35]:

1) Finding the illegal wireless access point;
2) Locking up the active illegal links;

3) Setting the honeypots in wireless;

4) Setting the intrusion detection system, such as a wireless intrusion prevention systems (WIPS) and wireless intrusion detection systems (WIDS).

In the digital forensics SOP, this paper proposes a wireless forensics SOP based on DFSOP. In the DFSOP, It has four phases: Concept, Preparation, Operation, and Report. In the Concept phase of DFSOP, it's used to describe the concepts of collecting evidence and forensics based on Laws, Principle, and Cognitive. The concepts have seven parts including collecting the evidences quickly and preserving them; ensuring the continuity of evidence; establishing a procedure to record the audit information and analysis of the digital evidences; operating the digital evidences by the experts; recording and filming the process of evidence collection, analysis and forensics; ensuring the integrity and security of data storage; using the copy instead of the original evidence in the operate analysis, investigation and forensics. On the other hand, the Concept phase of wireless DFSOP increases a procedure part to establish SOP and tools; in the laws part, it increases two subparts, acceptance at complaint only, and Non-acceptance at complaint only; in the cognitive, it in-creases three subparts; Forensic Expertise and Skills, Computer Professional and Skills, and Network Professional and Skills.

In the Preparation phase of DFSOP, it's used to collect related information to prepare the work before the forensics and the four parts based on Authenticity and Security Police, Collection of the Basic Information of the Target to Ensure the 5W&1H (Who, Why, When, Where, What and How), and Preparation of Tools and Information and Mission Education . The four parts are Collection of the basic information of the crime target, Preparation of tools, Professional members, and Education before the operation. On the other hand, the Preparation phase of wireless DFSOP increases a subpart, Simulation of Task Allocation and Action.

In the Operation Stage, it's divided to three procedures based on Crime Scene and Laboratory. The three procedures are Collection Procedure, Analysis Procedure, and Forensics Procedure based on Crime Scene and Laboratory. The procedures is used to collect evidence of every type by different tools, analyze these evidences, and then reconstruct the crime scene. Further, in the Operation Stage of wireless DFSOP, it presents three sources of collect evidences: Wireless Devices of Suspect, Wireless Devices of Scene, and Other Devices. And the Presentation forms the Collection phase, so the evidences are divided to the Volatile and Non-volatile type. The data collected from the wireless cybercrime will be analyzed including Picture, Images, Files, Connection History, Log Files of AP and PC, Wireless Network Event Viewer, and Wireless Packets.

In the Report Stage, it's used to produce a report about the content of cybercrime event, the evidences related to the cybercrime event, and the suspects of cybercrime event. This report will be sent to court, and become the basis of judgment. Therefore, the report must has the following related data: Copywriting and Presentation, Examination of Forensics Result, Court Preparation, and File Establishment and Learning. The Copywriting and Presentation are used to describe the content of this crime case, the collected evidences, the evidence sources, and the process of forensics. The Examination of Forensics Result is the procedures of evidence forensics and utilities usage. The Court Preparation means the dig-ital evidence forensics must be classified, and matched with the control procedure. At last, in the File Establishment and Learning, the forensics process, evidence types, and investigation experience of each cybercrime cases will be classified to establish in the file and sharing mode, it will help the future of cybercrime investigation.

3 The Discussion of Investigator Process and Investigation Procedure

3.1 Analysis and Comparison

In this paper, we collect five papers of the cybercrime investigation procedure, and analyze whether these proposed investigation procedure has the following features and content, the compatibility of traditional investigative procedures, cybercrime behavior analysis, evidence forensic procedures, case analysis and verification, the methods of evidence collection and analysis, and the area of judicial jurisdiction. In addition, we put the area of judicial jurisdiction into the comparison items, so it will help to understand the purpose and legal basis of the investigation procedure. The comparison of cybercrime investigation procedures are shown in Table 1.

1) With the compatibility of traditional investigative procedures: This is used to illustrate the investigation procedure of cybercrime, and whether it is proposed or not according to the conventional investigation procedure. It will affect whether this investigation procedure is easy to use or not by the police or investigators without the professional knowledge.

2) With the analysis of cybercrime behavior: In the investigation procedure of cybercrime, whether it has the analysis of cybercrime behavior clearly, and describes the focus types of this cybercrime procedure. It will help the investigators to find scope of this investigation procedure applies.

3) With the evidence forensic procedures: Whether an investigation procedure has the process and steps of forensic, it will affect the process of collecting the digital evidences. Without the forensic process, the investigators, perhaps, will not know what the digital evidences exist, and where can collect them.

4) With case analysis and verification: When the investigations procedure are used actually before, the investigation procedure of cybercrime only is a hypothesis. If the investigation procedure is based on an instance, or it can be used to analyze and verify for an instance, it will increase the feasibility of investigation and evidence collection procedures.

5) The methods of evidence collection and analysis: If the investigations procedure has a method of scientific or mathematical analysis, it will make the digital evidences of this procedure collected has more probative force.

6) The area of judicial jurisdiction: The investigations procedure we collected is not in the same judicial jurisdiction. To clarify these judicial jurisdictions will help the investigators to understand the purpose and the legal basis of investigation procedures.

In [12], it provides a Cybercrime Execution Stack. This framework stack presents the technology of cybercrime, the criminal object of attack, and attack mode. The main purpose of this framework stack is used to classify the cybercrime, and become a step in the cybercrime investigation procedure. Therefore, in [12], it only had the cybercrime analysis, but it did not establish a full investigation and evidence collection process. In [13], it provided a combination of investigative procedure with [12]. This procedure is based on an investigation procedure that already exists, and combine the frame-work of [12] proposed to become an investigative procedure focus on cybercrime. However, in [13], it presents a conceptual investigation procedures, but it did not provide the evidence forensic procedures and other methods. Therefore, in [13], it is an investigative procedures that have the compatibility of traditional investigative procedures and cybercrime behavior analysis. In [26], it provides a more clearly investigation procedure than [13]. In every investigation stage of [26], it describes the purpose of stage and source of forensic evidence clearly. However, in [26], it did not provide and describe the applicable type of cybercrime for the investigation procedure, and did not provide a clear evidence collection and analysis methods, as well as case analysis and verification. It makes the investigation procedure of [26] proposed still need to be proved that it can be used in the cybercrime events.

In [35], it provides a SOP investigation procedure of digital forensics used to investigate the wireless cybercrime. In this SOP, it provides a clear investigation phase based on the conventional investigative procedures. It makes the investigation procedures of [35] compatible with the conventional investigative procedures. In addition, the proposed investigation procedures of [35] defined the each step of investigation clearly, the behavior of wireless cybercrime, and a real pro-cess of investigating a cybercrime case. In this investigation procedure, it describes the process and source of evidence forensic process clearly. Therefore, the investigation procedure of [35] provides a high viability investigation procedures. In [16], it provides cybercrime investigation procedure based on criminology. This procedure is used to investigate Trojans cybercrime, and to illustrate the current situation of this type of crime. It makes the investigation procedure of [16] feasible. In addition, the investigation procedure of [16] uses the MDFA as the forensics process, and uses the M-N mod-el as a method of analysis the evidence in the forensics process. Since the investigation procedure of [16] conforms the above-mentioned characteristics of each, which makes it became a more complete cybercrime investigation procedures than others.

3.2 The Viewpoints of Cybercrime Investigation Procedure

In this paper The Digital evidence forensic process is one of stages belonging to the cybercrime investigation procedures. When a cybercrime occurs, the investigators will collect the digital evidences according to crime types, and preserve them. These Digital evidences are very important in the investigation procedure. The investigators confirm the crimes suspects, crime facts, time of occurrence, location, and possible criminal tools by analyzing these Digital evidences. The digital evidence forensic process is used to make cybercrime investigation procedures can be carried out smoothly. Since the every cybercrime case is independent, the digital evidence presented these cases will be in different ways. Therefore, the primary purpose of digital evidence forensic process should be "whether can collect direct evidences"; the second is "whether can collect indirect evidences"; and finally, "Which method of forensic evidence is the fastest." The reasons of this order is when the cybercrime is on the trial, and the judge will determine the outcome of the judgment based on the direct evidences; in the investigation procedure, the direct and indirect evidences will be the key to confirm the facts and suspects. There evidences will become a relevance indicator used to confirm the crime facts and the suspects, it is called the probative force of the evidence. If the process of forensic evidence is very fast, but cannot guarantee to collect evidence of high probative force, it will increase the time of investigation, as well as waste the judicial resources. Therefore, the digital evidence forensic methods should be focused on how to collect the direct and indirect evidences effectively.

In the conventional crime, the evidence type is substantive evidence, and the perpetrator can be found easily; there is an actual location of the crime, and the crime tools are easy to find. Therefore, the purpose of investigation procedures in the conventional crime is how to protect the crime scene, how to collect evidence from the crime scene, and how to quick to arrest the criminals. However, the cybercrime is a new type of criminal offense. The perpetrator of this crime is not easy to be found directly due to no actual location of the crime, so the evidences of crime are not easy to preserve and view, and criminal means and tools are not easy to find. There-

Table 1: The comparison between the each investigation forensics procedures of cybercrime

	Compatibility of traditional investigative procedures	Cybercrime behavior analysis	Evidence forensic procedures	Case analysis and verification	The methods of evidence collection and analysis	The area of judicial jurisdiction
The growing phenomenon of crime and the internet: A cybercrime execution and analysis model [6]	X	V	X	X	X	UK
The stages of cybercrime investigations: Bridging the gap between technology examination and law enforcement investigation [4]	V	V	X	X	X	UK
New Model for Cyber Crime Investigation Procedure[5]	V	X	V	X	X	Korea
A Study on Digital Forensics Standard Operation Procedure for Wireless Cybercrime [3]	V	V	V	V	X	Taiwan
SoTE: Strategy of Triple-E on solving Trojan defense in Cyber-crime cases [26]	V	V	V	V	V	Taiwan

fore, in the cybercrime investigation procedures, how to collect the key digital evidences becomes the important key. According to these Digital evidences, the investigators can confirm the criminal facts, the perpetrator, criminal tools and criminal means. Once the digital evidences are forged, altered, deleted or destroyed, it will cause the investigation hard to continue implementing, or even mislead the investigators. Finally, the results will make the innocence person is punished, and the guilty person is released. Therefore, in the investigation procedure of cybercrime, how to find the perpetrator accurately will be the primary purpose in the procedure; secondly, since the judicial resources are limited, how to reduce the use of judicial resources is one of the key points in the investigation procedure. In addition, all the investigation behavior must base on the relevant laws and regulations. Only the evidence forensic by the legal process can be used in the trial, and it is called the evidence capability. The evidence from unlawful conduct investigations obtained at trial would lose the evidence capability, and cannot be used to prove the defendant is guilt. The collected evidence must have the evidence capability, and then it will have the probative force. Therefore, how to find and verify the perpetrator accurately and lawfully and reduce the use of judicial resources will be the focus in the cybercrime investigation procedures.

4 Future Works

In the future, the types, methods, and targets of cybercrime will be changed continuously, and every types of computer, network equipment, and smart phone will be the target of attack. The points are how to combine the digital forensic methods and the resent investigation procedure, or even establish a defense method in the investigation procedure, resulting the purposes to defense, detect, and investigate effectively. Since the cybercrime will constantly change in the future, the cybercrime investigation procedure should be established based on the type of crime. In addition to these investigative procedures used to investigate the crime fact after the event occurring, it must has the functions of real time detection and forensic. Therefore, before the investigation procedure establishing, we propose to establish an architecture figure of cybercrime factors first. Once the cybercrime occurs, the investigators will decide which investigation procedure will be used based on the factors of case, and determine whether the subsequent criminal behavior has. However, many factors can affect cybercrime, so in the following we will enumerate several factors that will affect the cybercrime, including Criminal objects, Crime Environment, Connection Technology, Source areas of crime, Crime types, and Criminal objects. The affection factors of cybercrime as shown in Figure 6. In the Criminal objects, we divide the targets of crime into three types: equipment, single victim, and multiple victims. In this category, we wish to confirm the purposes of offenders for this type of victims.

In the Crime environment, we divide the environment into the Public network, Private network, and Half-Public net-work based on the classification of the network type. The purpose of this classification is used to find the place of exist-ing crime clues through the criminal environment. In the Connection technology, we enumerate three common technologies of network connection: Ethernet, Wireless Fidelity (WiFi), and Mobile communication technologies (MCTs). This classification will help the investigators to collect the digital evidences. In the Source areas of crime, we will confirm the jurisdiction area of crime, External or Internal, through the area that found the sus-

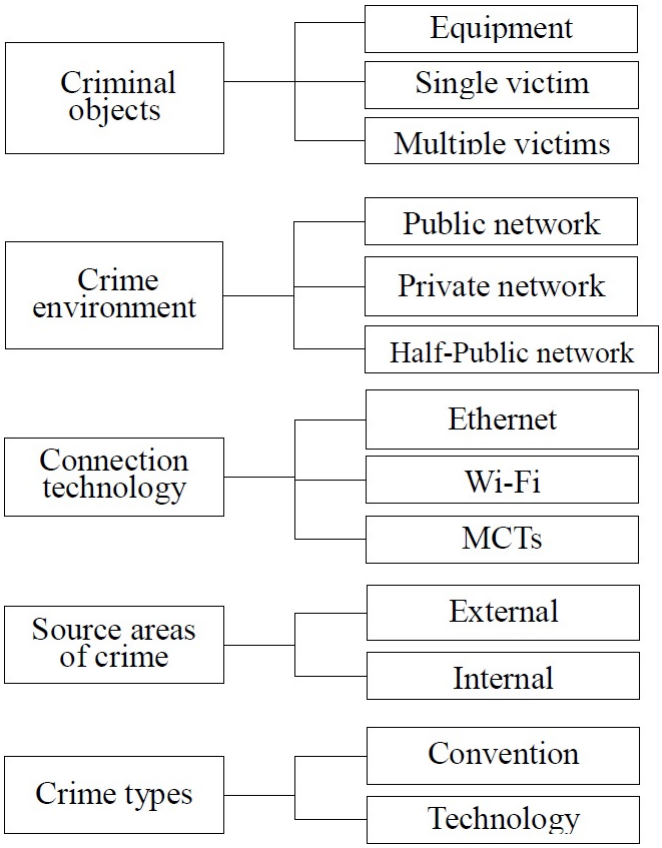


Figure 6: The affection factors of cybercrime

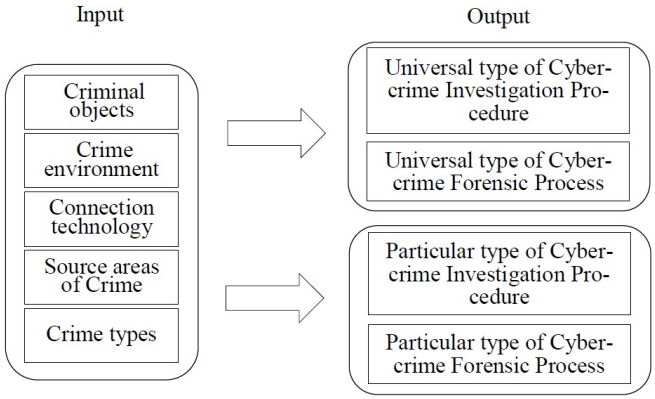


Figure 7: The investigation and forensic of cybercrime

pect. Finally, we will divide the crime types into Convention and Technology. This classification of crime types will be used to confirm the perpetrator of the crime and establish the tactics of investigation as the cumulative experience of investigation. In these factors, the order does not be constructed, but rather as the analysis items of cybercrime, and used to develop the evidence forensic process and investigation procedure.

According to the combination of these factors, it can be summarized to the concept of two types: Universal type of Cybercrime Investigation Procedure (UCIP) and Particular type of Cybercrime Investigation Procedure (PCIP); and two types of cybercrime forensic process: Universal type of Cybercrime Forensic Process (UCFP) and Particular type of Cybercrime Forensic Process (PCFP), as shown in Figure 7.

The universal type is used to describe the type of conventional crime. This crime type refers the criminal offenses al-ready existed before the Internet development, such as Fraudulence, intimidation, defamation, and so on. These scene of conventional crimes are gradually transferred to the Internet with the development of Internet. In order to investigate the conventional crimes and collect the digital evidences on the Internet, we propose to establish the UCIP and UCFP. The UCIP and UCFP aims to provide a simple and accurate method of investigation, and make the general security police also to investigate the cybercrime. And avoiding the criminal investigations is hindered because of the investigators lacking the knowledge of network technology. The Particular type is used to describe the crime type of technology-based. This crime type refer that the perpetrator uses the expertise and tools to commit the cybercrime offenses, and make the investigators without the expertise not to understand the method of crime, such as the Network attack, System intrude, Identity camouflage and hide, Data theft, and so on. Since investigating these crimes requires technical expertise, it will make the investigation process very difficult, and the general security police also cannot investigate this kind of cybercrime. Therefore, we propose to establish the PCIP and PCFP for the particular type of cybercrime. The purpose of PCIP and PCFP is to allow the general public security police and the investigators with technical expertise to cooperate together in the investigation of the cybercrime, and improve the efficiency of the investigation.

Since Internet still has the unknown development in the future, the affect factors of cybercrime and sub-factors will not be confined to the range of Figure 6; the investigation procedure and forensic process will not only include the two types in Figure 7. Once the new type of cybercrime event occurs, it still need the investigators to analyze the technology and features of cybercrime, and establish the emphasis investigation and forensic procedure.

Furthermore, after the investigation procedure, the criminal case will turn into the judgment procedure in the court. In the judgment of cybercrime, the result of trial will be different between cybercrime and conventional crime. The judgment procedure will affect the evidence that need to collect in the investigation procedure, and the evidences will affect the judge to find crime facts and the result of trial. Therefore, the investigation procedure and the forensic method of cybercrime still need to adjust and modify according to the result of trial.

5 Conclusions

In this research, we focus on how to collect the digital evidences from the cybercrime events, and how to propose an effective cybercrime investigation procedure. The digital evidences will help find the real perpetrators during the investigation procedure of cybercrime, and brings the perpetrators to justice in the trial; the effective cybercrime investigation procedures will help reduce the waste of judicial resources, and protect the human rights. A good method to collect digital evidences, in addition to focus on how to collect quickly the evidence, should focus on how to collect the digital evidence of high probative force. Whether these digital evidences are collected automatically by the computer system, or collected manually by the system administrator, the value of evidences are based on how many probative force that can provide to prove in the trial. In cybercrime investigation procedure, a good investigation procedure requires the less use of judicial resources, and avoids the mandatory punishment of suspects.

Acknowledgments

The author expresses deep sense of gratitude to the Department of Science & Technology (DST) , Govt. of India, for financial assistance through INSPIRE Fellowship leading for a PhD work under which this work has been carried out, at the department of Computer Science & Engineering, University of Kalyani.

References

[1] I. O. Ademu, C. O. Imafidon, D. S. Preston, "A new approach of digital forensic model for digital forensic investigation," *International Journal of Advanced Computer Science and Applications*, vol. 2, no. 12, pp. 175–178, 2011.

[2] D. Brezinski, T. Killalea, "Guidelines for evidence collection and archiving," RFC 3227, 2002.

[3] R. P. Bryant, *Investigating Digital Crime*, Wiley, 2008.

[4] E. Casey, *Digital Evidence and Computer Crime: Forensic Science, Computers and the Internet*, Academic Press, pp. 41-46, 2000.

[5] E. Casey, *Handbook of Digital Forensics and Investigation*, Academic Press, 2009.

[6] E. Casey, *Digital Evidence and Computer Crime*, Academic Press, 2004.

[7] B. Carrier, "Defining digital forensic examination and analysis tools using abstraction layers," *International Journal of Digital Evidence*, vol. 1, no. 4, pp. 1–12, 2003.

[8] B. Carrier, E. H. Spafford, "Getting physical with the digital investigation process," *International Journal of Digital Evidence*, vol. 2, no. 2, pp. 1–20, 2003.

[9] Y. Chen, S. Das, P. Dhar, A. E.Saddik, A. Nayak, "Detecting and preventing IP-spoofed distributed DoS attacks," *International Journal of Network Security*, vol. 7, no. 1, pp. 69–80, 2008.

[10] Alan M. Gahtan, *Electronic Evidence*, Thomson Canada Limited, 1999.

[11] M. Geva, A. Herzberg, Y. Gev, "Bandwidth distributed denial of service: Attacks and defenses," *IEEE Security & Privacy*, vol 1, pp. 54–61, 2014.

[12] P. Hunton, "The growing phenomenon of crime and the Internet: a cybercrime execution and analysis model," *Computer Law & Security Review*, vol. 6, no. 6, pp. 528–535, 2009.

[13] P. Hunton, "The stages of cybercrime investigations: Bridging the gap between technology examination and law enforcement investigation," *Computer Law & Security Review*, vol. 27, no. 1, pp. 61–67, 2011.

[14] N. Jeyanthi1, N. Ch. Sriman Narayana Iyengar, "An entropy based approach to detect and distinguish DDoS attacks from flash crowds in VoIP networks," *International Journal of Network Security*, vol. 14, no. 5, pp. 257–269, 2012.

[15] D. Y. Kao, and S. J. Wang, "The IP address and time in cyber-crime investigation," *Policing: An International Journal of Police Strategies & Management*, vol. 32 no. 2, pp. 194–208, 2009.

[16] D. Y. Kao, S. J. Wang, Frank F. Y. Huang, "SoTE: Strategy of Triple-E on solving Trojan defense in Cyber-crime cases," *Computer Law & Security Review*, vol. 26, no. 1, pp. 52–60, 2010.

[17] G. C. Kessler, "Anti-forensics and the digital investigator," in *Proceedings of the 5th Australian Digital Forensics Conference*, 2007.

[18] G. A. Lee, D. W. Park, and Y. T. Shin, "A study on the chain of custody for securing the faultlessness of forensic data," *Journal of the Korea Society of Computer and Information*, vol. 11, no. 6, pp. 175–184, 2006.

[19] S. H. Lee, H. Kim, S. Lee, J. Lim, "Digital evidence collection process in integrity and memory information gath-ering," in *Systematic Approaches to Digital Forensic Engineering, First International Workshop on Systematic Ap-proaches to Digital Forensic Engineering (SADFE'05)*, pp. 236–247, 2005.

[20] C.Y. Liu, C.H. Peng, and I.C. Lin, "A survey of botnet architecture and batnet detection techniques," *International Journal of Network Security*, vol. 16, no. 2, pp. 81–89, 2014.

[21] M. Mahmoud, M. Nir, and A. Matrawy, "Survey on botnet architectures, detection and defences," *International Journal of Network Security.* (in press)

[22] B. Mihajlov and M. Bogdanoski, "Analysis of the WSN MAC protocols under Jamming DoS attack," *International Journal of Network Security*, vol. 16, no. 4, pp. 304–312, July 2014.

[23] E. Moulton, *The Future of Cybercrime*, Police Professional, 2008.

[24] S. Mukkamala, A. H. Sung, "Identifying significant feature for network forensic analysis using artificial intelligent techniques," *International Journal of Digital Evidence*, vol. no. 4, pp. 1–17, 2003.
[25] J. S. Park, U. H. Choi, J. Moon, T. Shon, "A study on network forensics information in automated computer emergency response system," *Journal of the Korea Institute of Information Security and Cryptology*, vol. 14. no. 4, pp. 149–162, 2004.
[26] Y. D. Shin, "New model for cyber crime investigation procedure," *Journal of Next Generation Information Technology*, vol. 2, no. 2, pp. 1–7, 2011.
[27] D. L. Shinder, M. Cross, *Scene of the Cybercrime*, Second Edition, Syngress, 2008.
[28] C. Sorrells and L. Qian, "Quickest detection of denial-of-service attacks in cognitive wireless networks," *Inter-national Journal of Network Security*, vol. 16, no. 6, pp. 468–476, 2014.
[29] M. Subramanian, T. Angamuthu, "An autonomous framework for early detection of spoofed flooding attacks," *International Journal of Network Security*, vol. 10, no. 1, pp. 39–50, 2010.
[30] J. Udhayan, T. Hamsapriya, "Statistical segregation method to minimize the false detections during DDoS attacks," *International Journal of Network Security*, vol. 13, no. 3, pp. 152–160, 2011.
[31] D. S. Wall, *Cybercrime: The Transformation of Crime in the Information Age*, Polity Press, 2007.
[32] S. J. Wang, "Measures of retaining digital evidence to prosecute computer-based cyber-crimes," *Computer Standards & Interfaces*, vol. 29, pp. 216–223, 2007.
[33] S. J. Wang, "Measures of retaining digital evidence to prosecute computer-based cyber-crimes," *Computer Standards & Interfaces*, vol. 29, no. 2, pp. 216-223, 2007.
[34] M. Yar, *Cybercrime and Society*, Sage Publishing Ltd, 2006.
[35] Y. S. Yen, I. L. Lin, A. Chang, "A study on digital forensics standard operation procedure for wireless cyber-crime," *International Journal of Computer Engineering Science*, vol. 2, no. 3, pp. 26–39, 2012.

Jia-Rong Sun Jia-Rong Sun received the B.S. degree and M.S. degree in Computer Science and Information Engineering from Asia University, Taiwan in 2010. He is currently a Ph.D student in the Department of Computer Science and Information Engineering, Taichung, Taiwan. His research interests include Information security and cybercrime investigation.

Mao-Lin Shih received the B.S. degree in College of Law National Taiwan University, Taipei, Taiwan; and the honorary Ph.D from Woosuk University, Korea, in 2009; Dr. Shih was a judge in 1984-1993. He was also a Chief Prosecutor during 1997-2004. From 2005 to 2008, he was the Minister of Ministry of Justice in Taiwan. He is currently a professor of the Department of Financial and Economic Law in Asia University. He is the director-general of Legal Risk Management Society of Taiwan. His current research include Criminal Law and Legal Case Study.

Min-Shiang Hwang received the B.S. in Electronic Engineering from National Taipei Institute of Technology, Taipei, Taiwan, Republic of China, in 1980; the M.S. in Industrial Engineering from National Tsing Hua University, Taiwan, in 1988; and the Ph.D. in Computer and Information Science from National Chiao Tung University, Taiwan, in 1995. He also studied Applied Mathematics at National Cheng Kung University, Taiwan, from 1984-1986. Dr. Hwang passed the National Higher Examination in field "Electronic Engineer" in 1988. He also passed the National Telecommunication Special Examination in field "Information Engineering", qualified as advanced technician the first class in 1990. From 1988 to 1991, he was the leader of the Computer Center at Telecommunication Laboratories (TL), Ministry of Transportation and Communications, ROC. He was also a project leader for research in computer security at TL in July 1990. He obtained the 1997, 1998, and 1999 Distinguished Research Awards of the National Science Council of the Republic of China. He is a member of IEEE, ACM, and Chinese Information Security Association. His current research interests include database and data security, cryptography, image compression, and mobile communications.

An Extended Identity Based Authenticated Asymmetric Group Key Agreement Protocol

Reddi Siva Ranjani, D. Lalitha Bhaskari, P. S. Avadhani
(Corresponding author: Reddi Siva Ranjani)

Department of Computer Science and Systems Engineering, Andhra University
Visakhapatnam, Andhra pradesh, India.
(Email: rsivaranjani552008@gmail.com)
(Received Sept. 29, 2013; revised and accepted Feb. 18 & Mar. 13, 2014)

Abstract

Lei et al. [26] proposed a new asymmetric group key agreement protocol allowing a set of users to negotiate a common encryption key accessible by any user, and each user holds her respective decryption key. This enables the confidential message communication among group users, and grants any outsider to send message to the group. In this paper, an authenticated asymmetric group key agreement protocol is proposed, which offers security against active as well as passive attacks. Proposed protocol uses broadcast encryption mechanism without relying on the trusted dealer to distribute the secret key. An identity based feature is included in the protocol to provide authentication.

Keywords: Identity based, group key agreement, pairings, public key cryptography

1 Introduction

Group Key Agreement (GKA) Protocols [1, 14] allow a group of users to derive a common secret key, from which a session key can be inferred. Therefore, they are used in any group oriented communication applications, such as video conference, priced VCD distribution and collaborative computations. All these applications require secure broadcasting at the network layer among the parties in the group communication. In conventional group key agreement, all the users in the group establish a common shared secret key, which is used in message encryption and decryption. In the recently developed asymmetric key agreement protocol by Wu et al. [23], all the group participants negotiate a common encryption key which is accessible to all including non group members, unlike the regular GKA. Each group participant holds his own contribution, which is used in his secret decryption key derivation. Therefore, beside the group participants, Asymmetric Group Key Agreement Protocols (AGKAP) allows outsiders of the group to broadcast the cipher messages to the group participants, provided that the sender knows the negotiated public key.

1.1 Motivation and Contributions

Group key management protocols [19, 22] are classified into group key distribution protocols and group key agreement protocols. The group key distribution protocols [2] are used to distribute group key to the group participants. In group key agreement, group participants are actively involved in the derivation of group key. Compared with conventional group key agreement protocol, AGKAP is having the advantage of one round efficiency. Many of the popular conventional GKA protocols require two or more rounds for sharing the common secret key. In these protocols, all the participants should be connected concurrently in order to share the key. However, if the participants are located in different locations with different time zones, it is very difficult for them to be connected concurrently. But, single round ASGKA protocols [17, 23] have several advantages over the GKA protocols with two or more rounds. The single round ASGKA allows each participant to publish their public key contribution by holding their respective secret key. The participant need not be connected during the key sharing. To send a message to participants in the group, the sender encrypts the message commonly using the derived common group public key and generates the cipher text. The protocols developed are efficient but secure against passive attacks only. However, in real world attackers are active attackers, who can control the communication channel to place powerful attacks. Man-in-middle attack and also, with which the active attackers can delay, modify, replay and insert the messages during the execution of the protocol. Hence, it is imperative for an ASGKA protocol to resist against the attacks from active adversaries.

Any Authenticated key agreement protocol [9, 10, 15, 20, 27], which ensures that no entities other than intended participant can possibly compute the agreed group session key, even the attacker is active or passive. In au-

thenticated key agreement protocols, each user can obtain others certificate, extract other participant's public key, checks the validity of the certificate and then finally a common group key was computed. Consequently, the management of the certificate incurs overheads computation, storage and communication. To eliminate such overhead costs, Identity Based Public Key Cryptography (IB-PKC) that was introduced by Shamir [21]. The distinct feature of IBPKC is that the public key is derived using the participant identity such as telephone number and email-ID. The corresponding private key is derived only by the trusted third party, Private Key Generator (PKG) who owns the master secret of the system.

In this paper, a security model for identity based authenticated asymmetric group key agreement protocol is developed. Our protocol is based on the identity based batch multi signature with batch verification [8, 25] to generate identity based signature. Furthermore, participant identity is used in the derivation of broadcast message computations. The proposed protocol is like an authenticated group key agreement protocol with following features:

- Permits the group having any number of members without compromising the security.
- Facilitates the mutual authentication between the Group Controller and members in the group.
- Performance is compared with existing protocols.
- Allows users to broadcast public information by concealing private information. A Common group key is inferred from public information, which is received from other group members.

1.2 Related Works

Firstly, Diffie and Hellman [12] proposed a solution to key agreement; later Joux [17] extended the key agreement to three parties. Many attempts have been performed to extend the Diffie-Hellman and Joux protocols to n participants. Burmester-Desmedt [7] protocol succeeded in extending the key agreement protocol with two rounds and irrespective on participants' count. For key agreement protocols in open networks, communication should be secure against active adversaries. But, Diffie- Hellman, Joux and Burmester-Desmedt protocols do not authenticate the communicating entities.

To add authentication, several protocols have been proposed among them, the GKA protocol [16, 18] is based on IBPKC, which refers to Katz and Yung's result [11] for an authenticated version. Bresson et al. [5] formalized the first security model for group key agreement protocol, extending the group key agreement between two or three parties. Subsequently, the model was refined and modified by Bresson et al. [4, 6]. Later Lei et al. [24] extended these models to define the security of IB-AAGKA protocol, later it was extended to broadcast encryption application for open networks [26]. In this paper, we extended these models to define identity based asymmetric asynchronous group key agreement protocol.

1.3 Paper Outline

Section 2, reviews of Bilinear maps and some complexity assumptions were discussed. Section 3 defines the proposed protocol, security issues of the proposed protocol are discussed in Section 4, Section 5 discusses whole about the performance evaluation and finally we concluded the work in Section 6.

2 Preliminaries

In this section, we put forward the notations, definitions that we used in the discussion of the forth coming sections.

Bilinear Maps. We review the basic notations of the bilinear maps [3, 25] under our proposal. Let $(G_1, +)$ and $(G_T, *)$ be two groups of prime order $q > 2k$ for a security parameter $k \in N$. A function $e: G_1 \times G_1 \longrightarrow G_T$ is said to be a bilinear map if it satisfies the following properties:

1) Bilinearity:

$$\begin{aligned} e(aP, bQ) &= e(P,Q)^{ab}, \forall P, Q \in G_1; a, b \in Z. \\ e(P+Q, R) &= e(P,R) * e(Q,R) \\ e(P, Q+R) &= e(P,Q) * e(P,R), \forall P, Q, R \in G_1. \end{aligned}$$

2) Non-degeneracy: $e(P,Q) = 1; iff \quad P = 1.$

3) Computability: There exists a polynomial time algorithm to compute $e(P,Q), \forall P, Q \in G_1$.

A bilinear map is defined as a probabilistic polynomial time algorithm (E) that takes a security parameter k and returns a uniformly random tuple (G_1, G_T, e, g, q) of bilinear parameters, where g is the generator of G_1 and e is the bilinear map.

Consequences of Pairings. Pairings have important consequences on the hardness of certain variants of the Diffie-Hellman problem. For instance, symmetric pairings lead to a strict separation between the intractability of the Computational Diffie-Hellman problem and the hardness of the corresponding decision problem. The security of our proposal is based on the hardness of the computational Diffie-Hellman (CDH) problem, Divisible computational Diffie-Hellman and K-Bilinear Diffie-Hellman exponent, which are described below:

- Computational Diffie-Hellman (CDH): Given g, g^α, g^β for unknown $\alpha, \beta \in Z_q$, compute $g^{\alpha\beta}$.
- CDH Assumption: The assumption states that Adv[E] cannot be negligible for any polynomial time algorithm E, where $Adv[E] = Pr[E(g, g^\alpha, g^\beta) = g^{\alpha\beta}]$ and Pr describes the probability.

- Divisible Computational Diffie-Hellman (DCDH) Problem: Given g^{α}, g^{β} for unknown $\alpha, \beta \in Z_q$, compute $g^{\alpha/\beta}$.
- DCDH Assumption: There is no polynomial time algorithm that can solve the DCDH problem with the non negligible property.
- k-Bilinear Diffie-Hellman Exponent ($k - BDHE$) Problem: Given g, h, and $y_i = g^{\alpha^i}$ in G_1 for $i = 1, 2, \cdots, k, k+2, \cdots, 2k$ as the input and compute $e(g,h)^{\alpha^{k+1}}$. Since the input vector lacks $g^{\alpha^{k+1}}$ term, the bilinear map does not seem to help to compute $e(g,h)^{\alpha^{k+1}}$.
- k-BDHE Assumption: Let E be an algorithm which has an advantage in solving k-BDHE problem. There is no polynomial-time algorithm that can solve the k-BDHE problem with non-negligible probability.

$$Adv[E] = Pr[E(g, h, y_1, y_2, \cdots, y_k, y_{k+2}, \cdots, y_{2k}, e(g,h)^{\alpha^{k+1}})].$$

3 Our Proposal

In this section, we proposed the identity based asymmetric group key agreement protocol based on [23, 26]. We considered a group of n participants who are intended to receive secure messages from the participants, who may or may not be the group participants. Our scheme adopts bilinear pairings; it can be organized in terms of following stages. The algorithm works as follows:

Setup. The group Controller (U_0) generates the system parameters in this stage. The U_0 generates a uniformly tuple $P = (G_1, G_T, e, H, g, q)$ of bilinear instance. U_0 chooses a cryptographic hash function $H : \{0,1\}^* \longrightarrow G_1$, where G_1 be the group with prime order q, $e : G_1 X G_1 \longrightarrow G_T$ is a bilinear map and g is the generator of G_1. Also generate and propagate securely the private (s_i) and public keys ($Ppub_i$) to each user.

Key Establishment. At this stage, the participants in the group communication generate and publish the messages which will be used in the generation of group encryption and decryption keys. Let $U_1, U_2, U_3, \cdots, U_n$ be the participants involved in the group communication. Each participant U_i with identity ID_i for $1 \leq i \leq n$ in group communication will perform the following steps.

1) Randomly choose $h_i \in G_1, r_i \in Z_q^*$ and compute $x_i = g^{r_i}$, $A_i = e(H(ID_i) + h_i, g)$.
2) For $1 \leq j \leq n$, compute $\sigma_{i,j} = h_i * H(ID_j)^{r_i}$.
3) Generate a signature ρ_i on x_i using s_i. In order to keep the protocol efficient, one may choose the an identity based signature scheme, which provides the batch verification to generate ρ_i.
4) Publish $\{\sigma_{i,1}, \cdots, \sigma_{i,i-1}, \sigma_{i,i+1}, \cdots, \sigma_{i,n}, (x_i, A_i, ID_i, \rho_i)\}$.

After completion of this stage, each participant can get the messages as shown in the table 1, where $\sigma_{i,i} = h_i * H(ID_i)^{r_i}$ is not be published to any other user in the group communication, but it is kept secret by U_i.

Encryption Key Derivation. Any user in the group can compute the group encryption key (W, A, Q), where

$$W = \Pi_{i=1}^{n} x_i$$
$$A = \Pi_{i=1}^{n} A_i$$
$$Q = \Pi_{i=1}^{n} H(ID_i).$$

The group encryption key (W, A, Q) is accepted if all the n message signatures pairs $(x_1, \rho_1), (x_2, \rho_2), \cdots, (x_n, \rho_n)$ are valid.

Decryption Key Derivation. The user U_i computes the individual decryption key $d_i = \Pi_{l=1}^{n} \sigma_{l,i}$ and accepts the d_i if all the n message signatures pairs $(x_1, \rho_1), (x_2, \rho_2), \cdots, (x_n, \rho_n)$ are valid.

Encrypt. After knowing the public parameters generated by U_0, and the group encryption key (W, A, Q), any user U_i in the group communication can encrypt any message m by executing following steps.

1) Select a random number $t \in Z_q$.
2) Compute the variables $C_1 = g^t$, $C_2 = W^t$ and $C_3 = m * A^t$.
3) Communicate the cipher text $C = (C_1, C_2, C_3)$ to the receiver.

Decrypt. To deduce the plaintext from the cipher text, each participant U_i can decrypt

$$m = \frac{C_3}{e(d_i, C_1) * e(Q, C_1) * e(H(ID_j)^{-1}, C_2)} \quad (1)$$

4 Security Analysis

Our proposed protocol is equipped with all the following security attributes.

1) Known Key Security: For each session, the participant randomly selects h_i and r_i, results separate independent group encryption and decryption keys for other sessions. Therefore, a leakage of group decryption keys in one session will not help in the derivation of other session group decryption keys.
2) Unknown Key Share: In our protocol, each participant U_i should generate a signature ρ_i using x_i. Therefore, only group participants can verify whether the coming ρ_i is from authorized person or not. Hence, no non group participant can be impersonated.

Table 1: Message obtained by the participants

User	U_1	U_2	U_3	...	U_n	All
U_1	–	$\sigma_{1,2}$	$\sigma_{1,3}$	...	$\sigma_{1,n}$	(x_1, A_1, ID_1, ρ_1)
U_2	$\sigma_{2,1}$	–	$\sigma_{2,3}$	...	$\sigma_{2,n}$	(x_2, A_2, ID_2, ρ_2)
U_3	$\sigma_{3,1}$	$\sigma_{3,2}$	–	...	$\sigma_{3,n}$	(x_3, A_3, ID_3, ρ_3)
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
U_n	$\sigma_{n,1}$	$\sigma_{n,2}$	$\sigma_{n,3}$	...	–	(x_n, A_n, ID_n, ρ_n)

Table 2: Comparison with various protocols

	[13]	[18]	[26]	Our Protocol
Exponentiation	3	3	0	0
Multiplications	2n-2	$n^2/2 + 3n/2 - 3$	Sender: 2n Reciever: 2n	Sender: 2n Reciever: 2n
Verification or Comparisons	n+1	2n-2	n: sender n-1: participants	n-1:participants
No. of Rounds	2	3	1	1

3) Key Compromise Impersonate: Due to generation of unforgeable signature by the user U_i, the adversary cannot generate the valid signature on behalf of U_i. Even if the participant U_j's private key is compromised by the adversary, he cannot impersonate other participant U_i with U_j's private key. Hence, key impersonation property is not possible in the proposed protocol.

5 Performance Evaluation

In this section we are summarizing the performance of the proposed protocol and two other authenticated asymmetric group key agreement protocols under the same cryptosystem setting. Table 1 shows message obtained by the participants.

Round Efficiency. To constitute a session key, the existing Group Key Agreement (GKA) protocols [13, 18] requires two or more rounds, Therefore, all the participants in these protocols should connect concurrently. As in [26], our protocol, each participant needs to transmit message only once. Although both require only one round, we achieved a stronger security against active attacks. In the one round feature each participant can simply send the intermediate value and leave.

Computational Overhead. As to the computational overhead our protocol is same as the protocol [26] at the key Agreement, Encryption and Decryption, but ours not require any certificate. At the Encryption and decryption sides the computational overheads are same, because only three cipher text variables C_1, C_2 and C_3 are being sent at the encryption side.

Communication Overhead. We observed that the communication overhead is slightly lower than that in [26] at the group key agreement stage, since no certificates are required. In [26], signature is computed over $n + 4$ parameters, but, in proposed one signature is computed on one parameter x_i.

Storage Overhead. Our protocol requires less storage. In [26] protocols each user requires a storage area of $n + 4$ for system parameters, a group encryption, decryption keys and private key. However, our protocol requires only ten storage locations to store the variable P, group encryption and private decryption key.

Simulation. A desktop having Intel(R)Core(TM) i5-2400 CPU at 3·10GHz, frequency 3·09GHz and 2·91GB of RAM is used in evaluating the performance of the proposed protocol. A pairing based cryptography library functions are used in the protocol development, the experimentation is held by varying the number of participants from 2 to 100 by considering the length of group elements in G_1 and G_T is assigned to 171 and 1024 bits respectively. Involvement of exponentiation and multiplication in [13, 18] protocols results more computation time compared to other two protocol. Hence, time cost comparison is done between [26] and proposed protocols (See Table 2.

Figure 1 shows the relationship between the number of group participants and the time required to generate the group encryption key for sender and a participant. From the figure, the time needed by the sender and a participant to generate a group encryption key are almost same and raise linearly as

the number of participants increases. However, the cost of time is not high when the number of participants is 100. The [26] protocol consumes 500ms and 495.34ms for the sender and for a participant respectively, where as the proposed protocol needs 502ms and 496.97ms for the sender and for a participant respectively. Figure 2 shows the relationship between the number of group participants and the running time required to generate a group decryption key in both [26] and proposed protocols. From the figure, one can observe that the time cost to generate a group decryption key is increasing linearly with the number of participants. The time cost is ranges from 0.44ms to 132.244ms in [26], where as proposed protocol cost ranging from 0.46ms to 136.523ms. Figure 3 & Figure 4, shows the time cost to generate the ciphertext and decrypt a ciphertext in both the protocols, from figure one can see that time cost is constant irrespective of the number of group participants. The [26] protocol consumes 6.6ms and 6.8ms respectively towards encryption and decryption process. But, the proposed one needs 6.6ms and 6.9ms. Finally, the proposed protocol is consuming almost same time cost as in [26] protocol.

6 Conclusion

We have defined a one round identity based authenticated asymmetric group key agreement protocol from Bilinear maps. The protocol allows the participants in the group to derive a common encryption key, offers the key security and unknown key share properties. Evaluation shows that, the overheads of the proposed protocol are less when compared to others [13, 18, 26]. Computation cost for group common encryption and decryption key generation, ciphertext generation and plaintext extraction is almost same as [26]. Based on our authenticated asymmetric group key agreement protocol,a broadcast based encryption system was proposed. Also, batch multi-signature can be separated into individual signature.

Acknowledgments

The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] A. Abdel-Hafez, A. Miri, and L. O. Barbosa, "Authenticated group key agreement protocols for ad hoc wireless networks," *International Journal of Network Security*, vol. 4, no. 1, pp. 90–98, 2007.

[2] M. J. Bohio and A. Miri, "Self-healing group key distribution," *International Journal of Network Security*, vol. 1, no. 2, pp. 110–117, 2005.

[3] D. Boneh, X. Boyen, and E. Goh, "Hierarchical identity based encryption with constant size ciphertext.," in *EUROCRYPT'05*, LNCS 3494, pp. 440–456, 2005.

[4] E. Bresson, O. Chevassut, and D. Pointcheval, "Dynamic group diffie hellman key exchange under standard assuptoions," in *Proceedings of EUROCRYPT'02*, LNCS 2332, pp. 321–336, 2002.

[5] E. Bresson, O. Chevassut, and J. Quisquater, D. Pointcheval, "Provably authenticated group Diffie-Hellman key exchange," in *Proceedings of ACM CCS'01*, pp. 255–264, 2001.

[6] E. Bresson, O. Chevassut, and J. Quisquater, D. Pointcheval, "Provably authenticated group Diffie-Hellman key exchange," in *Proceedings of AASIACRYPT'01*, LNCS 2248, pp. 290–309, 2001.

[7] M. Burmester and Y. Desmedt, "A secure and efficient conference key distribution system," in *Proceedings of EUROCRYPT'94*, LNCS 950, pp. 275–286, 1995.

[8] J. Camenish, S. Hohenberger, and M. Pedersen, "Batch verification of short signatures," in *Proceedings of EUROCRYPT'07*, LNCS 4515, pp. 246–263, 2007.

[9] X. Cao, W. Kou, and X. Du, "A pairing free identity-based authenticated key agreement protocol with minimal message exchanges," *Information Science*, vol. 180, no. 15, pp. 2895–2903, 2010.

[10] T. Y. Chang, M. S. Hawng, and W. P. Yang, "A communication efficient three party password authenticated key exchange protocol," *Information Science*, vol. 181, no. 1, pp. 217–226, 2011.

[11] K. Y. Choi, J. Y. Hwang, and D. H. Lee, "Efficient ID-based group key agreement with bilinear maps," in *Proceeding of Public Key Cryptography (PKC'04)*, LNCS 2947, pp. 130–144, 2004.

[12] W. Diffie and M. Hellman, "New directions in cryptography," *IEEE Transactions on Information Theory*, vol. 22, no. 6, pp. 644–654, 1976.

[13] R. Dutta and R. Barua, "Constant round dynamic group key agreement," in *Proceedings of ISC'05*, LNCS 3650, pp. 74–88, 2005.

[14] R. Dutta and R. Barua, "Password-based encrypted group key agreement," *International Journal of Network Security*, vol. 3, no. 1, pp. 23–34, 2006.

[15] H. Guo, Z. Li, Y. Mu, and X. Zhang, "Provably secure identity based authenticated key agreement protocols with malicious private key generators," *Information Science*, vol. 181, no. 3, pp. 628–647, 2011.

[16] S. Hong, "Queue-based group key agreement protocol," *International Journal of Network Security*, vol. 9, no. 2, pp. 135–142, 2009.

[17] A. Joux, "One round protocol for tripartite diffie-hellman," *Journal of Cryptology*, vol. 17, no. 4, pp. 263–276, 2004.

[18] J. Katz and M. Yung, "Scalable protocols for authenticated group key exchange," in *Proceedings of Crypto'03*, LNCS 2729, pp. 110–125, 2003.

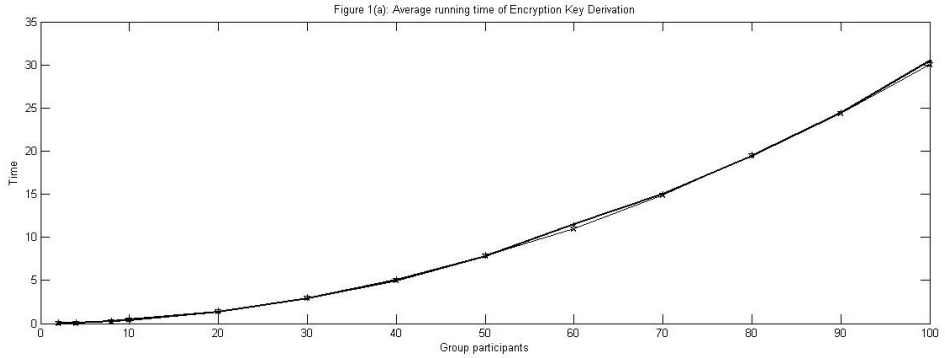


Figure 1: Average running time of encryption key derivation

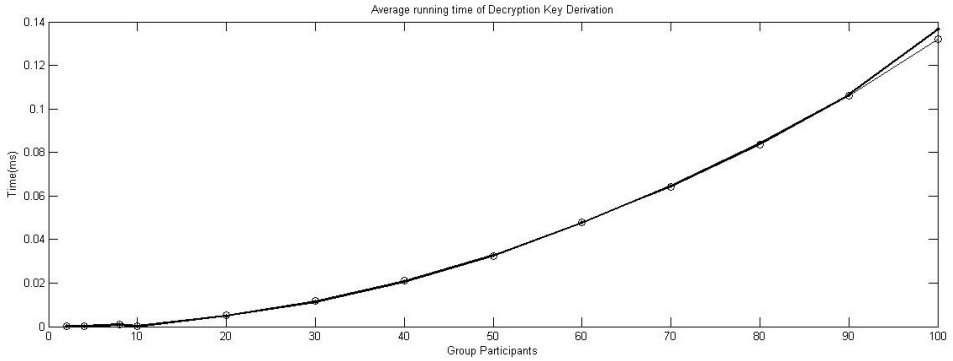


Figure 2: Average running time of decryption Key Derivation

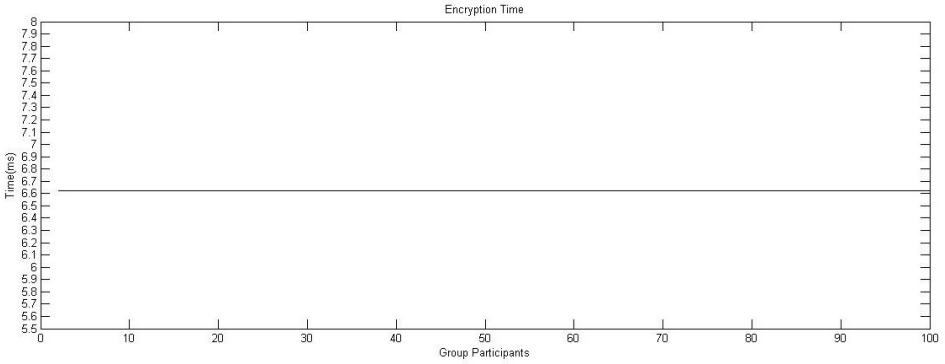


Figure 3: Message encryption time

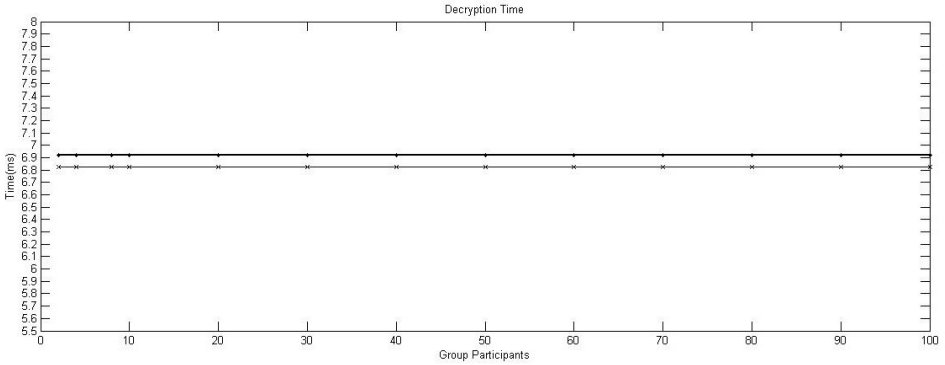


Figure 4: Message decryption time

[19] D. Li and S. Sampalli, "A hybrid group key management protocol for reliable and authenticated rekeying," *International Journal of Network Security*, vol. 6, no. 3, pp. 270–281, 2008.

[20] R. Sivaranjani, D. L. Bhaskari, and P. S. Avadhani, "Current trends in group key management," *International Journals of Advanced Computer Science and Applications*, vol. 2, pp. 82–86, 2011.

[21] A. Shamir, "Identity based cryptosystem and signature schemes," in *Advances in Cryptology (Crypto'84)*, LNCS 196, pp. 47–53, 1984.

[22] R. Srinivasan, V. Vaidehi, R. Rajaraman, S. Kanagaraj, R. C. Kalimuthu, and R. Dharmaraj, "Secure group key management scheme for multicast networks," *International Journal of Network Security*, vol. 11, no. 1, pp. 33–38, 2010.

[23] Q. Wu, Y. Mu, W. Susilo, B. Qin, and J. Domingo-Ferrer, "Asymmetric group key agreement," in *Proceedings of EUROCRYPT'09*, LNCS 5479, pp. 153–170, 2009.

[24] Q. Wu, Y. Mu, W. Susilo, B. Qin, and J. Domingo-Ferrer, "Provably secure one-round identity based authenticated asymmetric group key agreement protocol," *Information Sciences*, vol. 181, no. 19, pp. 4318–4329, 2011.

[25] L. Zhang, B. Qin, Q. Wu, and F. Zhang, "Efficient many-to-one authentication with certificateless aggregate signatures," *Computer Networks*, vol. 54, no. 14, pp. 2482–2491, 2010.

[26] L. Zhang, Q. Wu, U. G. Nicolas, B. Qin, and J. Domingo-Ferrer, "Asymmetric group key agreement protocol for open networks and its application to broadcast encryption," *Computer Networks*, vol. 55, no. 15, pp. 3246–3255, 2011.

[27] L. Zhang, F. Zhang, Q. Wu, and J. Domingo-Ferrer, "Simulatable certificateless two party authenticated key agreement protocol information," *Information Science*, vol. 180, no. 6, pp. 1020–1030, 2010.

Reddi Siva Ranjani is a research scholar in Andhra University under the supervision of Prof.P.S.Avadhani and Prof.D.Lalitha Bhaskari in Computer Science and Systems Engineering. She received her M.Tech (CSE) from Andhra University and presently working as Associate Professor in CSE Department of GMRIT. She is a Life Member of ISTE. Her research areas include Network Security, Cryptography, Group Key Management.

D. Lalitha Bhaskari is an Professor & HOD in the department of Computer Engineering, Andhra University college of Engineering for women . She is guiding more than 8 Ph. D Scholars from various institutes. Her areas of interest include Theory of computation, Data Security, Image Processing, Data communications, Pattern Recognition. She is a Life Member of CSI and CRSI. Apart from her regular academic activities she holds prestigious responsibilities like Associate Member in the Institute of Engineers, Associate Member in the Pentagram Research Foundation, Hyderabad, India. She also received young engineer award from Institute of Engineers (India) in the year 2008.

P. S. Avadhani is a Professor in the department of Computer Science and Systems Engineering and Vice Principal of AU College of Engineering, Andhra University. He has guided 10 Ph. D students and right now he is guiding 12 Ph. D Scholars. He has guided more than 100 M.Tech. Projects. He received many honors like best researcher award and best academician award from Andhra University, chapter patron award from CSI for CSI-Visakhapatnam Chapter and he has been the member for many expert committees, member of Board of Studies for various universities, Resource person for various organizations. He has coauthored 4 books. He is a Life Member in CSI, AMTI, ISIAM, ISTE, YHAI and in the International Society on Education Technology.

A Security Quantitative Analysis Method For Access Control Based on Security Entropy

Tian-Wei Che[1], Jian-Feng Ma[1], Na Li[2], Chao Wang[3]
(Corresponding author: Tian-Wei Che)
School of Computer Science and Technology, Xidian University, Xi'an, Shanxi, China[1]
(Email: tianweiche@163.com)
School of Computer Science and Technology, Northwestern Polytechnical University, Xi'an, Shanxi, China[2]
The information Engineering University, Zhengzhou, Henan, China[3]
(Received May 9, 2014; revised and accepted Jan. 16 & Apr. 21, 2015)

Abstract

This paper has proposed a quantitative analysis method based on security entropy to work out the problem of quantitative analysis in classificatory information systems. Firstly, the security theorems of classificatory information systems have been defined, the uncertainty for the system's determinations on the irregular access behaviors by means of the theory of security entropy have been calculated. Then a security quantitative analysis method has been proposed. By the security method, the typical access control models have been analyzed, security and applicability of these models have been compared. Finally, the method's practicability is validated and has been proved to be suit for security quantitative analysis on access control model and evaluation to access control capability in information system.

Keywords: Access control model, access violations, security entropy, unauthorized access

1 Introduction

Access control is a kind of security technology to achieve the established security strategy. The goal is to prevent users from unauthorized access to information resource. On the basis of the security policy to control user's access behavior, access control capability is directly related to the system access control of information system security attributes such as confidentiality, integrity. Access control is one of the five basic ISO7498-2 security services. There are several forms of unauthorized access in the system, such as explicitly access behavior, indirect access behavior in violation of the access control matrix and other access behavior in violation of information flow. Under normal circumstances, since there are obvious differences to detect these unauthorized access behavior's methods and models, so a problem which we had to face is how to quantify and measure an access control system (or model) for different unauthorized access behavior, in other words, how to calculate the possibility of uncertainty for all kinds of unauthorized access behavior in a system.

Classic access control model mainly has the BLP model, access control matrix model [3, 8], no interference model , RBAC [11] and so on. But research on the model of security measure theory of access control is still insufficient and inadequate, only for a single access to identify behavior is illegal, and fail to prevent unauthorized access behavior indirectly. Even recognized safety higher BLP model [3, 11] also can only prevent information flow from high level security to low security level by the indirect unauthorized access behavior and how to control Effectively indirect unauthorized access behavior to cause the data flow between subject and object in the same security level is powerless.

Due to the lack of safety for quantitative analysis and test method of access control strategy, for the information system managers' selection and application of appropriate access control policy or security mechanism caused confusion and difficulties. Because the information entropy theory has been applied in many fields, so far Information entropy has been successfully introduced it into the quantification analysis of information security risk and event uncertainty [4, 6, 7, 14]. On the basis of the information entropy can be measured the uncertainty things, an access control capability evaluation method is proposed which provides a scientific method for the quantitative analysis of hierarchical security access control model.

2 Weighted Entropy

Information Entropy is the tool to evaluate average uncertainty of event. Its definition is as follows, Let X be discrete random information source, its symbol set be $K : k_i$ $(i = 1, 2, \cdots, q)$, q is size of symbol set, the probability of event k_i is $P(k_i)$, its probability space $[X, p(x)]$ is defined

as:

$$\begin{bmatrix} X \\ P(x) \end{bmatrix} = \begin{bmatrix} k_1 & k_2 & \cdots & k_q \\ p(k_1) & p(k_2) & \cdots & p(k_q) \end{bmatrix}$$

The discrete random information source's information entropy is:

$$H(X) = -\sum_{i=1}^{q} p(x_i) \log p(k_i),$$

where $p(k_i) \geq 0 (i = 1, 2, \cdots, q)$ and $\sum_{i=1}^{q} p(x_i) = 1$.

In real environment, although the stochastic event happened with certain probability, but different event has different value and effect to people, and is of different importance. It is hard to ignore human factors. So, assign a nonnegative real $w_i \geq 0$ $(i = 1, 2, \cdots, q)$ to event k_i, the set of real is called weight of event. Let weight of information source's distribution $[X, w_i]$ be

$$\begin{bmatrix} X \\ w \end{bmatrix} = \begin{bmatrix} k_1 & k_2 & \cdots & k_q \\ w_1 & w_2 & \cdots & w_q \end{bmatrix}$$

Then information resource X's weighted entropy is

$$H_w(X) = -\sum_{i=1}^{q} w_i p(x_i) \log p(k_i).$$

3 Security Entropy

3.1 The Types of Access Security

In the information system, the access request is divided into two types: "legal" and "illegal", and the system's responds to user access request will be "allow" or "deny". So the response will be the four types: k_1 (allow legally access), k_2 (refuse legally access), k_3 (allow access violation) and k_4 (refuse access violation). Obviously, the response can be considered as a basis for judging if a system is good or bad. The more the denial responses to legitimate access gets, the poorer the system availability is. The more the allowable responses to violation access gets, the worse the system's confidentiality is.

In general, illegal access can be classified into three types:

1) Directly legally access;

2) Right about access;

3) Indirectly legally access.

The directly legally access refers to explicitly violating the authorized strategy such as the access control matrix and so on.

The right about access refers to the one which leads to violating information flow direction that the system stipulates, in other words, the one which leads information flow from high class to low class. The indirectly legally access refers to the one that violates the authorized strategy through information indirect transmission.

For instance, there is two users (u_1, u_2) and two resources $o_1, o_2)$ in the information system, and the relationship of security level is $f(u_1) \triangleright f(u_2) \triangleright f(o_1) = f(o_2)$, the authorized strategy of the system is that "u_1 read o_2", "u_2 read o_1", "u_2 write o_2".

The following are four events:

1) $b_1 : u_2$ read o_1;

2) $b_2 : u_2$ write o_2;

3) $b_3 : u_1$ read o_2;

4) $b_4 : u_1$ read o_1.

Because b_4 explicitly violates the authorized strategy, b_4 is therefore directly legally access; the Sequence of access $b_1 b_2 b_3$ cause the information to flow from u_1 into o_1, which equals that u_1 read o_1 indirectly. Therefore $b_1 b_2 b_3$ is indirectly legally access. b_1 and b_3 cause the information flowing to the violation of the direction made by the system, so b_1 and b_3 are right about access.

3.2 Definition of Security Entropy

Definition 1. *(Security Entropy) If the whole access requests are seen as the input, the system's request responses to each access result as the object of study, and the variable X as this response results, then the value of X will be: k_1, k_2, k_3, k_4. If the Symbol p_i stands for the statistical probability of k_i, and $p_i \geq 0$ $(i = 1, 2, 3, 4)$, $\sum_{i=1}^{4} p_i = 1$. Let $0 \leq w_i \leq 1$, $\sum_{i=1}^{4} w_i = 1$, the security entropy of X will be*

$$H(X) = -\sum_{i=1}^{4} w_i p_i \log p_i. \quad (1)$$

The w_i is the impact factor of the system security. The greater w_i is, the higher the k_i's influence to system safety is, otherwise the smaller the k_i's influence is. According to the common sense of information security, the response k_2 gives negative effects on the usability of the system, and the response k_3 gives negative effects on the confidentiality of the system, while the response k_1 and k_4 have less influence on system security. Therefore, if we let $w_2, w_3 \gg w_1, w_4$, the meaning of safety entropy in Equation (1) is the average uncertainty of the happened harmful responses. The bigger the value of security entropy is, the more the harmful response uncertainty is; the smaller the value of security entropy is, the less the response uncertainty is. As for the same set of access request, the smaller the security entropy of different access control model is, the less the possibility that model make harmful response is.

If $w_2 > 0$, $w_3 > 0$, $w_1 = w_4 = 0$, and at the same time $w_2 + w_3 = 1$, security entropy is the ground on which the system satisfies usability and confidentiality. If $w_2 = 1$, $w_3 = w_1 = w_4 = 0$, security entropy of Equation (1) will be the ground on which the system satisfies usability. If

$w_3 = 1$, $w_1 = w_2 = w_4 = 0$, security entropy of Equation (1) will be the ground on which the system satisfies confidentiality.

The number of the four responses is related to the number of input samples. If all input samples are legitimate accesses, k_3 and k_4 will be 0, and if all input samples are illegal access, k_1 and k_2 will be 0. In order to make the safety entropy reflect accurately the system security, the input samples must be complete. In addition, the responses are related to the number of input samples. If an input number of the access request is much more than others, the response will be distorted.

Therefore, when security entropy is calculated, the input samples (access requests) must be complete and its probability distribution must be uniform. The smaller the security entropy is, the less the Uncertainty of the harmful response that system do to is, the more the security of the model is. When the security entropy approaches 0, then the model will achieve the theoretical security.

3.3 Security Entropy of Different Types of Illegal Accesses

For the different types of legally access, the meaning of Equation (1) is different. If the legally access is defined as "directly legally access", the security entropy of Equation (1) is called "directly security entropy" recorded as $H_D(X)$.

Again, if the legally access is defined as "right about access", the security entropy will be $H_M(X)$: Mandatory security entropy. To "indirectly legally access", the security entropy will be $H_I(X)$: Indirectly security entropy.

4 Safety Theorem

Theorem 1. *(Direct Safety of Access Control Model) Access control model is direct safety, if and only if*

$$H_D(X) = -\sum_{i=1}^{4} w_i p_i \log p_i \equiv 0,$$

in which $w_1 = w_4 = 0$, $w_2 \neq 0$, $w_3 \neq 0$, $w_2 + w_3 = 1$.

Proof. Here we need to prove that when $H_D(X) \equiv 0$, the event "refuse legally access" and "allow access violation" will never happen, that is, $p_2 = p_3 = 0$. Because $w_1 = w_4 = 0$, so

$$H_D(X) = -w_2 p_2 \log p_2 - w_3 p_3 \log p_3.$$

If $H_D(X) \equiv 0$, must be $p_2 = p_3 = 0$. End. □

Similarly, we can get theorems as follows:

Theorem 2. *(Mandatory Safety of Access Control Model) The access control model has mandatory safety, if and only if*

$$H_M(X) = -\sum_{i=1}^{4} w_i p(a_i) \log p(a_i) \equiv 0,$$

in which $w_1 = w_4 = 0$, $w_2 \neq 0$, $w_3 \neq 0$, $w_2 + w_3 = 1$.

Theorem 3. *(Indirectly Safety of Access Control Model) The access control model has indirectly safety, if and only if*

$$H_I(X) = -\sum_{i=1}^{4} w_i p(a_i) \log p(a_i) \equiv 0,$$

in which $w_1 = w_4 = 0$, $w_2 \neq 0$, $w_3 \neq 0$, $w_2 + w_3 = 1$.

5 Analysis of Typical Access Control Model Based on Security Entropy

Now, we apply the theory to analyze the security of typical access control model, verify the practicability of this method, and point out the defect of these access control model.

5.1 Security Analysis to HRU Model

Directly Safety. Suppose there are m users in the system: $u_1, u_2, \cdots, u_m$, n resources: o_1, o_2, $\cdots$, o_n. Access requests can be divided into read and write atomic request, so there will be $2mn$ access request, which can be expressed respectively by symbol $b_1, b_2, \cdots, b_q$ $(q = 2mn)$. Results of the access can be divided into two kinds: legitimate access $B^+ = b_1^+, b_2^+, \cdots, b_s^+$, and direct legally access $B^- = b_1^-, b_2^-, \cdots, b_t^-$ $(s + t = q)$.

Based on the access control matrix, HRU [6] controls access behaviors. As long as access behaviors disobey the policy, it would be refused. So the responds to any $b_j^- \in B^-$ is k_4. As long as access behaviors don't disobey the policy, it would be allowed, so the responds to any $b_t^+ \in B^+$ is k_1, so $p_2 = 0$ and $p_3 = 0$.

The statistical probability distribution of responses is

$$\begin{bmatrix} X \\ P(x) \end{bmatrix} = \begin{bmatrix} k_1 & k_2 & k_3 & k_4 \\ \frac{s}{q} & 0 & 0 & \frac{t}{q} \end{bmatrix}$$

Since $H_D(X)|HRU \equiv 0$, the model HRU is direct safety.

Mandatory Safety. Divide all requests $B = b_1, b_2, \cdots, b_s$ $(q = 2mn)$ into three kinds: the requests $B^\uparrow = b_1^\uparrow, b_2^\uparrow, \cdots, b_{q/4}^\uparrow$ that causes information to flow form the low level into the high level, the requests $B^\downarrow = b_1^\downarrow, b_2^\downarrow, \cdots, b_{q/4}^\downarrow$ that causes information to flow form the high level into the low level, and the requests $B^\leftrightarrow = b_1^\leftrightarrow, b_2^\leftrightarrow, \cdots, b_{q/2}^\leftrightarrow$ that causes information to flow between the same level. Obviously, request $B^\downarrow$ in the second kind is a right about access.

Because the access control matrix is the base on which the model HRU judges the legality of the access request, the access request $b_i^{\uparrow}$ and $b_1^{\leftrightarrow}$ does not necessarily satisfy the access control matrix. It may be refused or allowed, because of which $p_2 \equiv 0$ cannot be always deduced.

Indirectly Safety. Indirectly illegal access is composed of several directly un-illegal accesses, so it can be denoted by $f_i^- = b_{i1}^+ b_{i2}^+ \cdots b_{iq}^+$, where $b_{i1}^+, b_{i2}^+, \cdots, b_{iq}^+ \in B^+$. Because $H_D(X)|HRU \equiv 0$, the system will allow every directly un-illegal access in f_i^-. Consequently, f_i^- will be allowed, therefore $p_3 > 0$ is deduced.

$H_I(X)|HRU > 0$, which shows that HRU model doesn't satisfy indirectly safety.

The above analysis shows that, the model HRU satisfies directly safety, and doesn't satisfy mandatory safety and indirectly safety.

5.2 Security Analysis to BLP

Directly Safety and Indirectly Safety.] The model BLP uses two methods: DAC and MAC. DAC uses the HRU model, so the directly safety and the indirectly safety of the BLP model coincide with that of the HRU, that is, BLP satisfies directly safety and doesn't satisfy indirectly safety.

Mandatory Safety. The BLP model forbids high level subjects writing low level objects and low level subjects reading high level objects, and prevents the information flowing from high level into low security level. So any right about access $b_i^{\downarrow} \in B^{\downarrow}$ will be refused by BLP, and any un-right about access $b_i^{\leftrightarrow} \in B^{\leftrightarrow}$ and $b_i^{\uparrow} \in B^{\uparrow}$ will be allowed. Consequently, the Probability distribution of BLP's response X is

$$\begin{bmatrix} X \\ P(x) \end{bmatrix} = \begin{bmatrix} a_1 & a_2 & a_3 & a_4 \\ \frac{q}{4} & 0 & 0 & \frac{q}{4} \end{bmatrix}$$

So, $H_M(X)|BLP \equiv 0$, which shows that BLP satisfies mandatory safety.

5.3 Security Analysis to RBAC

The model RBAC [9, 13] assigns roles for users, and then based on these roles grants authorization. The RBAC's rights management and access control manner is similar to HRU's. so its safety is similar to that of HRU, which is, satisfying directly safety and not satisfying mandatory safety and indirectly safety.

5.4 Security Analysis to FGBAC

The FGBAC [5, 10, 12] is the improved BLP, which introduces the information flow graph as a judgment auxiliary tool. In FGBAC, any directly illegal access, right about access and indirectly illegal access will be refused. So

$$\begin{aligned} H_D(X)|FGBAC &= H_M(X)|FGBAC \\ &= H_I(X)|FGBAC \\ &\equiv 0. \end{aligned}$$

It shows that, the model satisfies directly safety, mandatory safety and indirectly safety.

6 Conclusion

According to the characteristics of Information entropy, we introduce the concept of security entropy and provide an access control capability evaluation method for access control quantification analysis of information security risk and event uncertainty.

Acknowledgments

The research in this paper is supported by National Natural Science Foundation of China via grants numbers 60872041, 61072066 and Fundamental Research Funds for the Central Universities under grant JYl0000903001, JYl0000901034. The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] D. E. Bell and L. J. Lapadula, *Secure Computer Systems: Mathematical Foundations*, Technical Report M74-244, The MITRE Corporation, Bedford, Massachusetts, 1973.

[2] E. B. David, *Looking Back at the Bell-La Padula Model*, Reston VA, 20191, Dec. 7, 2005. (http://selfless-security.offthisweek.com/papers/looking-back.pdf)

[3] Z. Y. Fu, *Information Theory – Fundamental Theory and Application*, Beijing: Press of Electronics Industry, 2007. (in Chinese)

[4] GB/T 17859-1999, *Classified Criteria for Security*, Beijing: Standards Press of China, 1999. (in Chinese)

[5] Y. Liu, C. C. Chang, and S. C. Chang, "An access control mechanism based on the generalized aryabhata remainder theorem," *International Journal of Network Security*, vol. 16, no. 1, pp. 58-64, 2014.

[6] J. Peter, "Third generation computer systems," *Computer Surveys*, vol. 3, no. 4, pp. 175–216, 1971.

[7] R. S. Sandhu and E. J. Coyne, "Role-based access control models," *IEEE Computer*, vol. 29, no. 2, pp. 38–47, 1996.

[8] T. G. Si, Z. Y. Tan and Y. Q. Dai, "A security proof method for multilevel security models," *Journal of Computer Research and Development*, vol. 45, no. 10, pp. 1711–1717, 2008. (in Chinese)

[9] N. Sklavos and O. Koufopavlou, "Access control in networks hierarchy: implementation of key management protocol," *International Journal of Network Security*, vol. 1, no. 2, pp. 103-109, 2005.
[10] C. Wang, X. Y. Chen and N. Li, "An access control mode based on information flow graph," in *Proceedings of the International Conference on Computational Intelligence and Security*, pp.998-1000, 2011.
[11] G. B. Wang, H. Z. Huang and X. L. Zhang, "Risk possibility number – A new model for risk evaluation and prioritization based on maximum entropy theory," *Acta Aeronautica Et Astronautica Sinica*, vol. 30, no. 9, pp. 1684–1690, 2009. (in Chinese)
[12] M. Zareapoor, P. Shamsolmoali, and M. A. Alam, "Establishing safe cloud: Ensuring data security and performance evaluation," *International Journal of Electronics and Information Engineering*, vol. 1, no. 2, pp. 88–99, 2014.
[13] D. G. Zhai, Z. Xu, D. G. Feng, "Violation of static mutual exclusive role constraints in dynamic role transition," *Journal of Computer Research and Development*, vol. 45, no. 4, pp. 677–683, 2008. (in Chinese)
[14] D. M. Zhao, J. F. Ma, Y. S. Wang, "Model of fuzzy risk assessment of the information system," *Journal on Communications*, vol. 28, no. 4, pp.51–56, 2007. (in Chinese)

Tian-Wei Che was born in Xi'an, Shaanxi Province of China in 1971. He received the master degree in computer network and information security from the Information Engineering University, Zhengzhou, China in 2003. He is current Ph.D. candidate studying at School of Computer Science and Technology, Xidian University, and his supervisor is Prof. Jianfeng Ma. His main research interests include computer architecture, information security, and cloud computing.

Jian-Feng Ma was born in Xi'an, Shaanxi Province of China in 1963. He received his Bachelor of Science degree from the Department of Mathematics at the Shaanxi Normal University in July 1985; obtained his Master of Engineering degree in computer software from the Department of Computer software from the Department of Computer Science and Technology, Xidian University in March 1988. He earned his Doctorate of Engineering in communication and electronic system from the Department of Information Engineering, Xidian University. His major research fields include computer architecture, cryptology, information security, cloud computing and system survivability. He is Inside-school specially appointed professor; advisor of Ph.D candidates of computer system architecture and cryptology; director of the Ministry of Education /Ministry of Information Industry Key Laboratory of Computer Network and Information Security; dean of the School of Computer Science and Technology; outstanding returned student of Shaanxi province. Prof. Ma has published 7 books, and more than 200 research papers in journals and international conferences. In addition, Prof. Ma is the committee members of International Journals.

Na Li was born in Xi'an, Shaanxi Province of China in 1972. She received the master degree in computer network and information security from the Information Engineering University, Zhengzhou, China in 2004. She is currently Ph.D. candidate studying at School of Computer Science and Technology, Northwestern Polytechnical University, Xi'an, China. Her main research interests include computer information security and software Engineering.

Chao Wang was born in Zhengzhou, Henan Province of China in 1975. He received his Ph.D. degree in network and information security from the Information Engineering University, Zhengzhou, China in 2003. He works now as the associate professor in the Information Engineering University. He has published 3 books, and more than 10 research papers in journals and international conferences. His main research interests include computer architecture, information security, cloud computing.

Adoption of a Fuzzy Based Classification Model for P2P Botnet Detection

Pijush Barthakur[1], Manoj Dahal[2], and Mrinal Kanti Ghose[3]
(Corresponding author:Pijush Barthakur)
Department of Computer Applications, Sikkim Manipal Institute of Technology, Sikkim, India[1]
(Email: pijush.barthakur@gmail.com)
Novell IDC, Bagmane Tech Park, C V Ramannagar, Bangalore, India[2]
Department of Computer Science and Engineering, Sikkim Manipal Institute of Technology, Sikkim, India[3]
(Received Feb. 20, 2014; revised and accepted Nov. 15, 2014 & April 26, 2015)

Abstract

Botnet threat has increased enormously with adoption of newer technologies like root kit, anti-antivirus modules etc. by the hackers. Emergence of botnets having distributed C & C structure that mimic P2P technologically, has made its detection and dismantling extremely difficult. However, numeric flow feature values of P2P botnet C & C traffic can be used to generate fuzzy rule-set which can then be used to develop an efficient fuzzy based classification model. We generated fuzzy rule based models using Fuzzy Unordered Rule Induction Algorithm (FURIA) from C & C traffic collected from Nugache, Zeus and Waledac botnets. We also provide a comparative analysis of fuzzy based classification models with that of classification models obtained from C4.5 Decision Tree algorithm of Quinlan. Experimental results shows that using fuzzy based classification models, it is possible to achieve very promising result in predicting suspicious P2P botnet flows in the network and hence can be used for proactive detection of P2P botnets.

Keywords: Botnet, classification model, fuzzy unordered rule induction algorithm, P2P

1 Introduction

A botnet is a coordinated group of compromised machines controlled via Command & Control (C&C) communication channels that are connected to some C & C servers/peers and managed by botmasters/botherders. It can be used in performing various malicious activities like sending spam mails, distributed denial-of-service (DDoS) attacks, phishing attacks and click frauds [16]. Detection of such framework is never easy because of wide distribution of its bots and C & C servers that spread across thousands of individual networks around the globe. Thus, number of bots in a given network might be very less. Moreover, botnet C & C traffic is usually low in volume and is usually hidden in existing application traffic [18]. Complexities have further increased with shifting from its traditional Internet Relay Chat (IRC) protocol for C & C operations to more general and commonly used protocols like HTTP, POP3, Peer-to-Peer(P2P) etc. The most popular method of managing botnet C&Cs in recent past is the use of IRC, either in standard form or through use of customized implementations of IRC servers and clients intended to thwart mitigation efforts [26]. IRC based botnets uses centralized topological structures for C & C operations. Botnets with centralized C & C suffer from single-point-of-failure problem i.e. if C & C is detected and taken down the botnet cripples. However, IRC botnets with their source codes widely available and their setup and maintenance simple and relatively easy, are still the most popular among bot-herders. The newer and more resilient variants of botnet have emulated Peer-to-Peer technologically for C & C operations. A P2P botnet uses distributed C & C architecture that avoids single-point-of-failure problem. Also, newer P2P botnets are using advanced techniques like Rootkits, Fast-flux etc. to avoid detection. However, there also exists a newer and stealthy variant of centralized C & C architecture that uses HTTP for C & C operations. C & C traffic of HTTP botnets hide behind normal web traffic to pass through firewalls and other detection tools used by security personnel.

A pure peer-to-peer botnet is a decentralized architecture allowing bot-master to use any peer at random to distribute commands to other peer-bots in the P2P network. Some of the well-known P2P botnets are Nugache [26], Trojan.Peacomm or Storm [26] and Waledac [25]. Nugache is the pure-P2P bot artifact that does not depend on any central server including DNS. It handles C & C through encrypted P2P Channel using a variable bit length RSA key exchange, which is used to seed symmetric Rijndael-256 session keys for each peer connection. A new Nugache peer joins the network through an already known active servant peer in the network and each Nu-

gache peer may maintain a list of up to 100 servant peers for future use in rejoining the network. Nugache peers maintain an in-degree of connections that totals no more than ten clients at any time. The out-degree varies, but it is typically less than half of the ten-client limit. The result is a typical peer with at most about 13-15 connection active at any given time. Storm uses Overnet protocol initially to join the P2P network and then to keep track of the state of the network with other overnet peers. For initial peer seeding, the storm binary carries a text file containing IP addresses of approximately 300 static peers. However, the core of Storm C & C is handled via TCP using pull C & C technology directed at servers, i.e. server do not push commands down to the clients; rather, clients pull data from server. Storm uses a Hash mechanism for encrypting data requests to peers and servers. Instead of overnet, Waledac uses HTTP communication and a fast-flux based DNS network exclusively. In order to make initial contact with the botnet, each Waledac binary carries a list of IP addresses to use as a bootstrap list. Additional resiliency is provided in Waledac binaries through a hardcoded URL to access the botnet in the event a bot is unable to find an active node in the bootstrap list. The domain used for the URL is part of the fast flux network created by the botnet. Waledac botnet consists of three hierarchical layers of servers TSL servers, Upper tier servers (UTS) and Head End C&C. At the bottom are Repeater nodes and Spammer nodes. Those infected hosts having private IP addresses are spammer nodes. Repeaters are used primarily to move requests and replies between spammers and the head-end C & C server. Repeater nodes form the Repeater layer which is connected using P2P network. Each peer bot in the repeaters layer contains a node table having a maximum capacity of 500 to 1000 entries (depending on the version of binary). Waledac assigns each IP in the node table a timestamp, in order to keep this list as fresh as possible, by replacing the older entries with newer ones. P2P Zeus [2] is the decentralized version of the popular credential- stealing trojan Zeus. Earlier centralized version is mainly known for stealing banking credentials, where as P2P Zeus is also used for stealing Skype and MSN database files, Bitcoin wallets etc. With the adoption of P2P for communication, Zeus network has become more resilient against take down efforts. P2P Zeus uses RSA-2048 to sign sensitive messages originating from the bot-masters such as updates and proxy announcements. Peerlist poisoning is made difficult due to per-bot IP filter which only allows a single IP per / 20 subnet. P2P Zeus also includes an automatic blacklisting mechanism, which blacklists IPs that contact a bot too frequently in a specified time window. This mechanism further complicates efficient crawling and poisoning of the network. The C & C communications in a P2P Zeus network can be categorized into two parts:

1) Bots exchange binary and configuration updates with each other. P2P Zeus bots check the responsiveness of their neighbors every 30 minutes. Each neighbor is contacted in turn, and given 5 opportunities to reply. If a neighbor does not reply within 5 retries, it is deemed unresponsive, and is discarded from the peer list. During this verification round, every neighbor is asked for its current binary and configuration file version numbers. If a neighbor has an update available, the probing bot spawns a new thread to download the update.

2) Bots exchange list of proxy bots, which are designated bots where stolen data can be dropped and command can be retrieved. Additionally, bots exchange neighbor lists (Peer lists) with each other to maintain a coherent network.

Apart from P2P botnets, a new trend in the evolution of botnets is the rise of botnets that spread through social networking sites. One of the largest social networking botnet is KOOBFACE [27], and its infection starts with a spam sent through Facebook, Twitter, MySpace, or other social networking sites containing a catchy message with a link to a video. Each bot in the Koobface botnet connects to any one of roughly a hundred compromised hosts acting as C & C master servers that disseminate spam instructions. The Koobface C&C is a fully-connected graph where each master server is aware of every other master server.

In this paper, we propose a fuzzy rule-set through application of Fuzzy Unordered Rule Induction Algorithm (FURIA) [12] on flow attribute values of P2P botnet command & control traffic. A flow is defined by <source IP, destination IP, protocol, source port, destination port>. Fuzzy logic often leads to creation of small rule, where each rule is an embodiment of meaningful information. Moreover, we believe that there is an inherent fuzziness in security issues and an approximate fuzzy rule set can be generated for detection of security threats. In our earlier work, we proposed a rule induction algorithm [5] using indirect method of rule generation from C4.5 algorithm [21] of Quinlan. Inference using conventional rules proposed in paper [5] depends on crisp boundaries that lead to abrupt transition between the two classes. However, a more general rule where its support for a class decreases from full (inside the core of the rule) to zero (near the boundary) in a gradual rather than an abrupt way is more appropriate. Therefore, a set of fuzzy rules that have soft boundaries definitely has merit.

Our approach is a proactive novel approach for detecting likely P2P botnet C & C traffic flows through identification of significant flow-level features of P2P botnet C & C traffic. Flow level features are basically aggregation of packet-level features in that flow. Thus, our approach can handle encrypted traffic and is also free from privacy issues. The core of our detection approach relies on identification of likely botnet C & C traffic flows through development of efficient machine learning based models and then correlating the marked botnet flows to identify the group of flows that belong to the same bot-

net. Hypotheses that forms our detection approach are stated as follows:

1) A bot is a program and therefore has a limited set of commands and every command issued by a bot in its normal C & C operations is followed by a response from either a server in its hierarchy in the botnet or from some other bot in its peer group. In other word, C & C interactions in P2P botnets must follow a strict command-response pattern and the manner in which a bot responds to a specific set of commands are also more-or-less uniform.

2) A P2P bot needs to keep itself updated about other bots that are still active in its network and therefore needs to keep communicating with them.

3) In normal C & C operations, a P2P bot establishes numerous small sessions. More specifically, they keep changing communicating ports for normal C & C interaction or until they lunch attack. Therefore, the number of packets in each of the bot generated flow during normal C & C operation is usually small.

4) We also observe that most of the packets in bot generated flows are small in size i.e. the size of the largest packet in most of the bot generated flows is less than the MTU. This is to keep privacy and to avoid detection by not influencing normal internet services.

5) Among the few packets transferred in a bot generated flow,the largest sized packets are transferred at a specific proportion (usually < 1), whereas, the normal P2P traffic carries most of the packets to the size of MTU.

6) Finally, we observed that each bot generates mutually similar communicating flows to its peers in the same P2P botnet.

Rest of the paper is organized as follows: Section 2 provides a brief overview of related works. In Section 3, we provide a brief overview of botnet detection problem using network flows and the proposed architectural overview. In Section 4 we discuss our approach for dataset preparation and description of features selected for classification. In Section 5, we briefly describe the fuzzy rule generation algorithms used for botnet C & C traffic classification. In Section 6, we provide a detail analysis of results obtained from our classification models. In Section 7, we elaborate on future works and also the conclusion.

2 Related Works

Botnet threats have been continuously growing with adoption of newer technologies and propagation techniques by the bot-masters. Much resiliency has been achieved by recent botnets through migration from purely centralized C & C architecture to a partly or wholly decentralized architecture. New botnets have emerged on other digital devices like mobile phones or Smartphones. Mobile devices could send SMS and MMS to connect to their C & C proxy servers. Emergence of Online Social Network (OSNs) botnets is another recent development. A recently published survey paper [18] has vividly covered mobile botnets (e.g. iKee.B an iPhone bot) and OSNs botnets (e.g. KOOBFACE). Compared to enormity of threats posed by recent botnets, the detection and mitigation approaches developed till date is simply inadequate. Most of the botnet detection approaches are based on the anomalies being observed in network traffic, unusual system behavior etc. Botnet detection based on anomalies may not be useful always for several reasons. First, anomalies may not be always prominent to indicate a botnet attack. Second, it requires continuous monitoring of the network. Third, traffic belonging to botnets using HTTP protocol hides under the cover of normal web traffic and thus gets passed through everywhere.

There is a long time gap between initial infection with bot codes and its final deployment for active participation in botnets activity. This pre-attack period involves many stages such as, the process of rallying, i.e. the procedure adopted by a botnet for self-identification of newly created bots so that it can initiate contact with Command & Control (C & C) server, and the process of securing the newly created bot client. Measures taken to make a bot client secure normally involves deployment of anti-antivirus tools and Rootkit [20] or similar tools in order to hide itself from applications already installed by security agencies. In a newly created bot client, the hacker also employs tools to retrieve details of the computer (e.g. processor speed, memory, network speed etc.) and to search for location of any leftover tools by an earlier infection [24]. It is imperative to study botnet behavior during these early phases of exploitation in order to neutralize a bot possibly before its active participation in malicious activities. We may term such detection approaches as proactive. Many good reactive techniques [10, 11] have been suggested so far for botnet detection. Reactive techniques are about anomaly being observed in compromised machines mainly due to its use in cyber attacks or its exploitations over a long period of time. However, the aforementioned time gap is a good pointer for annihilation of bots before it causes any damage.

Lin et al. [14] proposed an automatic classification of obfuscated bot binaries by using system call sequences. The framework tested on 2256 binaries, achieves a 94% true positive rate and 93% true negative rate. A bot detection mechanism on a single host, proposed by Soniya et al. [3] initially identifies suspicious traffic by filtering out normal traffic from traffic generated on a host. For filtering out normal traffic, normal profiles of users are created. Suspicious traffic is then subjected to detail analysis based on observations made from characterization of bot traffic. A game bot detection framework through analysis of temporal characteristics of online game traffic has been proposed by Lu et al. [17]. The proposed approach is based on modelling of gaming behaviors of game bots

using Hidden Markov Models (HMM). The proposed approach can detect game bots accurately with only a small number of training traces.

Basheer et al. [1] proposed BotDigger, which utilizes fuzzy logic to derive logical rules based on defined botnet characteristics. The system uses fuzzy rule base to identify suspicious hosts within the set of monitored hosts after filtering out unlikely flows. The conserved flows are correlated with each other, looking for group of flows that may be part of same botnet. The fuzzy rule base is generated using few attributes that characterize IRC and HTTP based botnet activities. Kuochen et al. [28] uses fuzzy pattern recognition techniques to propose a novel behavior-based botnet detection system based on frequently observed bots TCP and DNS behaviors. The proposed system attempts to identify malicious domain names and IP addresses using maximum membership principle. The system achieved false positive rate of 0 - 3.08%, leaving room for further improvement. However, the proposed algorithm can detect inactive bots, which can be used to identify vulnerable hosts. Roshna et al. [23] proposed a botnet detection technique using Adaptive Neuro Fuzzy Inference System (ANFIS), which is a kind of neural network that incorporates the techniques of fuzzy inference system. However, there is scope for improvement in results claimed in this research work. Another fuzzy based intrusion detection system [9] has been proposed to detect anomalous network traffic by comparing with a behavioral model of normal network activity. A fuzzy rule based detection model is better suited for adaptive anomaly based intrusion detection when compared with static models due to changes in network traffic patterns over time.

3 Problem Description and Architectural Overview

Data captured from various applications in the internet involving different data types, such as files, e-mails, web contents, real-time audio/video data streams etc., are heterogeneous in terms of volume, time etc. Flows involving such data streams are in many cases unidirectional. Types of data transfer that needs to be reliable such as transfer of files, e-mails, Web contents etc. use Transmission Control Protocol (TCP) as transport layer protocol, whereas for transfer of real-time audio/video data streams, which is time-sensitive, the User Datagram Protocol (UDP) is typically used. Most P2P applications use UDP protocol for communication. Unlike normal web traffic, packets captured from botnets are largely uniform in terms of volume, time etc.

Network flows are extracted from packets captured from network traffic for both normal as well as botnet C & C traffic. Network flow level features used for classification are actually aggregate of packet-level features extracted from packet header. We derive twin advantages from our approach. First, we completely avoid packets payload analysis that involves a high amount of privacy issues. Second, our approach can handle encrypted traffic. Figure 1 shows architectural overview of our proposed Fuzzy rule based detection framework. It has two major components, first one is a module for extraction of flows from raw packets, and the second one generate fuzzy rules for classification.

4 Dataset Preparation and Feature Selection

We describe below the dataset preparation and feature selection procedure of our experiment.

4.1 Dataset Preparation

Botnet datasets used in this work were collected from the following sources: The Nugache botnet C & C traffic was obtained from Department of Computer Science, The University of Texas at Dallas. This is the same botnet traffic sample used in the botnet related research works of [19]. Similarly, Zeus and Waledac traffic traces were obtained from Department of Computer Science, University of Georgia. These are the botnet traces used in the botnet related research works of [22]. We acquired the benign data randomly from different machines in our campus network using Wireshark [13]. Our campus network is protected using network level firewall. Though this device has its own limitations, it is believed that the device can prevent malicious attacks entering the protected unit by setting different network zone and the rules that control the access in and out flow [16]. Moreover, we collected data from known benign applications only. Therefore, we assume that the data collected is benign.

We prepared three datasets having 20,000 flows each, one each for flow extracted for Nugache, Zeus and Waledac traces. Datasets are prepared in such a way, that each has 15000 flows of botnet C & C traffic and 5000 flows of benign traffic. Our benign traffic samples include varied traffic such as HTTP, FTP, SMTP etc. We also include traffic captured from legitimate P2P application in our benign dataset. Datasets are then labelled accordingly. While preparing the datasets, we discarded flows that are unlikely to contribute significantly in the process of classification viz.

1) Flows having single packet;

2) Flows that involves local area network broadcast activities.

Reasons for discarding these flows are as follows:

1) Flows carrying single packet does not carry any meaningful statistical information, and the proportion of largest sized packet attribute values in our dataset would become 1.

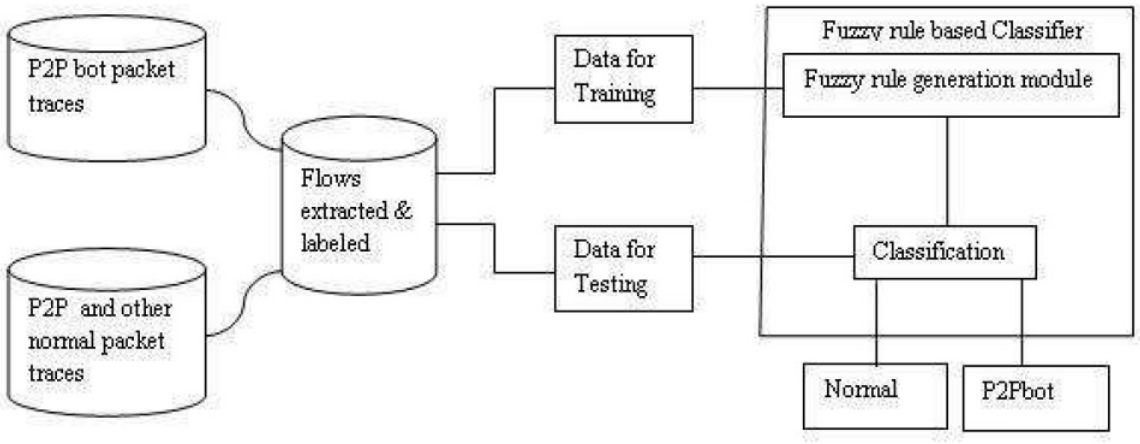


Figure 1: Architecture of the proposed Fuzzy rule based detection framework

2) The bot infected hosts may involve in local broadcasts activities. However, our objective is to consider host-to-host directed interaction in the network and broadcast traffic is never part of bot C & C interaction. Therefore, we tag such traffic as unwanted for our classification model. After removing unwanted flows, we scaled the datasets to the range of 0 to 1.

In our architectural framework shown in Figure 1, the first component has a module to extract flows from raw data. Then the attributes in the flows are scaled and useless flows are deleted. The final task of this component is to label the retained flows. The second component takes as input the dataset containing refined flows prepared by the first component and generates fuzzy rules for classification. We perform botnet C & C traffic classification using 10-fold cross validation. In general, in n-fold cross validation, the training set is first divide into n subsets of equal size. Sequentially one subset is tested using classifier trained on remaining n-1 subsets. Finally, when all subsets are tested, n results from folds are averaged to produce a single estimation.

4.2 Feature Selection

Detail analysis of behavioral characteristic of botnet C & C traffic flow was conducted after which, useful features for classification were extracted from packet headers. Following are the botnet flow and behavior characteristic features used in this work:

1) Total packets transferred (TPT): Number of packets transferred (or packet count) in a flow. It is a flow direction dependent attribute i.e. the numeric value of the attribute may be different for command and response flows within the same pair of peer bots.

2) Largest sized packet (LSP): Size of the packet carrying maximum bytes in a flow. It is also flow direction dependent attribute.

3) Total bytes transferred with largest sized packets (TBLSP): It is the multiplication of total number of largest sized packets (LSP) and the size of the largest packet.

4) Total bytes transferred (TBT): It is the summation of bytes transferred with all the packets in a flow. It is also flow direction dependent attribute.

5) Proportion of largest sized packet (PLSP): It is the ratio of largest sized packet transferred in a flow. It is also flow direction dependent attribute.

6) Variance of inter-arrival time (VIT): Variance calculated for inter-arrival time of packets within a flow. It is also flow direction dependent attribute.

7) Average packet length (APL): Average calculated for packet sizes of packets within a flow. It is also flow direction dependent attribute.

8) Variance of packet length (VPL): Variance calculated for sizes of packets within a flow. It is also flow direction dependent attribute.

9) Response packet difference (RPD): Difference in number of packets between two responding flows. The numeric value of this attribute is common for responding flows between a pair of hosts. For unidirectional flow (i.e. flow without a responding flow) we put a high numeric value for this attribute. For example, in our experiment we put 999, because the maximum difference is of three digits in our dataset.

10) Response time difference (RTD): Difference in time of last packet received for two responding flows between a pair of hosts. The numeric value of this attribute is also common for responding flows. For unidirectional flow (i.e. flow without a responding flow) we put a high numeric value for this attribute. For example, in our experiment we put 99999, because the maximum difference calculated in second is of five digits in our dataset.

5 Overview of FURIA

FURIA [12] is similar to well-known conventional rule learner RIPPER [7] with its distinctive features of generating fuzzy rules and of generating unordered rule sets instead of rule lists. By unordered, it means a set of rules for each class in a one-vs-rest scheme. In FURIA, fuzzy rules are obtained using fuzzy intervals derived with trapezoidal membership function. It uses four parameters stated as $I^F = (\Phi^{s,L}, \Phi^{c,L}, \Phi^{c,U}, \Phi^{s,U})$ to represent fuzzy intervals. The trapezoidal membership function for fuzzy sets (or fuzzy intervals) is given by:

$$I^F(v) \quad \stackrel{\text{df}}{=} \left\{ \begin{array}{ll} 1 & \Phi^{c,L} \leq v \leq \Phi^{c,U} \\ \frac{v-\Phi^{s,L}}{\Phi^{c,L}-\Phi^{s,L}} & \Phi^{s,L} < v < \Phi^{c,L} \\ \frac{\Phi^{s,U}-v}{\Phi^{s,U}-\Phi^{c,U}} & \Phi^{c,U} < v < \Phi^{s,U} \\ 0 & \text{else} \end{array} \right\} \quad (1)$$

$\Phi^{c,L}$ and $\Phi^{c,U}$ are, respectively, lower and upper bound of the core (elements with membership 1) of the fuzzy set; likewise, $\Phi^{s,L}$ and $\Phi^{s,U}$ are, respectively, the lower and upper bound of the support(elements with membership > 0). Thus, $\Phi^{s,L}$ and $\Phi^{s,U}$ are the fuzzy extensions of original RIPPER intervals [$\Phi^{c,L}$, $\Phi^{c,U}$] that are considered as core. Rules are fuzzified in a greedy way through fuzzification of every antecedent in a rule or in other word, through replacement of sharp boundaries of a rule with soft boundaries. Fuzzification of each antecedent is done by testing all relevant values $\{x_i | x = (x_1 \cdots x_k) \in D_i^T, x_i < \Phi_i^{c,L}\}$ as candidates for $\Phi_i^{s,L}$ and for all values $\{x_i | x = (x_1 \ldots x_k) \in D_i^T, x_i > \Phi_i^{c,U}\}$ as candidates for $\Phi_i^{s,U}$. Here, relevant data for each antecedent ($A_i \in I_i$) is the one considered by ignoring all those instances that are excluded by any other antecedent ($A_j \in I_j^F$), $j \neq i$: $D_T^i = \{x = (x_1 \ldots x_k) \in D_T | I_j^F(x_j) > 0$ for all $j \neq i\} \subseteq D_T$.

FURIA being a fuzzy rule generating algorithm is characterized by its core and its support. It is valid inside the core and invalid outside the support; in-between, the validity drops in a gradual way. Apart from having this definite advantage of fuzzy rule generation over other conventional rule generation algorithms such as RIPPER and C4.5, FURIA generates unordered rule set instead of rule lists and provides an efficient rule stretching method to deal with uncovered instances. All these features of the algorithm make it most suitable for rule generation for network security threat detection.

6 Results and Analysis

We use WEKA [8] Data Mining environment for fuzzy rule generation and subsequent classification of botnet C & C traffic flows of Nugache, Zeus and Waledac botnets. Weka provides a collection of Machine Learning (ML) algorithms and several visualization tools for data analysis and predictive modelling. We present results of our experiments in two parts: First, a brief analysis of structure of fuzzy rules generated for all bot flows is presented. Next we provide analysis of results using various performance metrics.

6.1 Analysis of Rule Sets

Unlike sharp boundaries generated by RIPPER, C4.5 etc. a fuzzy rule is characterized by soft boundaries. Each fuzzy rule consists of two parts: its "core"and its "support". For example, one of the rules generated from C & C traffic of Nugache botnet is:

(TBLSP in [-inf, -inf, 0.000006, 0.000006]) and (RPD in [-inf, -inf, 0.001, 0.002]) and (LSP in [0.0055, 0.0062, inf, inf]) ⇒ Class=bot (CF = 1.0).

The antecedents of the rule can be interpreted as: (1)TBLSP in [-inf, -inf, 0.000006, 0.000006]: it is valid for TBLSP $\leq$ 0.000006 and invalid for TBLSP $>$ 0.000006, (2) RPD in [-inf, -inf, 0.001, 0.002]: it is completely valid for RPD $\leq$ 0.001, invalid for RPD $>$ 0.002 and partially valid in-between,(3) LSP in [0.0055, 0.0062, inf, inf]: it is completely valid for LSP $\geq$ 0.0062, invalid for LSP $<$ 0.0055 and partially valid in-between. Now, performing logical AND operation for completely valid and partially valid cases of Part (1), (2) and (3) on the LHS of our above rule, we get the 'Coverage'of the rule in our dataset. Completely valid parts associated with antecedents of the rule are its 'core', whereas partially valid part forms the 'support'. The Certainty factor of the rule is 1. List of fuzzy rules generated that predict bot flows for Nugache, Zeus and Waledac are shown in Tables 1, 2 and 3 respectively.

In Table 4, we provide structural attribute values for comparative analysis of the structure of fuzzy rule sets generated for the three botnets. The attributes considered for comparison are: number of fuzzy rules generated (NFR), average number of antecedents in the rules generated for each botnet (ANAR), number of rules that predicts a bot flow (NRB), percentage of coverage of cases (PCC) and the number of rules with certainty factor 1.0 (NRCF).

From the structural attribute values, we find that least complex rules are generated for Waledac C & C traffic flows and the most complex rule set is generated for highly stealthy Zeus botnet. This can be observed from the number of fuzzy rules generated (NFR) and the average number of antecedents in the rules generated for each botnet (ANAR) attributes of the three botnet C & C traffic samples. Among the other attributes, Nugache rule set has 52% rules predicting bot flows followed by 42.5% for Zeus and 42.1% for Waledac. Similarly, percentage of rules with certainty factor 1.0 is 72% for Nugache, 47% for Waledac and 38.75% for Zeus. All these statistics along with the percentage of coverage of cases by the rule sets indicates that all the three botnet traces produced very efficient rule set.

Table 1: Fuzzy rules for detection of Nugache bot C & C traffic

Serial No	Rule
1	(TBLSP in [-inf, -inf, 0.000006, 0.000006]) and (RPD in [-inf, -inf, 0.001, 0.002]) and (LSP in [0.0055, 0.0062, inf, inf]) ⇒ Class=bot (CF = 1.0)
2	(TBLSP in [-inf, -inf, 0.000012, 0.000013]) and (TBT in [0.000034, 0.000036, inf, inf]) and (TBLSP in [0.000012, 0.000012, inf, inf]) and (LSP in [-inf, -inf, 0.0118, 0.0119]) and (APL in [-inf, -inf, 0.0072, 0.00828]) ⇒ Class=bot (CF = 1.0)
3	(LSP in [-inf, -inf, 0.0062, 0.0064]) and (VPL in [0, 0, inf, inf]) ⇒ Class=bot (CF = 1.0)
4	(APL in [-inf, -inf, 0.00725, 0.00828]) and (VPL in [0.000001, 0.000032, inf, inf]) and (LSP in [-inf, -inf, 0.0118, 0.0119]) and (LSP in [0.0115, 0.0118, inf, inf]) ⇒ Class=bot (CF = 1.0)
5	(TBLSP in [-inf, -inf, 0.000006, 0.000006]) and (TPT in [0.00003, 0.00004, inf, inf]) ⇒ Class=bot (CF = 1.0)
6	(APL in [-inf, -inf, 0.006133, 0.0062]) and (APL in [0.006, 0.006009, inf, inf]) and (VPL in [-inf, -inf, 0, 0.000001]) and (RTD in [-inf, -inf, 0.03273, 0.03388]) ⇒ Class=bot (CF = 1.0)
7	(APL in [-inf, -inf, 0.006189, 0.0062]) and (APL in [0.006, 0.006006, inf, inf]) and (VPL in [-inf, -inf, 0, 0]) and (VIT in [0, 0.0343, inf, inf]) and (RTD in [-inf, -inf, 0.06001, 0.06455]) ⇒ Class=bot (CF = 1.0)
8	(PLSP in [-inf, -inf, 0.04, 0.333333]) and (LSP in [0.0275, 0.0354, inf, inf]) and (TBLSP in [-inf, -inf, 0.000047, 0.000049]) and (VPL in [-inf, -inf, 0.000033, 0.00004]) ⇒ Class=bot (CF = 0.95)
9	(APL in [-inf, -inf, 0.0075, 0.00828]) and (LSP in [0.0115, 0.0118, inf, inf]) and (LSP in [-inf, -inf, 0.0118, 0.0119]) ⇒ Class=bot (CF = 1.0)
10	(APL in [-inf, -inf, 0.006093, 0.006133]) and (APL in [0.006, 0.006006, inf, inf]) and (VPL in [-inf, -inf, 0.000002, 0.000003]) and (TPT in [0.00008, 0.00009, inf, inf]) ⇒ Class=bot (CF = 0.99)
11	(PLSP in [-inf, -inf, 0.007937, 0.009524]) and (LSP in [0.0276, 0.0289, inf, inf]) and (VPL in [-inf, -inf, 0.00008, 0.000083]) and (RPD in [-inf, -inf, 0.154, 0.19]) ⇒ Class=bot (CF = 0.95)
12	(TPT in [0.00009, 0.00025, inf, inf]) and (LSP in [-inf, -inf, 0.006, 0.0065]) and (LSP in [0.0054, 0.0058, inf, inf]) ⇒ Class=bot (CF = 0.92)
13	(PLSP in [-inf, -inf, 0.04, 0.111111]) and (TBT in [-inf, -inf, 0.00034, 0.000344]) and (LSP in [0.0309, 0.0354, inf, inf]) and (VPL in [-inf, -inf, 0.000352, 0.000363]) ⇒ Class=bot (CF = 0.92)

6.2 Analysis of Classification Results

Final datasets prepared from botnet C & C traffic of the three bots under consideration are being used to build classification models using WEKA machine learning tools. We randomized flow instances in our datasets by passing it through Randomize filter available with WEKA's unsupervised instance filter category. This was necessitated because our original datasets are imbalanced having less normal web flows. While constructing classifier, we used 10-fold cross validation so that there is no over-fitting of our training set.

Results of Classification task by any classification algorithm during testing are usually displayed in a confusion matrix. A confusion matrix holds the count of the correct and incorrect classification from each class or the differences between the true and predicted classes for a set of labelled instances. Table 5 shows the format of a confusion matrix with TP, TN, FP, FN representing True Positive, True Negative, False Positive and False Negative counts respectively.

The row total, CN and CP are the number of truly negative and positive instances. Similarly, RN and RP are the number of predicted negative and positive instances, with N being the total number of instances ($N = CN+CP = RN+RP$). Although confusion matrix incorporates all the performance measures of a classification algorithm, more meaningful results can be extracted from it to represent certain performance criteria. Accuracy is the first performance criteria we are using to compare the three classification models on botnet datasets:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (2)$$

Figure 2 shows comparison of accuracy achieved with our fuzzy rule based classification models with that of decision tree based classification models. Decision trees can be used to generate crisp rule sets for classification. In this work we generate the decision tree from Quinlans famous C4.5 algorithm.

The percentage accuracy value achieved using FURIA are 99.745%, 99.715%, and 99.105% for Nugache, Waledac and Zeus flows respectively. Corresponding figures using C4.5 algorithm are 99.655%, 99.695%, and 98.615%. We find a distinct increase in correctly classified instances using fuzzy rule based classification models. We also observe that fuzzy based classifier is largely successful in classifying C & C traffic generated by stealthy botnets like Zeus, though accuracy achieved is lower than that of Nugache and Waledac. The increase in number of correctly classified flow instances by FURIA when compared with C4.5 algorithm is 18, 4 and 98 respectively for Nugache, Waledac and Zeus sample botnet datasets. This increase is because of fuzzification of classification rules by FURIA.

We also consider the following additional performance criteria to compare our fuzzy based classification models:

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

$$PositivePredictiveValue(PPV) = \frac{TP}{TP + FP} \quad (4)$$

$$FalsePositiveRate = \frac{FP}{FP + TN} \quad (5)$$

Sensitivity (or True Positive Rate) is the proportion of correctly identified bot flows out of total flows labelled as bot. Similarly, PPV or Precision is the proportion of correctly identified bot flows out of total predicted bot flows. Figure 3 shows graphical comparison of Sensitivity, PPV and FP rate of our three fuzzy based classification models. The graph shows a three dimensional view of changes in aforesaid three performance metrics values with respect to the three botnet sample data set in consideration. The results shown in the graph are in the range

Table 2: Fuzzy rules for detection of Zeus bot C & C traffic

Serial No	Rule
1	(VPL in [0.00017, 0.000171, inf, inf]) and (TPT in [-inf, -inf, 0.0006, 0.0007]) and (APL in [-inf, -inf, 0.013633, 0.015883]) and (LSP in [0.0308, 0.0324, inf, inf]) and (RTD in [-inf, -inf, 0.00252, 0.00253]) and (RTD in [0.00099, 0.0011, inf, inf]) ⇒ Class=bot (CF = 1.0)
2	(VPL in [0.043302, 0.043385, inf, inf]) and (TPT in [-inf, -inf, 0.0006, 0.0007]) and (VPL in [-inf, -inf, 0.049932, 0.052853]) and (RPD in [-inf, -inf, 0.001, 0.002]) ⇒ Class=bot (CF = 1.0)
3	(VPL in [0.000032, 0.000171, inf, inf]) and (LSP in [-inf, -inf, 0.02, 0.0202]) and (LSP in [0.0199, 0.02, inf, inf]) and (APL in [-inf, -inf, 0.00925, 0.01595]) ⇒ Class=bot (CF = 1.0)
4	(VPL in [0.000149, 0.00015, inf, inf]) and (LSP in [-inf, -inf, 0.0545, 0.0547]) and (LSP in [0.0523, 0.0529, inf, inf]) and (RTD in [-inf, -inf, 0.01448, 0.01463]) and (RTD in [0.00082, 0.001, inf, inf]) ⇒ Class=bot (CF = 1.0)
5	(VPL in [0.00006, 0.00015, inf, inf]) and (RPD in [-inf, -inf, 0.006, 0.999]) and (RPD in [0.001, 0.002, inf, inf]) and (RTD in [0.00497, 0.00805, inf, inf]) and (TBLSP in [-inf, -inf, 0.000055, 0.000062]) and (APL in [0.012533, 0.0126, inf, inf]) and (LSP in [-inf, -inf, 0.0335, 0.0365]) and (TBT in [0.00004, 0.000044, inf, inf]) ⇒ Class=bot (CF = 1.0)
6	(VPL in [0.00009, 0.000965, inf, inf]) and (RPD in [-inf, -inf, 0.001, 0.002]) and (RTD in [0.00105, 0.00122, inf, inf]) and (RTD in [-inf, -inf, 0.00218, 0.00219]) ⇒ Class=bot (CF = 1.0)
7	(VPL in [0.000656, 0.000942, inf, inf]) and (RPD in [-inf, -inf, 0.006, 0.043]) and (RTD in [0.00069, 0.00122, inf, inf]) and (LSP in [0.0523, 0.0529, inf, inf]) and (LSP in [-inf, -inf, 0.0546, 0.0547]) and (VPL in [-inf, -inf, 0.003206, 0.003209]) and (RTD in [-inf, -inf, 0.07234, 0.0724]) ⇒ Class=bot (CF = 1.0)
8	(RPD in [-inf, -inf, 0.012, 0.017]) and (TBLSP in [0.000012, 0.000018, inf, inf]) and (LSP in [-inf, -inf, 0.0062, 0.0066]) and (RTD in [-inf, -inf, 0.01018, 0.01052]) and (RTD in [0.00068, 0.00122, inf, inf]) ⇒ Class=bot (CF = 1.0)
9	(VPL in [0.000091, 0.00015, inf, inf]) and (RPD in [-inf, -inf, 0.012, 0.043]) and (RTD in [0.00001, 0.00122, inf, inf]) and (APL in [0.018127, 0.018409, inf, inf]) and (PLSP in [-inf, -inf, 0.142857, 0.166667]) and (VPL in [-inf, -inf, 0.002537, 0.002894]) and (RTD in [-inf, -inf, 0.06149, 0.06156]) ⇒ Class=bot (CF = 1.0)
10	(VPL in [0.000976, 0.001038, inf, inf]) and (RPD in [-inf, -inf, 0.005, 0.006]) and (RTD in [0.00078, 0.00091, inf, inf]) and (RTD in [-inf, -inf, 0.00332, 0.00818]) and (TBLSP in [-inf, -inf, 0.000034, 0.000036]) ⇒ Class=bot (CF = 1.0)
11	(VPL in [0.039564, 0.041469, inf, inf]) and (RTD in [0.00061, 0.00097, inf, inf]) and (RTD in [-inf, -inf, 0.03579, 0.04022]) and (TBLSP in [-inf, -inf, 0.000442, 0.000454]) and (TPT in [0.0002, 0.0003, inf, inf]) ⇒ Class=bot (CF = 1.0)
12	(VPL in [0.001119, 0.001122, inf, inf]) and (RTD in [0.00045, 0.001, inf, inf]) and (RPD in [-inf, -inf, 0.006, 0.999]) and (LSP in [0.0768, 0.0772, inf, inf]) and (LSP in [-inf, -inf, 0.0811, 0.0814]) ⇒ Class=bot (CF = 1.0)
13	(VPL in [0.000079, 0.000091, inf, inf]) and (RTD in [0.0493, 0.04944, inf, inf]) and (LSP in [0.0527, 0.0529, inf, inf]) and (LSP in [-inf, -inf, 0.055, 0.0552]) and (APL in [0.017975, 0.0182, inf, inf]) ⇒ Class=bot (CF = 0.99)
14	(VPL in [0.025873, 0.03012, inf, inf]) and (APL in [0.091857, 0.093214, inf, inf]) and (TBLSP in [-inf, -inf, 0.000757, 0.000908]) and (RTD in [0.00045, 0.00255, inf, inf]) and (RTD in [-inf, -inf, 0.0676, 0.06776]) ⇒ Class=bot (CF = 1.0)
15	(VPL in [0.002699, 0.002756, inf, inf]) and (LSP in [-inf, -inf, 0.0531, 0.0532]) and (LSP in [0.0527, 0.0529, inf, inf]) ⇒ Class=bot (CF = 1.0)
16	(VPL in [0.000065, 0.000091, inf, inf]) and (LSP in [-inf, -inf, 0.0335, 0.0343]) and (TBT in [0.000116, 0.000117, inf, inf]) and (APL in [0.016038, 0.01607, inf, inf]) and (TBLSP in [-inf, -inf, 0.000086, 0.000105]) ⇒ Class=bot (CF = 1.0)
17	(VPL in [0.001417, 0.001493, inf, inf]) and (TBLSP in [-inf, -inf, 0.000055, 0.000055]) and (LSP in [0.0522, 0.053, inf, inf]) and (RPD in [0.002, 0.003, inf, inf]) ⇒ Class=bot (CF = 1.0)
18	(VPL in [0.001059, 0.001108, inf, inf]) and (TBLSP in [-inf, -inf, 0.000034, 0.000034]) and (TBT in [0.00005, 0.000051, inf, inf]) and (TPT in [0.0003, 0.0004, inf, inf]) ⇒ Class=bot (CF = 1.0)
19	(VPL in [0.029541, 0.03012, inf, inf]) and (APL in [0.111343, 0.112722, inf, inf]) and (VIT in [-inf, -inf, 0.0123, 0.0149]) and (TBLSP in [-inf, -inf, 0.001211, 0.001363]) ⇒ Class=bot (CF = 0.99)
20	(RTD in [-inf, -inf, 0.00686, 0.02146]) and (RTD in [0.00066, 0.00073, inf, inf]) and (APL in [-inf, -inf, 0.00644, 0.006467]) and (LSP in [0.0074, 0.0098, inf, inf]) and (LSP in [-inf, -inf, 0.0098, 0.0122]) ⇒ Class=bot (CF = 0.99)
21	(VPL in [0.001957, 0.003153, inf, inf]) and (APL in [0.142629, 0.143664, inf, inf]) and (PLSP in [-inf, -inf, 0.536036, 0.676471]) ⇒ Class=bot (CF = 0.98)
22	(VPL in [0.040972, 0.041469, inf, inf]) and (RPD in [-inf, -inf, 0.005, 0.006]) and (RTD in [0.00047, 0.00058, inf, inf]) and (TBT in [0.000649, 0.000653, inf, inf]) ⇒ Class=bot (CF = 0.98)
23	(VPL in [0.01728, 0.022948, inf, inf]) and (RPD in [-inf, -inf, 0.006, 0.007]) and (RTD in [0.00795, 0.01713, inf, inf]) and (APL in [0.111506, 0.112722, inf, inf]) and (VIT in [-inf, -inf, 0.0123, 0.0139]) ⇒ Class=bot (CF = 0.98)
24	(LSP in [-inf, -inf, 0.0062, 0.0065]) and (TBLSP in [0.000013, 0.000018, inf, inf]) and (RTD in [-inf, -inf, 0.06825, 0.07552]) and (LSP in [0.006, 0.0062, inf, inf]) ⇒ Class=bot (CF = 0.98)
25	(VPL in [0.000088, 0.000116, inf, inf]) and (RPD in [-inf, -inf, 0.006, 0.008]) and (RPD in [0.003, 0.004, inf, inf]) and (VPL in [-inf, -inf, 0.000162, 0.000162]) ⇒ Class=bot (CF = 0.99)
26	(VPL in [0.00071, 0.000712, inf, inf]) and (LSP in [-inf, -inf, 0.0268, 0.0845]) and (TBT in [0.000036, 0.000037, inf, inf]) and (PLSP in [-inf, -inf, 0.4, 0.428571]) ⇒ Class=bot (CF = 0.93)
27	(VPL in [0.001417, 0.001905, inf, inf]) and (LSP in [-inf, -inf, 0.0548, 0.0549]) and (LSP in [0.0531, 0.0532, inf, inf]) and (APL in [-inf, -inf, 0.017625, 0.02075]) and (RPD in [0.001, 0.002, inf, inf]) ⇒ Class=bot (CF = 1.0)
28	(LSP in [-inf, -inf, 0.006, 0.0062]) and (TBLSP in [0.000016, 0.000018, inf, inf]) and (RTD in [-inf, -inf, 0.02393, 0.02514]) and (VIT in [0.0157, 0.0199, inf, inf]) ⇒ Class=bot (CF = 0.99)
29	(VPL in [0.026365, 0.028376, inf, inf]) and (RPD in [-inf, -inf, 0.004, 0.012]) and (LSP in [-inf, -inf, 0.1494, 0.1496]) and (LSP in [0.1468, 0.1472, inf, inf]) ⇒ Class=bot (CF = 1.0)
30	(LSP in [-inf, -inf, 0.0062, 0.0065]) and (TBLSP in [0.000018, 0.000019, inf, inf]) and (RPD in [-inf, -inf, 0.001, 0.006]) and (RTD in [-inf, -inf, 0.06889, 0.07552]) ⇒ Class=bot (CF = 0.99)
31	(RTD in [-inf, -inf, 0.05404, 0.06343]) and (APL in [0.009754, 0.012425, inf, inf]) and (TBLSP in [-inf, -inf, 0.000033, 0.000037]) and (TBT in [0.000112, 0.000116, inf, inf]) and (VIT in [0.0161, 0.0163, inf, inf]) ⇒ Class=bot (CF = 1.0)
32	(VPL in [0.000093, 0.000539, inf, inf]) and (TBLSP in [-inf, -inf, 0.000027, 0.000027]) and (LSP in [0.0254, 0.0255, inf, inf]) and (TBT in [-inf, -inf, 0.000056, 0.000057]) ⇒ Class=bot (CF = 0.89)
33	(VPL in [0.000744, 0.000913, inf, inf]) and (PLSP in [-inf, -inf, 0.03125, 0.035714]) and (LSP in [-inf, -inf, 0.055, 0.0686]) ⇒ Class=bot (CF = 1.0)
34	(RPD in [-inf, -inf, 0.001, 0.002]) and (VPL in [0.023281, 0.02549, inf, inf]) and (LSP in [-inf, -inf, 0.121, 0.1221]) ⇒ Class=bot (CF = 0.96)

Table 3: Fuzzy rules for detection of Waledac bot C & C traffic

Serial No	Rule
1	(APL in [-inf, -inf, 0.007491, 0.0075]) and (VPL in [0, 0.000001, inf, inf]) and (LSP in [-inf, -inf, 0.0062, 0.0064]) ⇒ Class=bot (CF = 1.0)
2	(LSP in [-inf, -inf, 0.0096, 0.0098]) and (LSP in [0.0094, 0.0096, inf, inf]) and (TPT in [0.0002, 0.0003, inf, inf]) ⇒ Class=bot (CF = 1.0)
3	(LSP in [-inf, -inf, 0.0062, 0.0065]) and (TBLSP in [0.000013, 0.000016, inf, inf]) and (RPD in [-inf, -inf, 0.002, 0.999]) and (LSP in [0.006, 0.0062, inf, inf]) ⇒ Class=bot (CF = 0.99)
4	(APL in [-inf, -inf, 0.005533, 0.006]) and (RTD in [0.01125, 0.01838, inf, inf]) and (TPT in [0.0002, 0.0003, inf, inf]) and (RPD in [-inf, -inf, 0, 0.001]) ⇒ Class=bot (CF = 1.0)
5	(APL in [-inf, -inf, 0.005533, 0.0059]) and (RTD in [0.00001, 0.00795, inf, inf]) and (RPD in [0, 0.001, inf, inf]) and (TBLSP in [-inf, -inf, 0.000011, 0.000012]) ⇒ Class=bot (CF = 0.98)
6	(APL in [-inf, -inf, 0.005533, 0.0058]) and (TPT in [0.0002, 0.0003, inf, inf]) and (RPD in [-inf, -inf, 0, 0.002]) ⇒ Class=bot (CF = 1.0)
7	(APL in [-inf, -inf, 0.005933, 0.00605]) and (TBLSP in [-inf, -inf, 0.000006, 0.000006]) and (LSP in [0.0055, 0.0058, inf, inf]) ⇒ Class=bot (CF = 1.0)
8	(APL in [-inf, -inf, 0.005933, 0.005967]) and (RPD in [-inf, -inf, 0.001, 0.002]) and (VPL in [0.000003, 0.000003, inf, inf]) and (VIT in [-inf, -inf, 0.0156, 0.018262]) ⇒ Class=bot (CF = 0.97)

Table 4: Structural attribute values of fuzzy rule sets

	NFR	ANAR	NRB	PCC	NRCF
Nugache	25	3.04	13	99.845%	18
Waledac	19	2.89	08	99.8%	09
Zeus	80	4.1	34	99.57%	31

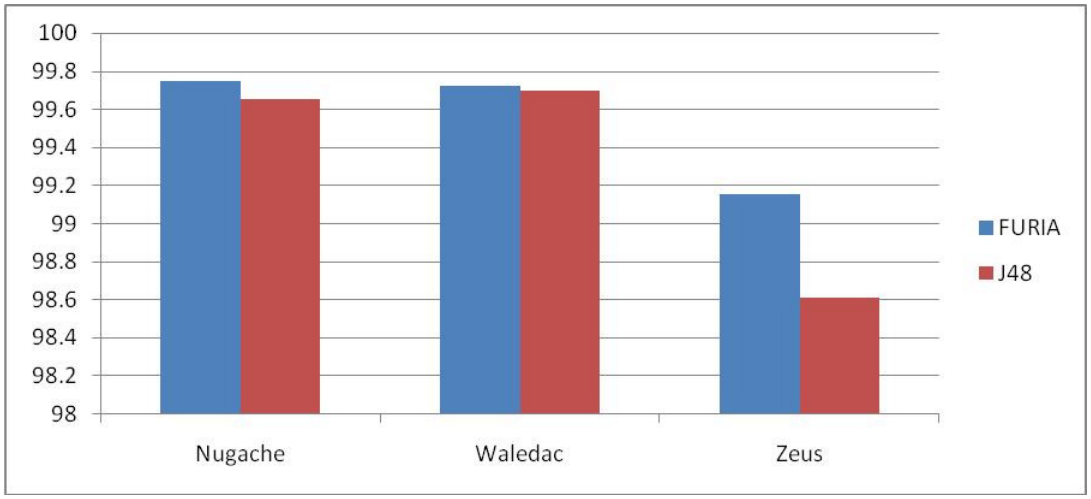


Figure 2: Percentage of accuracy

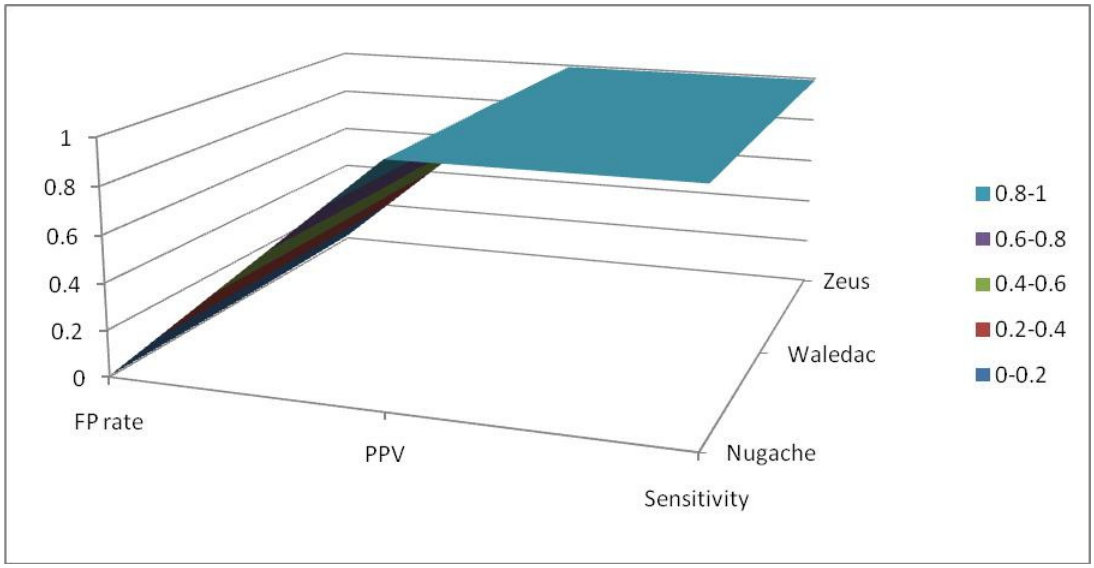


Figure 3: FP rate, PPV and sensitivity

of 0 to 1, as the performance measures are within this range only. Our fuzzy classifier produces the following results: (1) Sensitivity and PPV are 0.997 for both Nugache, Waledac traces, and 0.991 for Zeus. (2) FP rate is 0.005 for Nugache, 0.006 for Waledac and 0.017 for Zeus.

Table 5: A confusion matrix

	Predicted Class		
True class	-VE	+VE	
-VE	TN	FP	CN
+VE	FN	TP	CP
	RN	RP	N

Sensitivity, PPV and FP rate are inferior for Zeus sample dataset compared to that of Nugache and Waledac. From our analysis of sample datasets we find that Nugache and Waledac C & C flow samples are more distinguishable from normal traffic samples than the Zeus C & C flow sample. Following are the steps performed to do the analysis:

1) From the list of ten features in our feature set we proceed with two most influential pair of features i.e. Largest sized packet (LSP) and Proportion of largest sized packet (PLSP). Influence of a particular feature on classifiers performance has been judged through a simple performance-based input ranking methodology as has been described in our previous work [4].

2) We then removed the repeated values for this pair of features in all the datasets. After removal of duplicates we are left with only distinct values for each instance. The percentage of distinct combination obtained for Nugache is 0.313%, for Waledac it is 0.307%, for Zeus it is 5.887% and for Normal flow instances it is 28.3%.

3) We then calculated the percentage of distinct combinations having more than 1000 bytes in LSP for each dataset. We found that none of the packets in Nugache and Waledac datasets carry a payload of greater than or equal to 1000 bytes. For Zeus, the percentage of distinct combinations having more than 1000 bytes in LSP is 0.733% and for Normal the value is 7.64%.

From Step 2 we find that Zeus has a significantly higher percentage of distinct combinations compared to Nugache and Waledac. Similarly, from Step 3 we find that Zeus also has a good number of flows with LSP having more than 1000 bytes. Therefore, it is not difficult to ascertain that the classification error rate of Zeus is bound to be more compared to Nugache or Waledac.

The accuracy (the rate of correct classification) measure of a classifier is often used for comparison of predictive ability of learning algorithms. However, the accuracy measure completely ignores the probability estimations of the classification systems. Probability estimations generated by most classifiers can be used for ranking instances which gives likelihood estimations of instances and is therefore more desirable than just a classification. The AUC (area under the curve) of the ROC (Receiver Operating Characteristic) curve provides an alternative and better measure for machine learning algorithms by being invariant to the decision criterion selected, prior probabilities and is easily extendable to include cost/benefit analysis [6, 15]. ROC curve represents plotting of True Positive Rate against False Positive Rate as the decision threshold is varied, that can be used to compare the classifiers performance across the entire range of class distributions and error costs.

With a varied decision threshold and already obtained number of points on the ROC curve [FP rate = α, TP rate = 1 - β], the area under the ROC curve can be calculated by using the trapezoidal integration as follows:

$$AUC = \sum_i \{(1 - \beta_i . \triangle\alpha + \frac{1}{2}[\triangle(1-\beta).\triangle\alpha]\} \quad (6)$$

where, $\triangle(1-\beta) = (1-\beta_i) - (1-\beta_{i-1})$, $\triangle\alpha = \alpha_i - \alpha_{i-1}$.

In case of perfect predictions the AUC is 1 and if AUC is 0.5 the prediction is random. We provide a comparative analysis of our classification models using AUC values. Table 6 provides the AUC measures of our fuzzy based classification models and its corresponding values for decision tree based classification models. We find AUC measure for Zeus is significantly better in case of the fuzzy based classifier compared to the decision tree model, whereas for Nugache the fuzzy based classifier has a marginal edge over the one based on decision tree. The only exception is Waledac, where we have the AUC measure of decision tree classifier edge past the fuzzy classifier, though very marginally. Our explanation to this is that both these classifiers generates almost perfect classification models with equally good results for Waledac botnet C & C traffic sample, which is apparent from Figure 2 and Figure 3 having accuracy, sensitivity and PPV measures. AUC measure of a particular classification model is calculated through generation of a rank list based on probability estimations of instances. Thus it is not necessary that AUC measure of a classifier has to be higher compared to another classifier just because its other measures like accuracy, sensitivity, PPV etc. are on higher side. In fact, it implies that the error rate of decision tree based classifier generated from Waledac C & C traffic sample is slightly higher compared to fuzzy based classifier even though the decision tree classifier performs marginally better in terms of AUC measure. Nevertheless, from analysis of results we find that AUC measures of FURIA are much more consistent providing excellent predictions.

Table 6: AUC measures of fuzzy and decision tree based classification models

	FURIA	**C4.5**
Nugache	0.997	0.995
Waledac	0.997	0.998
Zeus	0.994	0.984

7 Conclusions

A fuzzy rule based detection framework for P2P botnets is presented here. The proposed approach leverages on flow level features and packet level features of network traffic to build excellent classification model for P2P botnet C & C traffic. The accuracy achieved by our system is as good as 99.745%, 99.715%, and 99.105% for Nugache, Waledac and Zeus botnet samples respectively. The fuzzy rule based approach is a supervised one and hence can detect known botnet traces only. P2P botnets have distributed C & C architecture and therefore complete annihilation of existing botnets is not easy. However, using our fuzzy rule based classification model, we can track botnet C & C traffic pro-actively as well as with high accuracy. In future, our effort will be to build similar detection model for botnets using other protocols and communication technologies such as social network based botnets, mobile botnets etc.

Acknowledgments

The authors would like to thank Mohammad M. Masud, Department of Computer Science, University of Texas at Dallas and Babak Rahbarinia, Department of Computer Science, University of Georgia, USA, for providing botnet traffic sample to carry out this research work. The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] B. Al-Duwairi and L. Al-Ebbini, "Botdigger: A fuzzy inference system for botnet detection," in *The Fifth International Conference on Internet and Web Applications and Services*, pp. 16–21, May 2010.

[2] D. Andriesse, C. Rossow, B. Stone-Gross, D. Plohmann, and H. Bos, "Highly resilient peer-to-peer botnets are here: An analysis of gameover zeus," in *Proceedings of the 8th IEEE International Conference on Malicious and Unwanted Software (MALWARE'13)*, pp. 116–123, Fajardo, Puerto Rico, USA, 2013.

[3] S. Balram and M. Wilscy, "User traffic profile for traffic reduction and effective bot C & C detection," *International Journal of Network Security*, vol. 16, no. 1, pp. 46–52, 2014.

[4] P. Barthakur, M. Dahal, and M. K. Ghose, "A framework for P2P botnet detection using SVM," in *4th International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery*, pp. 195–200, Oct. 2012.

[5] P. Barthakur, M. Dahal, and M. K. Ghose, "An efficient machine learning based classification scheme for detecting distributed command & control traffic of P2P botnets," *International Journal of Modern Education and Computer Science*, vol. 5, no. 10, pp. 9–18, 2013.

[6] A. P. Bradley, "The use of the area under the roc curve in the evaluation of machine learning algorithms," *Pattern Recognition*, vol. 30, no. 7, pp. 1145–1159, 1997.

[7] W. Cohen, "Fast effective rule induction," in *12th International Conference on Machine Learning*, pp. 115–123, 1995.

[8] R. Dimov, "Weka: Practical machine learning tools and techniques with Java implementations," in *AI Tools Seminar University of Saarland*, June 2007. (http://researchcommons.waikato.ac.nz/bitstream/handle/10289/1040/uow-cs-wp-1999-11.pdf?sequence=1&isAllowed=y)

[9] F. Geramiraz, A. S. Memaripour, and M. Abbaspour, "Adaptive anomaly-based intrusion detection system using fuzzy controller," *International Journal of Network Security*, vol. 14, no. 6, pp. 352–361, 2012.

[10] G. Gu, R. Perdisci, J. Zhang, and W. Lee, "Botminer: Clustering analysis of network traffic for protocol- and structure-independent botnet detection," in *17th USENIX Security Symposium*, pp. 139–154, Dec. 2008.

[11] G. Gu, V. Yegneswaran, P. Porras, J. Stoll, and W. Lee, "Active botnet probing to identify obscure command and control channels," in *Annual Computer Security Applications Conference*, pp. 241–253, 2009.

[12] J. Huhn and E. Hullermeier, "FURIA: An algorithm for unordered fuzzy rule induction," *Data Mining and Knowledge Discovery*, vol. 19, no. 3, pp. 293–319, 2009.

[13] U. Lamping and E. Warnicke, *Wireshark Users Guide*, Wireshark Foundation, 2008.

[14] Y. D. Lin, Y. T. Chiang, Y. S. Wu, and Y. C. Lai, "Automatic analysis and classification of obfuscated bot binaries," *International Journal of Network Security*, vol. 16, no. 6, pp. 477–486, 2014.

[15] C. X. Ling, J. Huang, and H. Zhang, "AUC: A better measure than accuracy in comparing learning algorithms," in *16th Canadian Conference on Artificial Intelligence*, pp. 329–341, June 2003.

[16] C. Y. Liu, C. H. Peng, and I. C. Lin, "A survey of botnet architecture and botnet detection techniques," *International Journal of Network Security*, vol. 16, no. 2, pp. 81–89, 2014.

[17] Y. Lu, Ye Zhu, M. Itomlenskis, S. Vyaghri, and H. Fu, "MMOPRG bot detection based on traffic analysis," *International Journal of Electronics and Information Engineering*, vol. 2, no. 1, pp. 18–26, 2015.

[18] M. Mahmoud, M. Nir, and A. Matrawy, "A survey on botnet architectures, detection and defences," *International Journal of Network Security*, vol. 17, no. 3, pp. 272–289, 2015.

[19] M. M. Masud, J. Gao, L. Khan, and B. Thuraisingham, "A multi-partition multi-chunk ensemble technique to classify concept-drifting data streams," in *13th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining*, pp. 363–375, Apr. 2009.

[20] M. Overton, "Rootkits: Risks, issues and prevention," in *The 2006 Virus Bulletin Conference*, 2006.

[21] J. R. Quinlan, *C4.5: Programs for Machine Learning*, San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1993.

[22] B. Rahbarinia, R. Perdisci, A. Lanzi, and K. Li, "Peerrush: Mining for unwanted P2P traffic," *Journal of Information Security and Applications*, vol. 19, no. 3, pp. 194–208, July 2014.

[23] R. S. Roshna and V. Ewards, "Botnet detection using adaptive neuro fuzzy inference system," *International Journal of Engineering Research and Applications*, vol. 3, no. 2, pp. 1440–1445, 2013.

[24] C. A. Schiller, J. Binkley, D. Harley, G. Evron, T. Bradley, C. Willems, and M. Cross, *Botnets the Killer Web APP*, United States: Syngress Publishing Inc., 2007.

[25] G. Sinclair, C. Nunnery, B. Byung, and H. Kang, "The waledac protocol: The how and why," in *Proceedings of 4th International Conference on Malicious and Unwanted Software (MALWARE'09)*, pp. 69–77, Feb. 2010.

[26] S. Stover, D. Dittrich, J. Hernandez, and S. Dietrich, "Analysis of the storm and nugache trojans: P2P is here," *USENIX*, vol. 32, no. 6, pp. 18–27, 2007.

[27] K. Thomas and D. Nicol, "The koobface botnet and the rise of social malware," in *5th International Conference On Malicious and Unwanted Software (Malware'10)*, 2010. (https://www.ideals.illinois.edu/bitstream/handle/2142/16598/malware2010.pdf?sequence=2)

[28] K. Wang, C. Y. Huang, S. J. Lin, and Y. D. Lin, "A fuzzy pattern-based filtering algorithm for botnet detection," *Computer Networks*, vol. 55, no. 15, pp. 3275–3286, 2011.

Pijush Barthakur received the Master of Computer Application (MCA) degree from Dibrugarh University, India in 2001. Currently, he is working as Associate Professor at Department of Computer Applications, Sikkim Manipal Institute of Technology, Sikkim, India. He is also pursuing his doctoral degree in Sikkim Manipal University. His research interests are in the area of Network Security. He served as a member of Technical Program Committee at 5th International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery, Beijing, China, 2013.

Manoj Dahal received his Ph.D degree on Networking from Tezpur University, India in 2008 for the thesis on Addressing Transport Layer Congestion Control Issues. Currently, he is working at Novell, India and his professional work mostly lie on File Access protocol areas. He is also associated with research on Detection of Botnets using Machine Learning Techniques at Sikkim Manipal Institute of Technology, Sikkim, India. He has around 15 years of experience in software industry. He was a Post-Doctoral Fellow for about a year with INRIA, France at LIP Labs, ENS de Lyon, where he has worked on Traffic Engineering for Optical Networks. He also worked as a Professor for a short period in the Department of Computer Science and Engineering, at

Sikkim Manipal Institute of Technology, Sikkim. He also worked with Nokia (via Satyam) on routing devices and National Informatics Centre on e-Governance Projects in India.

M. K. Ghose is currently the Dean (Academics), SMIT and Professor, Department of Computer Science & Engineering at Sikkim Manipal Institute of Technology, Majitar, Sikkim, India. Formerly he was Dean (R&D) since June, 2006. During June 2008 to June 2010, he had also carried out additional responsibilities of Head, SMU-IT. Prior to this, he worked in the internationally reputed R & D organization ISRO during 1981 to 1994 at Vikram Sarabhai Space Centre, ISRO, Trivandrum in the areas of Mission simulation and Quality & Reliability Analysis of ISRO Launch vehicles and Satellite systems and during 1995 to 2006 at Regional Remote Sensing Service Centre, ISRO, IIT Campus, Kharagpur in the areas of RS & GIS techniques for the natural resources management. He was also associated with Regional Engg. College (NIT), Silchar (1979-1981) as Teaching Asst. and Assam Central University, Silchar as COE and HOD of Computer Science Department (1997-2000). His areas of research interest are Data Mining, Simulation & Modelling, Network, Sensor Network, Information Security, Optimization & Genetic Algorithm, Digital Image processing, Remote Sensing & GIS and Software Engineering. He chaired a number of national/international conference sessions. He has conducted quite a number of Seminars, Workshop and Training programmes in the above areas and published 126 technical papers in various national and international journals in addition to presentation/publication in several international/national conferences. Till date, he has produced 8 Ph.Ds and research assistance given for 2 Ph.Ds. Presently 11 scholars are pursuing Ph.D work under his guidance. Dr. Ghose is having 8 sponsored projects worth of 1 crore (INR). Dr Ghose also served as technical consultant to various reputed organizations like IIT Chennai, IIT Kharagpur, WRI, Tricy, SCIMST, KELTRON, HLL, Trivandrum. Dr. Ghose can also be reached at mkghosesmu.edu,in; headcse.smit@gmail.com.

Analysing and Improving Performance and Security of Cryptographically Generated Address Algorithm for Mobile IPv6 Networks

Sana Qadir, Mohammad Umar Siddiqi, and Wajdi F. M. Al-Khateeb
(Corresponding author: Sana Qadir)

Electrical and Computer Engineering Department, International Islamic University Malaysia
P.O. Box 10, 50728 Kuala Lumpur, Malaysia
(Received Nov. 20, 2014; revised and accepted Feb. 20 & Apr. 21, 2015)

Abstract

A Cryptographically Generated Address (CGA) is a self-certifying address that a node generates when it joins a foreign network. Despite its advantages, generating a CGA is computationally expensive. This study examines the security and performance issues related to the use of the CGA Generation algorithm. It also scrutinizes the hash extension mechanism, different hash functions and how multithreading can be used to improve the performance of the CGA Generation algorithm. Based on the results, this research recommends imposing a minimal computational security of $\mathcal{O}(2^{80})$, the use of the HAVAL hash function and parallelizing the algorithm in order to take maximum advantage of multicore architectures of mobile node.

Keywords: CGA generation algorithm, hash functions, multithreading, parallel computing

1 Introduction

A Cryptographically Generated Address (CGA) is an IPv6 address generated by a node using the CGA Generation algorithm as defined in RFC 3972. The input to this algorithm is the public key of the node and some auxiliary parameters. The output of the algorithm is a CGA.

CGAs were introduced in IPv6 as part of *stateless address auto configuration* (SLAAC). This enables nodes to join a subnet and locally generate an IPv6 address. Although CGAs have several advantages, their main shortcoming is high computational cost. The aim of this paper is to carry out an in-depth analysis of the security and performance of the CGA Generation algorithm.

This is important for several reasons. Firstly, Mobile IPv6 (MIPv6) networks usually consist of low-end nodes that have limited resources (computational, memory, bandwidth, power, etc.) and therefore cannot be expected to perform computationally expensive operations. Secondly, CGAs are increasingly being included in protocols like Enhanced Route Optimization - ERO (where they are used to prove ownership of a MN's home address). Proving ownership of an address is important to protect against attacks such as address stealing, flooding, session hijacking and redirect attacks [13, 30]. One of the factors that dominates the cost of CGA-based authentication protocols is the CGA Generation algorithm [12, 16]. In the case of MIPv6 networks, delays have to be minimised to preserve the quality of real-time and interactive applications. In practice, this means operations like handovers should be completed within a few hundred milliseconds.

2 Related Work

Essentially, a CGA cryptographically binds the public key of a node to its IPv6 address. The details of the CGA Generation algorithm are illustrated in Figure 1. The CGA Parameters data structure that the sending node shares with the receiving node is shown in Figure 2. The receiving node verifies a CGA using the CGA Verification algorithm. This study will only focus on the CGA Generation algorithm and not the CGA Verification algorithm.

CGAs require the sending and the receiving node to share a 3-bit integer called `sec` that indicates the security level of the CGA against brute force attack. `sec` can take values from 0 (lowest security) to 7 (highest security) and is encoded in the three leftmost bits of the generated interface identifier (IID).

The main aim of a CGA is to prevent the stealing and spoofing of existing IPv6 addresses [4]. In other words, an impersonation attack - given a CGA, an adversary is able to find another public key that generates the same CGA. This would require the adversary to break the 2^{nd} pre-image resistance of `hash1` [4]. Because only 59 bits of `hash1` make up the IID, the cost of finding a hash collision is only $\mathcal{O}(2^{59})$. 59 bits are too few to provide

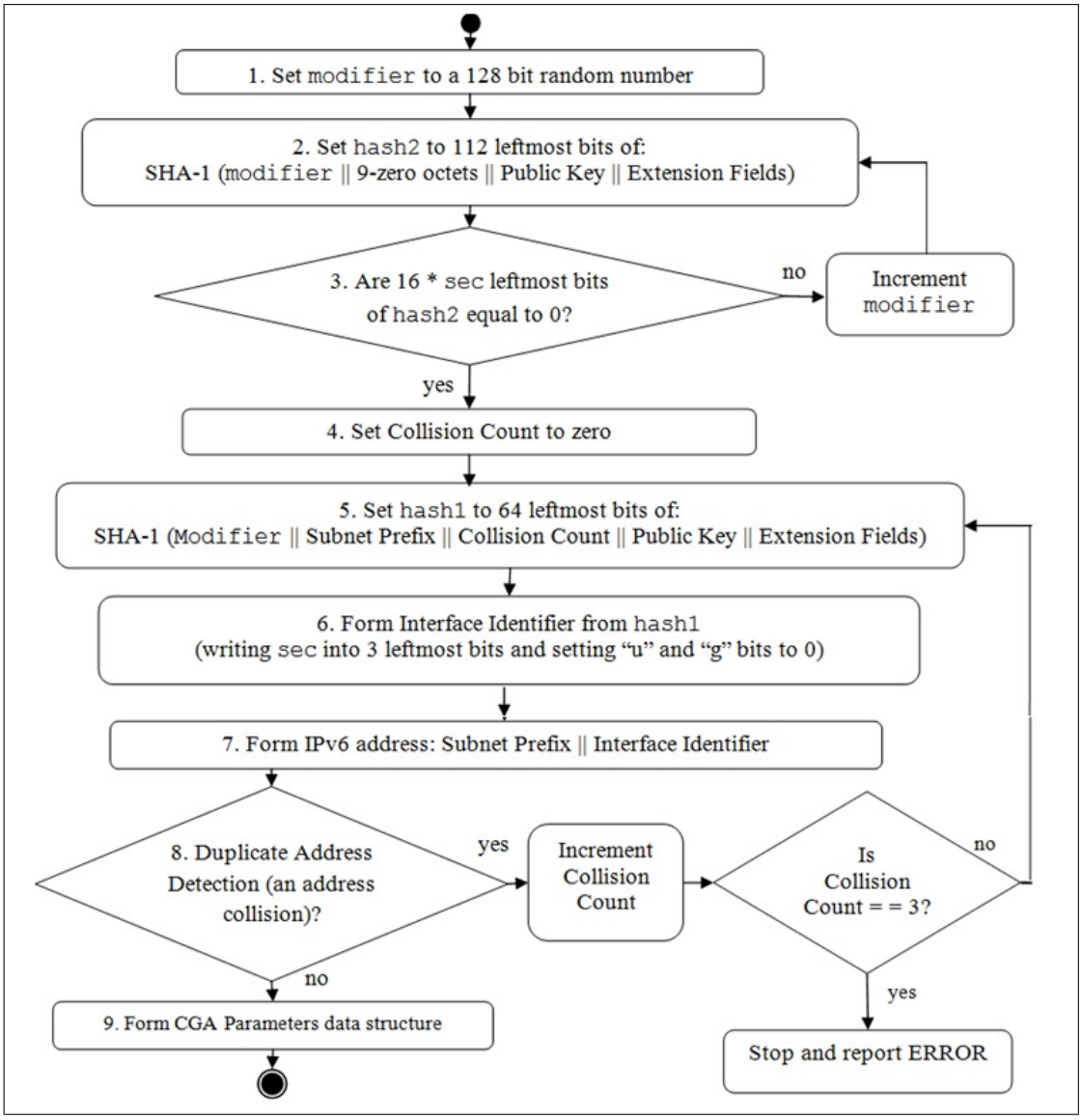


Figure 1: CGA generation algorithm [4]

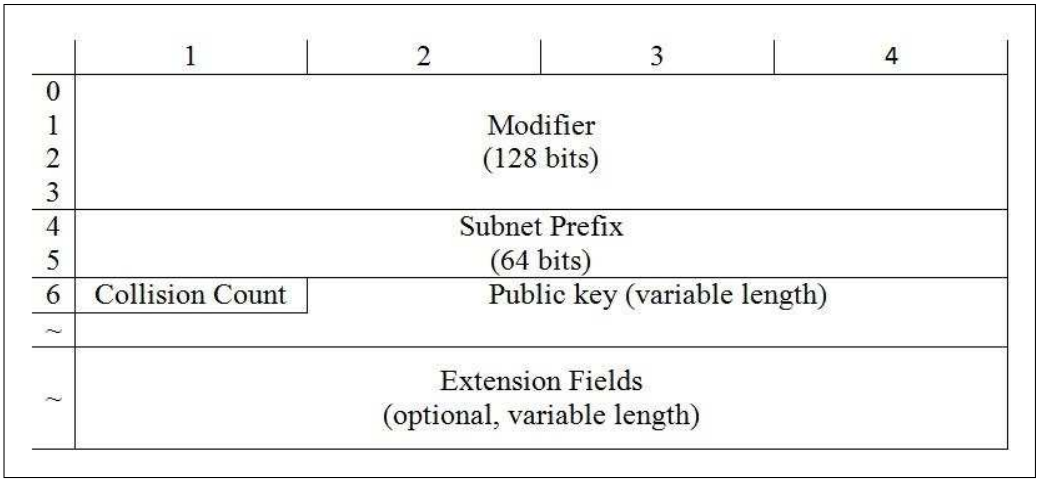


Figure 2: CGA parameters data structure [4]

strong security or any real protection against brute force attacks. CGAs should provide users with the option to increase this cost in the face of exponential growth of computational capacity and memory.

The *hash extension mechanism* was introduced to solve this shortcoming. This mechanism modifies the input to `hash2` until the leftmost $16*\texttt{sec}$ bits of the hash digest are zero [4]. This effectively increases the cost of an impersonation attack to $\mathcal{O}(2^{59+\texttt{sec}*16})$. However, this mechanism has the negative impact of increasing the cost of generating a CGA to $\mathcal{O}(2^{\texttt{sec}*16})$ [4]. In fact, several studies confirm that the largest contributor to the computational cost of the CGA Generation algorithm is the value of `sec`.

2.1 Performance Analysis

Table 1 summarizes the results from studies that have undertaken the performance evaluation of the CGA Generation algorithm. It is obvious to see that the performance of the CGA generation algorithm degrades substantially with increasing `sec` values. It is also important to note that for `sec` values greater than 0, the CGA Generation algorithm is not guaranteed to terminate.

RFC 3972 stipulates that nodes can choose `sec` value based on [4]:

- How long they expect to use the address;
- Their computational capacity;
- The perceived probability of being attacked.

The RFC also stipulates several solutions that can be used to overcome poor performance of the CGA Generation algorithm [4]:

- Using small `sec` values;
- Offloading computationally costly Steps 1-3 to a more powerful machine; or
- Completing the computationally costly Steps 1-3 offline or in advance.

Existing literature contains a number of studies that have investigated the factors that impact the performance of the CGA Generation algorithm. It also contains a number of possible solutions to improve the performance of the CGA Generation algorithm. These studies are summarized in Table 2.

2.1.1 Hash Extension Mechanism

Because it is vital that MIPv6 nodes complete address generation in less than a few hundred milliseconds, one obvious solution to improve the performance of the hash extension mechanism is through some form of time based termination. This was initially proposed in [5] and later refined in [21] as Time-Based CGA (TB-CGA). In TB-CGA, `sec` is not selected by the node. Instead the node decides the time after which CGA Generation algorithm must terminate. The best `hash2` value found during this time (i.e. the `hash2` value where the most $\texttt{sec}*8$ leftmost bits are zeros) is used to generate the CGA. Essentially, `sec` is automatically determined based on the time. A faster CPU will search for more `hash2` values within the same time meaning TB-CGA will automatically adjust `sec` according to the speed of the processor on which it is run [21]. This is a very advantageous design because it automatically adjusts based on the resources of the node.

Despite these advantages, the authors feel that users can be negligent and set an address generation time that is very small. This can result in the generation of an address that is detrimental to the security of the whole network. Also, using $\texttt{sec} * 8$ instead of $\texttt{sec} * 16$ was proposed as a good idea from a performance perspective in [5]. Reference [21] provides empirical evidence to support this claim especially in the case of low-end nodes. This approach also does not change the communication of `sec` to the verifying node nor does it change the CGA Verification algorithm.

2.1.2 Generation of Key Pair

For improving key generation time, the best solution is to use alternative cryptosystems. The best example is provided in [9]. This study reports that CGA Generation time using RSA-1024 (4.70 s) drops 31 times for ECC-163 (0.15 s). However, the choice of which public key cryptosystem is best to use in the CGA Generation algorithm is out of the scope of this paper. One reason for this is that the choice cannot be made solely on the performance of the algorithm used to generate the key pair. The performance of the CGA Signature generation and CGA Signature verification algorithms must also be taken into account as is investigated in [27].

2.1.3 Hash Function

The performance of the hash function is of importance because of the role it plays in the hash extension mechanism where Steps 2 - 3 of the algorithm are repeated in search of a suitable `modifier`. To this end, [14] replaces SHA-1 with MD-5 for use in Mobile Ad-Hoc Networks (MANETS). This is because of the latter's simplicity and superior performance. In [23], the CGA Parameters data structure is restructured and then some operations in SHA-1 and MD-5 are reordered to take advantage of this new structure. They report an 80% improvement in performance [23]. However, it must be noted that both SHA-1 and MD-5 are considered broken. An attack on the collision resistance property of SHA-1 can be carried out in $\mathcal{O}(2^{63})$ instead of $\mathcal{O}(2^{80})$ [19]. The authors are of the opinion that using weak hash functions to improve the performance of the CGA Generation algorithm is not an acceptable approach.

Table 1: Performance of CGA generation algorithm (RSA-1024)

Source	Setup	sec	Sample Size	Performance	Recommendation
[9]	Nokia 800	0	10000	4.7 s	Use sec = 0 for mobile nodes
[1]	Intel Duo 2.67 GHz CPU	0	1000	avg: 93.41 ms	Do not use sec value more than 1
		1	1000	avg: 402 ms	
		2	5	avg: 1 hr 39 min	
[17]	AMD64 with OpenSSL	1	–	0.2 s	Use sec = 1
		2	–	3.2 hr	
[22]	–	2	–	avg: several hours	Users should use sec values of 0 or 1

Table 2: Factors affecting performance of CGA generation algorithm and possible improvements

Source of computational cost	Aim of mechanism (importance to security)	Proposed solutions	Source(s)	Disadvantage
Hash extension mechanism	Increase security level of CGA against brute force attack (only 59 bits of hash digest are used as the interface identifier)	Users use small sec values (0 or 1)	[4, 5]	Computational security $< \mathcal{O}(2^{80})$
		Steps 1-3 can be done on a powerful machine beforehand	[3, 4]	Relies on a centralized model
		Time limit based on application or CPU speed	[21]	
		Time and probability based termination condition	[5]	
		Use cryptographic/graphic accelerator cards. Significant reduction in CGA generation time esp. for higher sec values	[9]	
		Take advantage of parallelism to speedup CGA generation particularly on devices with multiple cores	[2]	
Generation of key pair	The key pair is used in the generation and verification of CGA Signatures	Delegate to a more powerful key server to generate key pair	[3, 4, 29]	Relies on a centralized model
		Use public key cryptosystem with faster key generation time (e.g. ECC)	[9, 10]	
Hash function	Generate hash digest	Replace SHA-1 with an alternative faster hash function (e.g. MD-5)	[9, 14]	MD-5 is broken

2.2 Security Analysis

On a broader note, the performance of CGA Generation algorithm cannot be scrutinized without analyzing the security issues surrounding the use of CGAs (see Table 3). Using CGAs can still leave a network vulnerable to a few types of attacks. The attacks possible against the CGA Generation algorithm are discussed in this section.

2.2.1 Global Time-Memory Trade-Off (TMTO) Attack

This attack is explained in [17]. [17] also proposes an improved CGA algorithm called CGA++ to help prevent this attack. CGA++ protects against replay attack but at the cost of an additional signature generation and signature verification operation. [10] improves CGA++ by proposing the use of faster ECDSA signatures in their Compact and Secure CGA (CS-CGA). They also show that CS-CGA Generation algorithm (with ECC P-256) takes 1.96 s while the original CGA Generation algorithm (with RSA-3072) takes 2.183 s. Using ECC, also has the advantage of generating shorter signatures and smaller CGA Parameters data structures. However, the CS-CGA Verification algorithm (with ECC P-256) is 0.037 ms slower than the original CGA Verification algorithm (with RSA 3072). Despite the benefit of using ECC, the CS-CGA Generation algorithm is still computationally expensive. Moreover, [17] notes that the TMTO attack is prohibitive in the terms of the amount of storage required to launch the attack. Impersonating a random node in a network with 2^{16} nodes would require about 128 TB of storage [17].

2.2.2 Impersonation

The security of a CGA is also affected by the hash function used. Protection against impersonation requires a hash function that it is 2^{nd} pre-image resistant. The hash function must also be very efficient because it is repeatedly used in the computationally intensive Steps 2 - 3. Replacing SHA-1 with a more secure hash function was investigated in [9]. They found that SHA-1 outperforms most other commonly accepted hash functions like SHA-256 and SHA-512 [9]. One more study has also compared the performance of hash functions and found that SHA-256 performs better than BLAKE, Skein and SHA-3 (Keccak) [27]. BLAKE and Skein were included in the study for several reasons. Firstly, BLAKE has a simple design that is easy to implement and lends itself to excellent performance [15]. Skein is flexible, simple and also shows excellent performance on both hardware and software (including a version called Skein-256 that can be implemented on 8-bit smart cards) [26]. Lastly, SHA-3 was chosen because it is based on a sponge construction that is completely different from the Merkle-Damgard construction used in many commonly used hash functions (like SHA-1 and MD-5). The sponge construction is an iterative structure that supports variable length output and in addition to the basic security properties of hash functions, it has been proven to be indifferentiable from the random oracle [11]. There are a few disadvantages to hash functions based on the sponge construction. The most notable is the large state. This basically makes hash functions like Keccak more suitable for large messages and not small ones like in the context of the CGA Generation algorithm. However, we will include Keccak in this study because of its adoption as SHA-3.

We will not go into detail about the DoS attacks against the CGA Verification algorithm. The focus of this paper is the CGA Generation algorithm.

Also, we agree with the use of a timestamp option (in the CGA Parameters data structure) to protect against replay attacks.

We will also not go into further details about the privacy issue surrounding the use of CGAs and the garbage attack (as outline in Table 3).

3 Design of Enhanced CGA Generation Algorithm

3.1 Hash Extension Mechanism

The hash extension mechanism was proposed by [5] as a solution for applications where the hash digest was limited to less than 128 bits. Hash values longer than or equal to 128 bits are considered secure against brute force attacks for any reasonable future while a minimum of 80 bits are acceptable for the immediate future [5]. This is particularly important in scenarios where the adversary has a much more powerful computer while the victims node is a low-end mobile or embedded computer.

We think that the design of the enhanced CGA Generation algorithm should:

- Impose a minimal computational security. Users can be negligent and set an address generation time that is very small. This can be detrimental to the security of the whole network. There should be a minimal security level that a node must provide, i.e. $\mathcal{O}(2^{minimal})$. A reasonable value is 80 bits given the computational capacity of modern nodes. In future, this can be increased for nodes with greater computational capacity.

- Allow the value of `sec` to be guided by the three factors mentioned in RFC 3972:

  1) the duration a node is expected to use an address, i.e. $\mathbf{T}_{expected\_lifetime}$. Nodes frequently move from one subnet to another. It is a waste of resources to generate a CGA with high computational security when the user has no intention of staying in the subnet for any reasonable duration.

Table 3: Limitations of CGAs from a security perspective

Name of attack	**Algorithm / Data**	**Details of attack**	**Mitigation or counter mechanisms**
Denial of Service (DoS) against CGA Verification process	CGA Verification Algorithm	An adversary can reply to each DAD check performed by a node on a tentative CGA telling the node that the address is already in use. Effectively this prevents the node from joining the subnet.	• Sign DAD & NA messages [4]; • Verify each DAD response [1]; • Use DAD extension [22].
	CGA Parameters data structure	Adversary captures/sniffs, replays or changes the sender' CGA parameters so the verification process fails.	Use a *Timestamp Option* when CGA is used in protocols other than SeND [22].
Global Time-Memory Trade-off (TMTO) Attack	CGA Generation Algorithm	The adversary creates a large database of IIDs from its own key pair and then searches for matches for many addresses. • Attack can be assumed to be almost impractical because of massive storage requirements.	• Include subnet prefix in input to `hash2`. This forces adversary to create a separate database for each subnet prefix [3]. • CGA++ (also sign input to `hash1`; expensive and does not solve problem with local-link addresses). This prevents TMTO attack from being applied globally [10, 17, 22].
Garbage Attack	CGA	The adversary uses random data as public-key. • Limited practicality since node does not have corresponding private key.	• Include an authentication mechanism in CGA or use CGA in a protocol that demands authentication [17].
Impersonate an existing CGA	CGA Generation Algorithm	Find another key pair that produces the same CGA. • Break 2^{nd} pre-image resistance of SHA-1(`hash1`). • Cost of attack: $\mathcal{O}(2^{59+\texttt{sec}*16})$.	Replace SHA-1 with SHA-256 (see RFC 4982) [10, 27].
Violation of Privacy	CGA	A node that continues to use a valid CGA (in a subnet) for a long period of time can be tracked.	Set a lifetime for a CGA address [22]: $mT_G \leq T_l \leq T_A/n$ where T_G is time to generate a new CGA, T_l is the lifetime of a CGA, T_A is time to attack a CGA, m and n are integers.
		An adversary can track a node using its public key. • Difficult attack to carry out because nodes are usually tracked using their IP address.	• Generate a new key pair when joining a new network.

2) the perceived probability of an attack , i.e. $\mathbf{P}_{attack}$. This can be set to a high value when a user is joining an untrusted/public network or low when joining a secure/protected network.

3) the computational capacity of a node, i.e. $\mathbf{CPU}_{capacity}$.

Values that can be selected by the user for each of these three factors are shown in Table 4. In this way, the final value of `sec` remains between 0 and 7 and can be securely encoded in the three leftmost bits of the CGA.

- support a maximum computational security of more than 128 bits. This is to ensure that CGAs are applicable well until 2030.
- granularity of 8 (as in TB-CGA) instead of 16 (as in RFC 3972). The option of removing the granularity altogether is very attractive because then the `hash2` value with the most zero leftmost bits found in a given time can be used. However, this strategy is not possible because only three bits are available to securely transmit `sec`.

If all of the above mentioned design changes are adopted, then the overall computational security of the CGA can be calculated as in Equation (1):

$$\begin{aligned} & ComputationalSecurity \\ = \; & O(2^{(\mathbf{T}_{expected\_lifetime}+\mathbf{P}_{attack}+\mathbf{CPU}_{capacity})*8+80)}. \end{aligned} \tag{1}$$

The authors recognize that the above design depends on how accurately a user chooses values for $\mathbf{T}_{expected\_lifetime}$, $\mathbf{P}_{attack}$ and $\mathbf{CPU}_{capacity}$. However, the range of computational security (from $\mathcal{O}(2^{80})$ to $\mathcal{O}(2^{136})$) is optimal.

3.2 Hash Function

Hash functions are usually not considered to be a performance bottleneck specially on desktops. However, on embedded systems (with slower bandwidth), the performance of the hash function can have a more substantial impact (esp. when the hash function is executed in a loop as in Steps 2 - 3).

SHA-1 is used in the original CGA Generation algorithm because of its efficiency. Any hash function that replaces SHA-1 must have superior or comparable performance.

SHA-3 and Skein have been around for a few years, so this study will include them for comparison purposes. This study will also include the new improved version of BLAKE called BLAKE2 which is reported to have comparable performance to MD5 on 64-bit platforms. BLAKE2 comes in two versions. BLAKE2b is optimized for 64-bit architectures and BLAKE2s is optimized for 8-bit or 32-bit architectures [7].

This study will also examine two other hash functions that are not broken and produce hash digests of at least 128 bits. The first hash function is HAVAL. This hash function is based on the Davies-Meyer construction and is not susceptible to attacks that aim to exploit the Merkle-Damgard construction. The downside to this function is that an efficient algorithm, with a complexity of $\mathcal{O}(2^{59})$, has been demonstrated for constructing collisions for the 3-pass version of HAVAL [8]. As such, only the 4-pass and 5-pass versions of HAVAL, for which no weaknesses have been found, are considered secure. Also, HAVAL is reported to be faster than MD5. The last hash function included in this study is MD6 [24]. The current MD6 version is resistant to the buffer overflow error and has been proven to be resistant to differential cryptanalysis. Its design takes full advantage of opportunities for parallelism in multicore architectures. It is also considered to be a relatively simple and efficient hash function [6].

3.3 Timestamp

This is included as an Extension Field in the CGA Parameters data structure to protect against replay attacks. Figure 3 shows the Enhanced CGA Parameters data structure.

3.4 Include Subnet Prefix in Input to `hash2`

This is included to protect against the Global Time-Memory Trade-off (TMTO) attack.

3.5 Parallelism

One method of reducing the cost of the CGA Generation algorithm is to take advantage of the multicore architecture of most recent mobile nodes. Almost all platforms are becoming multicore, as manufacturers have realised that improving performance by increasing raw clock rates is reaching its physical limit and mutlicore chip design is the best approach to adopt. For example, the Qualcomm Snapdragon 808 (arrived at end of 2014) has six cores (a dual core Cortex A57 and four Cortex A53) [28].

Multicore systems have the most impact on performance when the main processing of an algorithm is split into multiple threads. In other words, when the algorithm is parallelized. However, it should be remembered, that the maximum speedup in performance is limited by Amdahl's law. Essentially, this law states that the speedup obtained from multiple processors is limited by the execution time of the sequential part of a program [25].

At first glance, the CGA Generation algorithm looks like a sequential set of instructions. But there are two obvious ways in which the computationally expensive parts of the algorithm can be parallelized. It is important to remember that the way an algorithm is parallelized has an impact on its performance. Two methods are illustrated in Figure 4 and Figure 5. In these examples, the main

Table 4: Values for three factors that determine overall value of `sec`

$T_{expected\_lifetime}$	P_{attack}	$CPU_{capacity}$	value
Small	*Negligible*	*Low*	0
Medium	*Low*	*Average*	1
Large	*Medium*	*Fast*	2
-	*High*	-	3

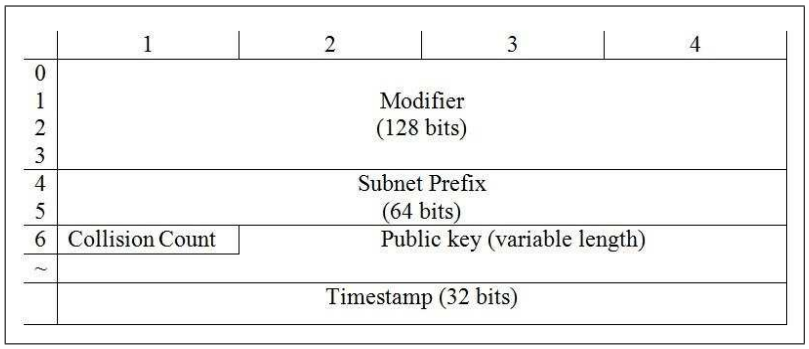


Figure 3: Enhanced CGA parameters data structure [5]

process spawns two additional threads (i.e. $t = 2$). More threads can be spawned if additional cores are available (e.g. four threads $t = 4$ when four cores are available).

Theoretically, assuming:

- T_i is the time taken by Step i that is executed by a thread in parallel;
- T_S is the total time taken by all the sequential steps; and
- c is the number of cores.

Each method can be analyzed in the following ways.

Method 1. *Each thread starts with a different random* `modifier`*:*

$$T_{CGA} \approx \min\left(\sum_1^{m_1}\sum_{i=1}^{3} T_i, ..., \sum_1^{m_t}\sum_{i=1}^{3} T_i\right) + T_S \quad (2)$$

Here, m_1 *is the number of* `modifiers` *searched by thread* 1*,* m_2 *is the number of* `modifiers` *searched by thread* 2 *and so on until* m_t *(i.e. number of* `modifiers` *searched by thread* t*).*

Method 2. t *threads equally share the number of* `modifiers` *to be searched, i.e.* m_{Total}*:*

$$T_{CGA} \approx \frac{m_{Total}}{t}\left(\sum_{i=2}^{3} T_i\right) + T_S \quad (3)$$

This study implements and reports results from both these methods.

4 Implementation of Enhanced CGA Generation Algorithm

4.1 Hash Function and Hash Extension Mechanism

The enhanced CGA Generation algorithm is implemented in C. The Meamo 5 SDK is used and the code cross-compiled for ARM architecture. Also, every effort is made to use the same library or implementation (e.g. of hash function) in order to ensure that performance indicates difference in design rather than difference in implementation [20]. The SAPHIR library (for SHA-2, SHA-3, Skein, HAVAL) and reference C implementations are used (e.g. `blake2_code_20140114.zip` from [7] and `md6_c_code-2009-04-15.zip` from [24]). The clock cycles are recorded for the following operations on an actual mobile architecture (i.e. a Nokia 900):

1) Calculate `hash2`;
2) CGA Generation algorithm.

It should be noted that the Nokia 900 has TI OMAP 3430 chipset with a 600 MHz Cortex-A8 CPU. It also has a PowerVR SGX530 GPU.

4.2 Parallelism

To implement parallelism, POSIX threads (or `Pthreads`) are used. `Pthreads` have a much lower overhead (at least 6 times faster) compared to `fork()`. Apart from basic multithreading, mutexes and condition variables are used to implement Methods 1 and 2 [18].

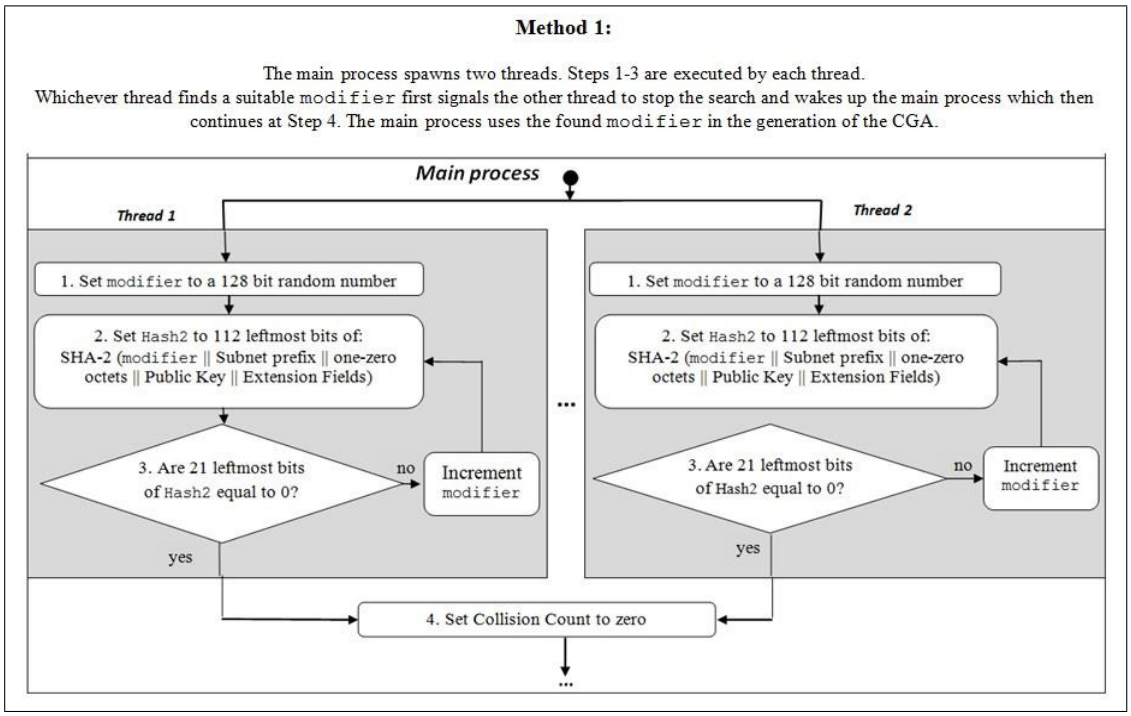


Figure 4: Method 1

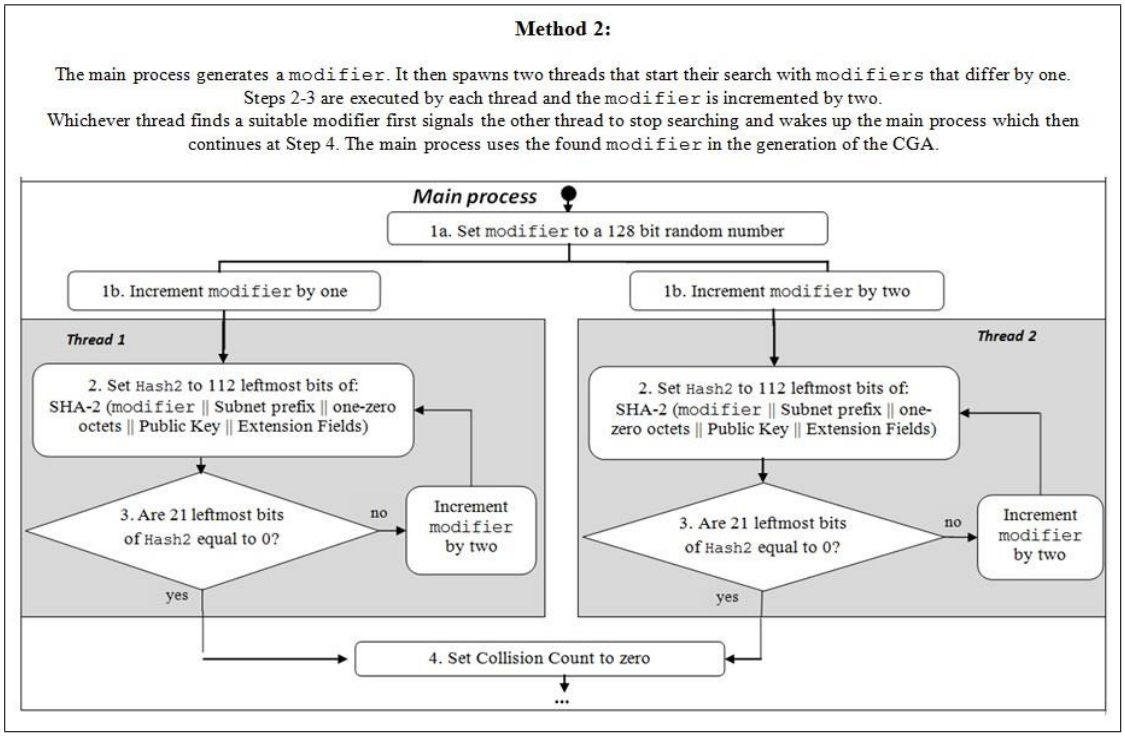


Figure 5: Method 2

The CGA Generation algorithm (implemented using Methods 1 and 2) is run on an Intel Core i7-3537U CPU @ 2.00 GHz (cache size: 4096 KB). This architecture has two cores with each core clocked at 2.0GHz. With hyper-threading, the two cores are capable of handling up to four parallel threads. In other words, the architecture acts as if it has four cores. This provides reasonable estimation since as of 2014 most Android smartphones are quad-cores processors. It is also important to note that only when `pthread_setaffinity_np()` is used to allocate a thread to run on a specific core, the utilization of the core reach 100%. The Gnome/GNU Linux `system monitor` is used to observe CPU utilization.

5 Results

5.1 Different `sec` Values

Table 5 shows the average number of clock cycles (10 runs) taken to generate a CGA for different levels of security. It is clear that the enhanced CGA Generation algorithm with a minimal computational security of $\mathcal{O}(2^{80})$ takes at least 180 ms on a N900. More modern mobile nodes will show better performance.

5.2 Different Hash Functions

Figure 6 shows the average number of clock cycles (30 runs) taken to compute `hash2` using different hash functions. As is obvious from the figure, the 4-pass HAVAL and the 5-pass HAVAL should be considered as excellent substitutes to SHA-3 because of their significantly superior performance. HAVAL-4 also provides the closest performance to SHA-256 out of all the hash functions compared in this work.

It is also important to remember that for hash functions, the level 1 cache size (for instructions) is one of the most important parameters affecting performance [20]. So in order to see improved results, manufactures should increase the level 1 cache size of mobile nodes. The N900 used to obtain the data in Figure 6 has configurable instruction and data caches of 16KiB - 32KiB.

5.3 Parallelism

Figure 7 compares the average number of clock cycles (100 runs) taken by the CGA Generation algorithm at $\mathcal{O}(2^{80})$. There are a few obvious points that can be noted from Figure 7.

- Spawning even one extra thread improves performance by about 20% (regardless of which method is used to parallelize the algorithm).
- In Method 1, the performance improves drastically (39%) when 2 threads (instead of 1 thread) are spawned by the main process. However, this improvement in performance slows down significantly as the number of threads increases to 3 or more.
- Likewise, for Method 2, the performance improves drastically (40%) when 2 threads (instead of 1 thread) are spawned by the main process. This improvement in performance slows down until four threads are spawned. After four threads, the performance actually gets worse.
- The best performance is obtained from Method 2 with four threads. Essentially, this means that the best performance is generally obtained by keeping the number of threads spawned equal to the number of cores (and they are 100% CPU-bound).

6 Conclusion

This paper reports a detailed investigation of the CCA Generation algorithm from a security and performance perspective. It proposes fixing a minimal computational security of $\mathcal{O}(2^{80})$ for the generation of a CGA and finds that this takes 180 ms on a typical mobile node like the N900. Over time (and increasingly powerful machines) this mimimal computational security should be increased. This paper also finds that HAVAL-4 and HAVAL-5 are the best alternatives to SHA-2 and SHA-3 from a performance viewpoint. With regards to taking advantage of multicore architectures, we find that Method 2 (for parallelising the CGA Generation algorithm) provides the maximum speedup when the number of threads spawned by the main thread equals the number of cores.

References

[1] A. AlSa'deh and C. Meinel, "Secure neighbor discovery: review, challenges, perspectives, and recommendations," *IEEE Security and Privacy*, vol. 10, no. 4, pp. 26–34, 2012.

[2] A. AlSa'deh, H. Rafiee and C. Meinel, "Multicore-based auto-scaling secure neighbor discovery for windows operating systems," in *Proceedings of 26th IEEE International Conference on Information Networking (ICOIN'12)*, pp. 257–262, Bali, Indonesia, Feb. 2012.

[3] T. Aura, "Cryptographically generated addresses," *Information Security*, LNCS 2851, pp. 29-43, 2003.

[4] T. Aura, *Cryptographically Gnerated Addresses (CGA)*, Technical Report RFC 3972, Mar. 2005. (http://tools.ietf.org/pdf/rfc3972.pdf)

[5] T. Aura and M. Roe, *Strengthening Short Hash Values*, May 10, 2015. (http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.145.7681)

[6] D. V. Bailey, C. Crutchfield, Y. Dodis, K. E. Fleming, A. Khan, J. Krishnamurthy, Y. Lin, L. Reyzin, E. Shen, J. Sukha, D. Sutherland, E. Tromer, R. Rivest, B. Agre and Y. L. Yin, *The MD6 Hash Function - A Proposal to NIST for SHA-3*, 2009. (`http://groups.csail.mit.edu/cis/md6/docs/2009-04-15-md6-report.pdf`)

Table 5: Different values of `sec`

`sec` security level	Computational security $\mathcal{O}(2^n)$	Average Number of Clock Cycles	Average Time on Nokia 900
0	$\mathcal{O}(2^{59})$	25,842	43μs
1	$\mathcal{O}(2^{67})$	44,201	74μs
2	$\mathcal{O}(2^{75})$	4,724,643	7.9ms
–	$\mathcal{O}(2^{80})$	107,806,357	180ms

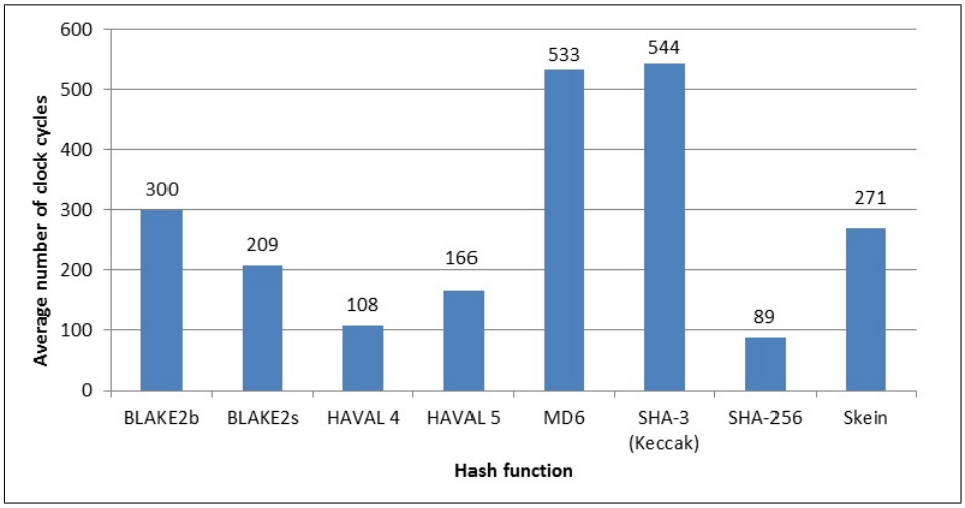


Figure 6: Comparison of different hash functions to calculate `hash2`

[7] Blake, *Blake2: Fast Secure Hashing*, May 10, 2015. (`https://blake2.net`)

[8] C. D. Canniere, J. Lano, H. Yoshida, A. Biryukov and B. Preneel, "Non-randomness of the full 4 and 5-pass haval," in *Security in Communication Networks*, LNCS 3352, pp. 324-336, 2005.

[9] T. Cheneau, A. Boudguiga, and M. Laurent, "Signifiantly improved performances of the cryptographically generated addresses thanks to ECC and GPU," *Computers and Security*, vol. 29, pp. 419–431, 2010.

[10] F. Cheng, A. AlSa'deh and C. Meinel, "CS-CGA: Compact and more secure CGA," in *Proceedings of 17th IEEE International Conference on Networks (ICON'11)*, pp. 299–304, Singapore, 2011.

[11] J. H. Davenport, S. Al-Kuwari, and R. J. Bradford, *Cryptographic Hash Functions: Recent Design Trends and Security Notions*, 2011. (`https://eprint.iacr.org/2011/565.pdf`)

[12] M. Doll, C. Vogt, R. Bless and T. Kuefner, "Early binding updates for mobile IPv6," in *Proceedings of IEEE Wireless Communications and Networking Conference*, vol. 3, pp. 1440–1445, Mar. 2005.

[13] C. C. Lee, M. S. Hwang and S.-K. Chong, "An improved address ownership in mobile IPv6," *Computer Communications*, vol. 31, no. 14, pp. 3250–3252, 2008.

[14] H. K. Lee and Y. Mun, "Design of modified CGA for address auto-configuration and digital signature in hierarchical mobile ad-hoc network," *Information Networking. Advances in Data Communications and Wireless Networks*, LNCS 3961, pp. 217-226, 2006.

[15] W. Meier, R. C.-W. Phan, J. P. Aumasson, L. Henzen, *SHA-3 Proposal BLAKE*, ver. 1.3, Dec. 16, 2010. (`http://131002.net/blake/blake.pdf`)

[16] K. E. S. Murthy, D. Kavitha and S. Z. ul Huq, "Security analysis of binding update protocols in route optimization of MIPv6," in *2010 International Conference on Recent Trends in Information, Telecommunication and Computing (ITC)*, pp. 44-49, Mar. 2010.

[17] O. Ozen, J. W. Bos, and J. P. Hubaux, "Analysis and optimization of cryptographically generated addresses," *Information Security*, LNCS 5735, pp. 17–32, 2009.

[18] A. Park, *Multithreaded Programming (POSIX Pthreads Tutorial)*, May 10, 2015. (`http://randu.org/tutorials/threads/`)

[19] Polarss, *Finding Collisions in the Full SHA-1*, May 10, 2015. (`http://polarssl.org/`)

[20] T. Pornin, *Comparative Performance Review of Most of the SHA-3 Second-round Candidates*, 2010. (`http://csrc.nist.gov/groups/ST/hash/`

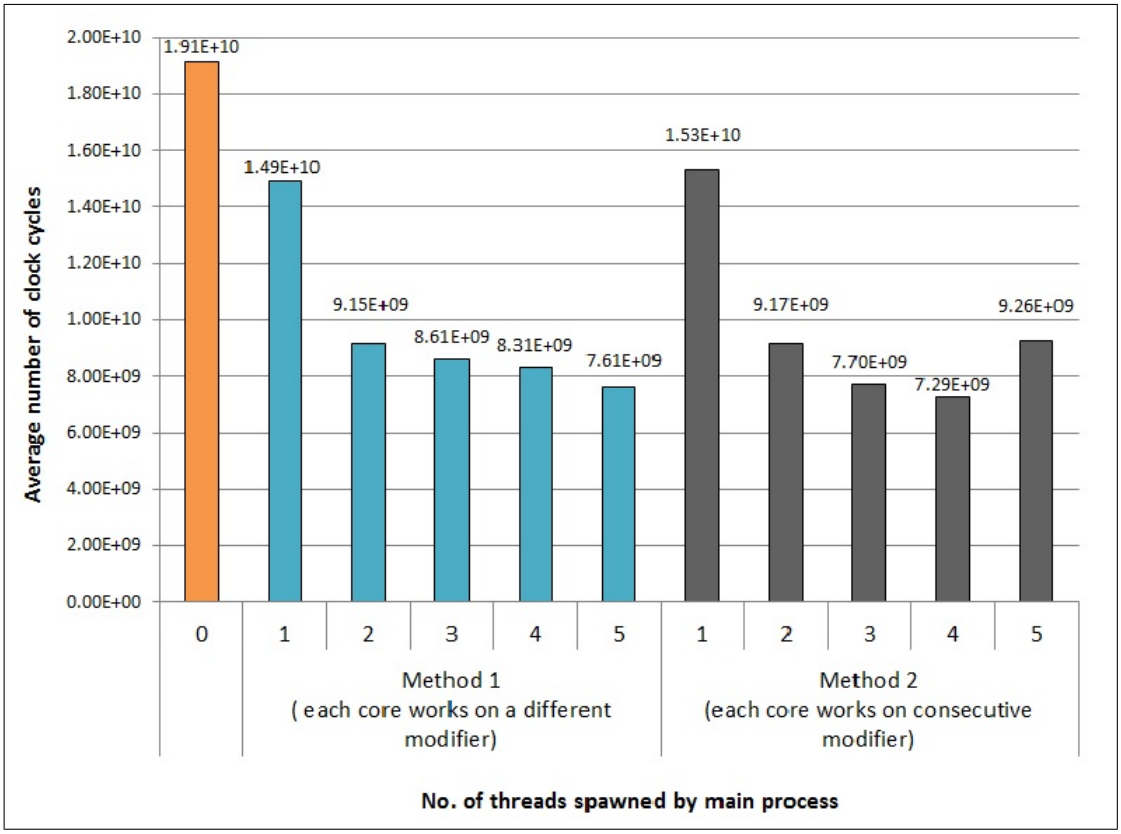


Figure 7: Effect of parallelism on CGA generation algorithm

sha-3/Round2/Aug2010/documents/papers/Pornin-report-sphlib-tp-final.pdf)

[21] H. Rafiee, A. AlSa'deh and C. Meinel, "Stopping time condition for practical IPv6 cryptographically generated addresses," in *Proceedings of International Conference on Information Networking (ICOIN'12)*, pp. 257–262, Bali, Indonesia, Feb. 2012.

[22] AlSa'deh H. Rafiee, A. AlSa'deh and C. Meinel, "Cryptographically generated addresses (CGAS): Possible attacks and proposed mitigation approaches," in *Proceedings of IEEE 12th International Conference on Computer and Information Technology (CIT'12)*, pp. 332–339, Chengdu, Sichuan, China, Oct. 2012.

[23] T. Rajendran and K. V. Sreenaath, "Hash optimization for cryptographically generated address," in *Proceedings of the 3rd International Conference on Communication Systems Software and Middleware and Workshops (COMSWARE'08)*, pp. 365–369, Bangalore, India, Jan. 2008.

[24] R. L. Rivest, *The MD6 Hash Algorithm*, May 10, 2015. (http://groups.csail.mit.edu/cis/md6)

[25] Scali, *Scali's Openblog: Multi-core and Multi-threading Performance (the Multi-core Myth?)*, June 1, 2012. (http://scalibq.wordpress.com/2012/06/01/multi-core-and-multi-threading)

[26] B. Schneier, D. Whiting, M. Bellare, T. Kohno, J. Callas, N. Ferguson, S. Lucks and J. Walker, *The Skein Hash Function Family*, ver. 1.3, Oct. 1, 2010. (https://www.schneier.com/skein1.3.pdf)

[27] M. U. Siddiqi, S. Qadir and W. F. M. Al-Khateeb, "An investigation of the merkle signature scheme (MSS) for cryptographically generated address (CGA) signatures in mobile IPv6," *International Journal of Network Security*, vol. 17, no. 3, pp. 311–321, 2015.

[28] R. Triggs, *What to Expect from Smartphone Hardware in late 2014 and into 2015*, 2014. (http://www.androidauthority.com/smartphone-hardware-2015-405022)

[29] G. Xiangyang, Q. Xirong, J. Sheng, S. Guangxue, W. Wendong and G. Xuesong, "A quick cga generation method," in *International Conference of Future Computer and Communication (ICFCC'10)*, pp. 769–773, Wuhan, China, May 2010.

[30] P. Zhang, J. Li and S. Sampalli, "Improved security mechanism for mobile ipv6," *International Journal of Network Security*, vol. 6, no. 3, pp. 291–300, 2008.

Sana Qadir received her MSc in Computer and Information Engineering in 2010. She is cuurently a PhD candidate at the Faculty of Engineering, International Islamic University Malayisia. Her research interests include information security, network security and implementation issues in cryptography.

Mohammad Umar Siddiqi received his B.Sc. and M.Sc. degrees from Aligarh Muslim University (AMU Aligarh) in 1966 and 1971, respectively, and a Ph.D. degree from the Indian Institute of Technology Kanpur (IIT Kanpur) in 1976, all in Electrical Engineering. He has been in the teaching profession throughout, first at AMU Aligarh, then at IIT Kanpur and Multimedia University Malaysia. Currently, he is a Professor in the Faculty of Engineering at International Islamic University Malaysia. His research interests are in coding, cryptography, and information security.

Wajdi Fawzi Mohammed Al-Khateeb received his MSc. Eng. degree in Telecommunications Engineering from the Technical University Berlin in 1968. After graduation he joined the University of Technology, Baghdad and Northern Petroleum Company, Iraq in 1971 as telecommunications engineer where he assumed various professional engineering activities including senior and chief telecommunications engineer until 1993. In 1995, he joined the Department of Electrical and Computer Engineering, International Islamic University Malaysia. Beside his academic activity, he was appointed as leader of consultancy team to plan, design, and supervise the ICT infrastructure project at the Universitys new campuses in Gombak and Kuantan with more than 30 thousand data/voice nodes to support the ICT applications of the University. He was later conferred a PhD in Engineering from IIUM in 2006. Dr. Wajdi is a professional telecommunications and IT engineer with expert knowledge in telecommunications engineering activities gained through 40 years of experience in many telecommunications systems covering: planning, design, consultation, project management and supervision of wide range of communications systems.

Provably-Secure Certificateless Key Encapsulation Mechanism for e-Healthcare System

Hui-Xian Shi[1] and Rui Guo[2]
(Corresponding author: Rui Guo)
Department of Mathematics and Information Science, Shaanxi Normal University, Xi'an 710119, China[1]
China Mobile Group Shaanxi Company Limited Xi'an Brand, Xi'an 710077, China[2]
(Email: grbupt@gmail.com)
(Received Feb. 10, 2015; revised and accepted Mar. 28 & Apr. 26, 2015)

Abstract

Modern information and communications technology have facilitated the traditional medical services in the healthcare system, which exchanges the physiological condition and diagnosis timely and remotely between the patient and the physician. However, there exist several privacy concerns as personal health information could be exposed to unauthorized parties. To ensure the confidentiality of this sensitive data, it is a promising method to encrypt it before delivering. Moreover, to generate and distribute a secure session key is a significant issue in the encryption algorithm. In this paper, we put forward a novel certificateless key encapsulation mechanism for the e-healthcare system, which is proven secure under the computational Diffie-Hellman assumption in the random oracle model. Furthermore, we compare our proposal with others in performance. Under the same simulation environment, the results show that the proposed scheme needs less computation and communication cost and appropriate to encapsulate the session key in the e-healthcare system.

Keywords: Certificateless key encapsulation mechanism, e-healthcare system, hybrid encryption, IND-CCA secure

1 Introduction

In recent years, information and communications technology have been employed in the traditional healthcare system. Some lightweight devices, such as wireless medical sensors, PDA and iPhone, increase the efficiency of this system and provide high-quality of care without sacrificing the patient comfort [17]. In the e-healthcare system, the wearable medical sensors are fixed on the patient to collect his/her physiological signals (e.g., blood pressure, pulse oximeter and temperature). Then, via a public wireless channel, these data are transmitted to the physician's handheld terminals, and the patient can be diagnosed timely and remotely.

However, the Health Insurance Portability and Accountability Act (HIPAA) enacted in 1996 [11] demonstrated that the patient's physiological conditions are all the sensitive information, which relate to his/her privacy and should be protected. If the privacy is eavesdropped by an unauthorized party, the safety and economic interests of the patient would be threatened. Thus, during the transmission in the public channel, it needs a secure encryption scheme to ensure the confidentiality of the transmitted data between the patient and the physician in the e-healthcare system.

Considering the encryption scheme for e-healthcare system, for the reason that this system consists of the lightweight devices with limited memory, small bandwidth and low power, it should preserve two outstanding characteristics with efficiency and confidentiality during designing an encryption scheme.

There are two models of encryption in cryptography, asymmetric and symmetric key encryption systems. In the public key encryption (i.e., asymmetric encryption), the most of schemes in the literature have limited message spaces, which means that a message to be encrypted is assumed to have a limited length or belong to a specific group. It is inconvenient and expensive for ensuring the confidentiality of arbitrary messages by using of a purely public key encryption. As enjoy high efficiency, symmetric encryption schemes are usually employed to encrypt large messages, such as DES [4, 16]. Unfortunately, they also suffer from the key distribution problem. To achieve high efficiency while avoiding the key distribution problem in the encryption system, the normal method of performing the public key encryption is to divide the encryption scheme into two parts: one part uses the public key techniques to encrypt a one-time symmetric key; the other part takes use of this symmetric key to encrypt the transmitted message. In such a construction, the public key part of the algorithm is called the key en-

capsulation mechanism (KEM) while the symmetric key part (where the message is actually encrypted) is known as the data encapsulation mechanism (DEM). According to this encryption model, KEM can provide an efficient and secure method to deliver a random key from a sender to a designated receiver, and DEM enables to increase efficiency over public key encryption. Combining KEM and DEM, the resulting scheme is then called a hybrid encryption scheme which has received much attention in recent years [1, 9, 10].

In the traditional public key infrastructure (PKI), encryption is achieved through the certificates issued by a trust certification authority (CA). In [5], Dent presented a lot of generic constructions of KEM from standard public key encryption, however these led to the problem of certificates management (including distribution, storage, revocation and verification of certificates), which placed a large computation cost on the system. To avoid these weaknesses, Shamir [15] proposed the identity based public key cryptography (ID-PKC) by deriving the user's public key directly from some public parameters and the user's identity, such as email and IP address. In 2008, Bentahar et al. [3] extended the concept of key encapsulation to the primitives of identity based encryption that are provably secure in the random oracle model. Nevertheless, there is a trusted third party called the private key generator (PKG) in ID-PKC whose behavior is in possession of a master secret key (which is used to derive the private key of any user in this system). Thus, the private key of all the users in ID-PKC is known to the PKG. This inherent issue in ID-PKC is called the key escrow problem [14]. Therefore, these two types of cryptographic primitive above are not suitable for protecting the entity's privacy with lightweight mobile devices, such as in e-healthcare system.

To overcome the key escrow problem, certificateless public key cryptography (CL-PKC) was introduced by Al-Riyami and Paterson [2]. In the certificateless key encapsulation mechanism (CL-KEM), the user's private key is split into two parts: one is the partial private key obtained from the key generation center (KGC), the other one is a user's selected secret value. Consequently, the trusted third party KGC cannot access the user's private key to reveal his/her privacy any more. Several CL-KEM protocols have been proposed in the last decade [3, 8, 12]. Huang and Wong [8] proposed the first generic construction of CL-KEM in the standard model, which was secure against malicious-but-passive KGC attacks. In [3], Bentahar et al. also took any IBE scheme plus a special form of public key scheme, such as RSA or ElGamal in certain groups, and used them to construct a CL-KEM, which was secure in a strong sense. However, these two schemes combined a public key based encryption scheme and an identity based KEM and thus very inefficient. Lippold and Boyd [12] presented a direct construction for a chosen ciphertext secure (CCA secure) CL-KEM in the standard model that was more efficient than the generic constructions.

In this paper, we put forward a certificateless key encapsulation mechanism for e-healthcare system, and prove that it is secure in the random oracle model against the chosen ciphertext attacks. Furthermore, this scheme achieves the Girault's trust Level 3 which ensures the credibility of the authority [6]. Compared to the related schemes, through the evaluations and experiments, our protocol offers a better performance in the computation and communication cost.

In the next section, we review some computational assumptions, the model and the security definitions of CL-KEM that will be used throughout the paper. In Section 3, we design a new protocol for e-healthcare and analyze the security of it. In Section 4, we compare the efficiency with related schemes and conclude the paper in Section 5.

2 Preliminaries

2.1 Complexity Assumptions

Let G be a cyclic additive group with prime order p, and P be a generator of G.

Definition 1. *Discrete Logarithm (DL) problem: Given* $(P, Q \in G)$*, find an integer* $x \in Z_p^*$ *satisfying* $Q = xP$*.*

The DL assumption is that there is no polynomial time algorithm that can solve the DL problem with non-negligible probability.

Definition 2. *Computational Diffie-Hellman (CDH) problem: Given* $(Q = xP, R = yP) \in G^2$ *for any* $x, y \in Z_p^*$*, compute* xyP*.*

The CDH assumption is that there is no polynomial time algorithm that can solve the CDH problem with non-negligible probability.

2.2 Certificateless Key Encapsulation Mechanism

A CL-KEM for e-healthcare system consists of the following seven probabilistic polynomial time (PPT) algorithms: *Setup*, *User-Key-Generation*, *Partial-Key-Extract*, *Set-Private-Key*, *Set-Public-Key*, *Encap* and *Decap*.

Setup. On input a security parameter 1^k, the medical server (MS) returns the system parameters *params*, the master public/secret key (*mpk*,*msk*). Then, MS publishes *params* and *mpk*, and keeps the *msk* secret.

User-Key-Generation. On input the system parameters *params*, the patient returns a pair of public/secret key (*pk*,*sk*).

Partial-Key-Extract. On input *params*, *msk*, patient's identity ID_P and his/her public key *pk*, MS

executes this algorithm and returns a partial private key D_P to patient via a confidential and authentic channel, and the corresponding partial public key P_P.

Set-Private-Key. On input *params*, patient's partial private key D_P and his/her secret key sk, this algorithm returns private key SK_P to the patient.

Set-Public-Key. On input *params*, patient's partial public key P_P and his/her public key pk, this algorithm returns the patient's public key PK_P.

Encap. Running by a doctor. On input *params*, the patient's identity ID_P, and his/her public key PK_P, this algorithm outputs an encapsulation key pair $(K, c) \in (\mathcal{K}, \mathcal{C})$, where c is called the encapsulation of key K, and K is considered to be distributed uniformly in the key space $\mathcal{K}$. (In the hybrid encryption primitive, the doctor encrypts the privacy data by using of this K in symmetric encryption scheme.)

Decap. Running this deterministic algorithm by a patient. On input *params*, the encapsulation c, and his/her private key SK_P, this algorithm outputs the corresponding key K, or an invalid encapsulation $\perp$. (Similarly, the patient decrypts the ciphertext above with this decapsulation K.)

In this system, to achieve the Girault's trust Level 3, the *User-Key-Generation* algorithm must be run prior to the *Partial-Key-Extract* algorithm. The patient fixes his/her secret key sk and public key pk firstly. Then, MS generates patient's partial key D_P by binding his/her public key to an identity ID_P. According to this way, although MS can replace patient's public key pk, there will exist a pair of working public keys (pk, pk') for only one patient. Moreover, two working different public keys (PK_P, PK'_P) binding one patient's identity can result from two partial private keys, and only the MS has ability to generate these two working partial private keys. Hence, the MS's forgery is easily tracked, which means that the trust level of MS is achieving to the Girault's trust Level 3 as described in [6].

2.3 Security Model

In certificateless cryptography, there are two types of adversaries $\mathcal{A}_I$ and $\mathcal{A}_{II}$. Type-I adversary $\mathcal{A}_I$ acts as a dishonest user who does not have access to MS's master secret key and patient's partial key, but it enables to compromise user's private key or replace the public key of any patient with its own choices value. By contrast, Type-II adversary $\mathcal{A}_{II}$ plays the part of a malicious-but-passive MS who controls the master secret key msk (hence it can compute patient's partial secret key). Besides, Type-II adversary $\mathcal{A}_{II}$ is allowed to receive private keys for arbitrary identities but cannot replace any patient's public key. The following oracles are the interactive game between challenger $\mathcal{C}$ and adversary $\mathcal{A}$.

Setup. The challenger $\mathcal{C}$ runs this algorithm to generate the public parameters *params* and the master public/private key pair (mpk, msk).

Partial-Key-Extract-Oracle. Upon receiving an identity ID, this oracle computes the corresponding partial public/private key pair $(P_{\mathrm{ID}}, D_{\mathrm{ID}})$ and sends this tuple to $\mathcal{A}$.

Private-Key-Request-Oracle. Upon receiving an identity ID, if the ID's public key has not been replaced, $\mathcal{C}$ responds it with the private key SK_{ID}. Otherwise, $\mathcal{C}$ does not provide the corresponding private key to $\mathcal{A}$.

Public-Key-Request-Oracle. Upon receiving an identity ID, $\mathcal{C}$ responds it with the public key PK_{ID}.

Replace-Public-Key-Oracle. $\mathcal{A}$ can repeatedly replace the public key PK_{ID} with any value PK'_{ID} of its own choice. The current value of the user's public key is used by $\mathcal{C}$ in any computations or to response to $\mathcal{A}$'s queries.

Decapsulation-Oracle. Upon receiving an identity ID and an encapsulation c, if there is no query on ID, return $\perp$. Otherwise, return $K \leftarrow Decap(\mathrm{ID}, SK_{\mathrm{ID}}, c)$ as a decapsulation of c.

We now specify two games for Type-I and Type-II security described as follows.

Game-I: Let $\mathcal{C}_I$ be a challenger to Type-I adversary $\mathcal{A}_I$ and 1^k be a security parameter.

1) $\mathcal{C}_I$ computes $(mpk, msk) \leftarrow Setup(1^k)$, and runs $\mathcal{A}_I$ on input 1^k and mpk.
2) $\mathcal{A}_I$ can query *Partial-Key-Extract-Oracle*, *Private-Key-Request-Oracle*, *Public-Key-Request-Oracle*, *Replace-Public-Key-Oracle* and *Decapsulation-Oracle*. Then, $\mathcal{A}_I$ submits a target identity $\mathrm{ID}^* \in \{0,1\}^*$.
3) $\mathcal{C}_I$ runs $(K_1, c^*) \leftarrow Encap(mpk, PK_{\mathrm{ID}^*}, \mathrm{ID}^*)$ and randomly selects $(K_0 \leftarrow \mathcal{K})$. Then, $\mathcal{C}_I$ flips a coin b, and returns (K_b, c^*) to $\mathcal{A}_I$.
4) $\mathcal{A}_I$ continues to issue queries as in Step (2). Finally, it outputs a bit b'.

$\mathcal{A}_I$ wins this game if $b' = b$. Note that $\mathcal{A}_I$ is not allowed to query *Partial-Key-Extract-Oracle* on ID^* and *Decapsulation-Oracle* on (ID^*, c^*). We define the advantage of $\mathcal{A}_I$ in **Game-I** to be $Adv(\mathcal{A}_I) = |\Pr(b' = b) - \frac{1}{2}|$.

Game-II: Let $\mathcal{C}_{II}$ be a challenger to Type-II adversary $\mathcal{A}_{II}$ and 1^k be a security parameter.

1) $\mathcal{C}_{II}$ runs $\mathcal{A}_{II}$ on input 1^k and returns $(mpk, msk) \leftarrow Setup(1^k)$ as an answer.

2) $\mathcal{A}_{II}$ can query *Private-Key-Request-Oracle*, *Public-Key-Request-Oracle* and *Decapsulation-Oracle*. Then $\mathcal{A}_{II}$ submits a target identity $\mathrm{ID}^* \in \{0,1\}^*$. Note that *Partial-Key-Extract-Oracle* is not allowed by $\mathcal{A}_{II}$ because of the knowledge of msk.

3) $\mathcal{C}_{II}$ runs $(K_1, c^*) \leftarrow Encap(mpk, PK_{\mathrm{ID}^*}, \mathrm{ID}^*)$ and randomly selects $(K_0 \leftarrow \mathcal{K})$. Then, $\mathcal{C}_{II}$ flips a coin b, and returns (K_b, c^*) to $\mathcal{A}_{II}$.

4) $\mathcal{A}_{II}$ continues to issue queries as in Step (2). Finally, it outputs a bit b'.

$\mathcal{A}_{II}$ wins this game if $b' = b$. Note that $\mathcal{A}_{II}$ is not allowed to query *Private-Key-Request-Oracle* on ID^* and *Decapsulation-Oracle* on (ID^*, c^*). We define the advantage of $\mathcal{A}_{II}$ in **Game-II** to be $Adv(\mathcal{A}_{II}) = |\Pr(b' = b) - \frac{1}{2}|$.

Definition 3. *A CL-KEM* Π *is secure against chosen ciphertext attack (IND-CCA secure) if neither polynomial bounded adversary* $\mathcal{A}$ *of Type-I nor Type-II has a non-negligible advantage against the challenger in the* ***Game-I*** *and* ***Game-II****.*

$\mathcal{A}$ breaks an IND-CCA secure CL-KEM Π with $(q_H, q_{par}, q_{pri}, q_{pub}, q_D, \varepsilon)$ if and only if the advantage of $\mathcal{A}$ that makes q_H times to the random oracle $H(\cdot)$, q_{par} times *Partial-Key-Extract-Oracle*, q_{pri} times *Private-Key-Request-Oracle*, q_{pub} times *Public-Key-Request-Oracle* and q_D times *Decapsulation-Oracle* queries is greater than ε. The scheme Π is said to be $(q_H, q_{par}, q_{pri}, q_{pub}, q_D, \varepsilon)$-IND-CCA secure if there is no adversary $\mathcal{A}$ that breaks IND-CCA secure scheme Π with $(q_H, q_{par}, q_{pri}, q_{pub}, q_D, \varepsilon)$.

3 Our CL-KEM

In this section, we put forward a novel CL-KEM without bilinear pairing to encapsulate a one-time symmetric key between the patient and doctor. The notations used throughout this protocol are listed in Table 1.

Table 1: Notions of this scheme

ID_P	the identity of Patient
$H_i(\cdot)$	the collision-resistant hash function (i=1,2)
p	the large prime number
G	the cyclic additive group
P	the generator of G
x	the master secret key
X	the master public key
P_P	the Patient's partial public key
D_P	the Patient's partial private key
PK_P	the Patient's public key
SK_P	the Patient's private key
$\|$	the connection operation

3.1 Construction

The proposed CL-KEM as shown in Figure 1 consists of the following seven PPT algorithms.

Setup. Let G be a cyclic group of prime order p with an arbitrary generator $P \in G$. The MS selects $x \in Z_p^*$ randomly and computes $X = xP$ as the master public key. Then, it chooses two collision resistant hash functions $H_1 : \{0,1\}^{l_0} \times G^* \times G^* \to Z_p^*$ and $H_2 : \{0,1\}^{l_0} \times G^{*5} \to \{0,1\}^*$. The system parameters are $params = (p, G, P, X, H_1, H_2)$, and the master secret key is $msk = x$.

User-Key-Generation. Patient picks $y \in Z_p^*$ uniformly at random and computes $Y = yP$, and he/she returns $(sk, pk) = (y, Y)$.

Partial-Key-Extract. MS picks $\alpha \in Z_p^*$ at random and computes $r_P = \alpha P$ and $z_P = \alpha + xH_1(\mathrm{ID}_P \parallel r_P \parallel pk)$, where ID_P is the patient's identity. Then MS returns $(P_P, D_P) = (r_P, z_P)$ as a pair of patient's partial key.

Set-Private-Key. Set $SK_P = (sk, D_P) = (y, z_P)$, it returns SK_P as the patient's private key.

Set-Public-Key. Let $PK_P = (pk, P_P) = (Y, r_P)$, it returns PK_P as the patient's public key.

Encap. Doctor picks $u \in Z_p^*$ randomly and computes the ciphertext:

$$\begin{aligned} c &= uP, \\ c_1 &= u(Y + r_P + XH_1(\mathrm{ID}_P \parallel r_P \parallel pk)), \\ c_2 &= uY, \end{aligned}$$

and the corresponding session key is

$$K = H_2(\mathrm{ID}_P \parallel PK_P \parallel c \parallel c_1 \parallel c_2).$$

Then the doctor delivers the encapsulation $\{c\}$ to patient.

Decap. To decapsulate c, the patient reconstructs the session key as

$$K = H_2(\mathrm{ID}_P \parallel PK_P \parallel c \parallel (y + z_P)c \parallel yc).$$

Then in the hybrid scheme, a symmetric encryption scheme is taken to protect the privacy under this K.

The above *Decap* algorithm is consistent if c is a valid encapsulation, then it is easy to verify that,

$$\begin{aligned} (y + z_P)c &= yuP + (\alpha + xH_1(\mathrm{ID}_P \parallel r_P \parallel pk))uP \\ &= u(yP + r_P + XH_1(\mathrm{ID}_P \parallel r_P \parallel pk)) \\ &= c_1, \end{aligned}$$

and

$$yc = yuP = uY = c_2.$$

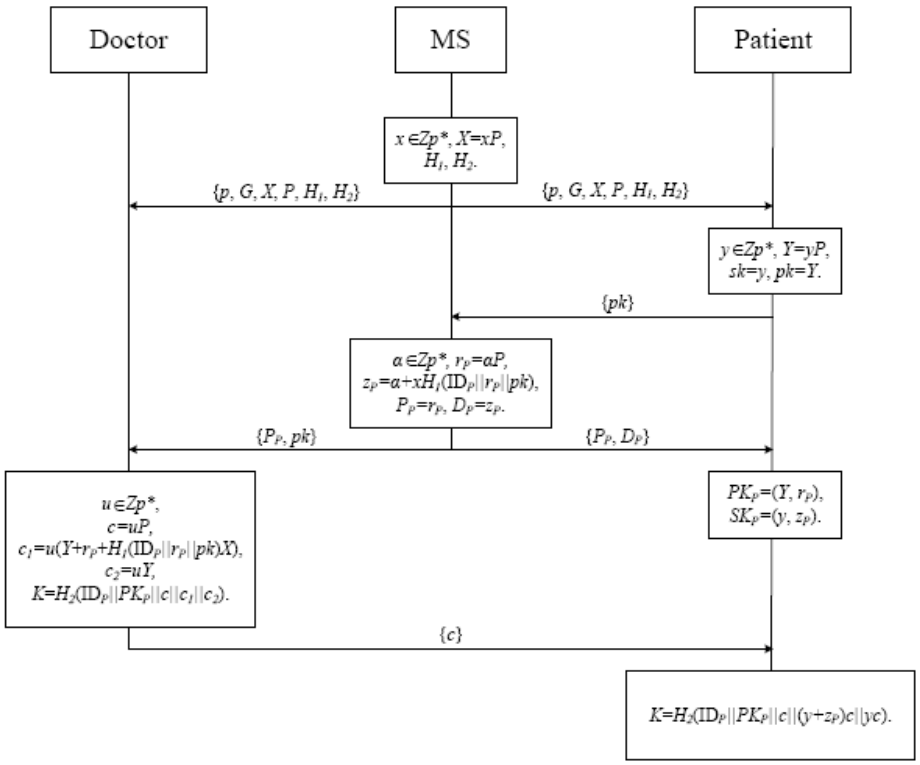


Figure 1: Our CL-KEM for e-healthcare system

3.2 Security Analysis

In this subsection, we prove that the CL-KEM presented in the previous is secure in the random oracle model.

Theorem 1. *Provided that H_1 and H_2 are two collision resistant hash functions. This CL-KEM is IND-CCA secure in the random oracle model assuming that there is no polynomial time algorithm that can solve the CDH problem with non-negligible probability.*

This theorem following from two lemmas will show that our CL-KEM is secure against the Type-I and Type-II adversaries whose behaviors are as described in the **Game-I** and **Game-II**.

Lemma 1. *This CL-KEM is $(q_H, q_{par}, q_{pri}, q_{pub}, q_D, \varepsilon)$-IND-CCA secure against the Type-I adversary $\mathcal{A}$ in the random oracle model, then there exists an algorithm $\mathcal{B}$ that solves the CDH problem with the following advantage*

$$\varepsilon' > \frac{1}{q_{H_2}}\left(\frac{2\varepsilon}{e(q_{prv}+1)} - \frac{q_D q_{H_1}}{2^{l_0}p^2} - \frac{q_D}{2^{l_0}p^5}\right).$$

Proof. Assuming there exists a Type-I adversary $\mathcal{A}_I$ imitating an "outside" adversary, who replaces the public key of arbitrary identities but cannot corrupt the master secret key.

Suppose that there is another PPT algorithm $\mathcal{B}$ can solve the CDH problem in the instance of (p, P, aP, xP) with probability at least ε' by interacting with $\mathcal{A}_I$. To solve this problem, $\mathcal{B}$ needs to simulate a challenger to run each algorithm of **Game-I** for $\mathcal{A}_I$ as follows:

Setup. Algorithm $\mathcal{B}$ sets the master public key $X = xP$, where $x \in Z_p^*$ is the master secret key that is unknown to $\mathcal{B}$. Then $\mathcal{B}$ gives $\mathcal{A}_I$ the $params = \{p, G, P, X, H_1, H_2\}$ as system parameters. $\mathcal{A}_I$ performs a series of polynomially bounded number of queries according to the following oracles:

H_1 **Queries.** $\mathcal{B}$ maintains a list of tuples $\langle(\mathrm{ID}, r_{\mathrm{ID}}, Y), v\rangle$ in H_1-**List** L_1. On receiving a query $(\mathrm{ID}, r_{\mathrm{ID}}, Y)$ to H_1:

1) If $\langle(\mathrm{ID}, r_{\mathrm{ID}}, Y), v\rangle$ already appears on the list L_1, $\mathcal{B}$ responds v as an answer.
2) Otherwise, pick $v \in Z_p^*$ randomly, add $\langle(\mathrm{ID}, r_{\mathrm{ID}}, Y), v\rangle$ to L_1 and return v as an answer.

H_2 **Queries.** $\mathcal{B}$ maintains a list of tuples $\langle(\mathrm{ID}, T), R\rangle$ in H_2-**List** L_2, where $T \in G^{*5}$. On receiving a query (ID, T) to H_2:

1) If $\langle(\mathrm{ID}, T), R\rangle$ exists in the list L_2, $\mathcal{B}$ responds R as an answer.
2) Otherwise, choose $R \in \{0,1\}^*$ uniformly at random, add $\langle(\mathrm{ID}, T), R\rangle$ to L_2 and return R as an answer.

Phase 1. $\mathcal{A}_I$ can issue a number of the following oracle queries.

Partial-Key-Extract-Oracle. $\mathcal{B}$ maintains a **PartialKeyList** of tuples $\langle \mathrm{ID}, (r_{\mathrm{ID}}, z_{\mathrm{ID}})\rangle$. On receiving a query ID, $\mathcal{B}$ responds as follows:

1) If $\langle \mathrm{ID}, (r_{\mathrm{ID}}, z_{\mathrm{ID}})\rangle$ exists in **PartialKeyList**, return $(r_{\mathrm{ID}}, z_{\mathrm{ID}})$ as an answer.
2) Otherwise, pick z_{ID}, $v \in Z_p^*$ at random, and compute $r_{\mathrm{ID}} = z_{\mathrm{ID}}P - vX$. Add $(\mathrm{ID}, r_{\mathrm{ID}}, v)$ to L_1 and $\langle \mathrm{ID}, (r_{\mathrm{ID}}, z_{\mathrm{ID}})\rangle$ to **PartialKeyList**, return $(r_{\mathrm{ID}}, z_{\mathrm{ID}})$ as an answer.

Public-Key-Request-Oracle. $\mathcal{B}$ maintains a **PublicKeyList** of tuples $\langle \mathrm{ID}, (r_{\mathrm{ID}}, Y), coin\rangle$. On receiving a query ID, $\mathcal{B}$ responds as follows:

1) If $\langle \mathrm{ID}, (r_{\mathrm{ID}}, Y), coin\rangle$ exists in **PublicKeyList**, return $PK_{\mathrm{ID}} = (r_{\mathrm{ID}}, Y)$ as an answer.
2) Otherwise, choose $coin \in \{0, 1\}$ at random so that $\Pr[coin = 0] = \delta$ (δ will be defined later).
3) If $coin = 0$, do the following:
   a. If $\langle \mathrm{ID}, (r_{\mathrm{ID}}, z_{\mathrm{ID}})\rangle$ exists in **PartialKeyList**, pick $y \in Z_p^*$ at random and compute $Y = yP$. Then, add $\langle \mathrm{ID}, (y, z_{\mathrm{ID}})\rangle$ to **PrivateKeyList** (which will be defined later) and $\langle \mathrm{ID}, (r_{\mathrm{ID}}, Y), coin\rangle$ to **PublicKeyList** respectively, return $PK_{\mathrm{ID}} = (r_{\mathrm{ID}}, Y)$ as an answer.
   b. Otherwise, run the *Partial-Key-Extract-Oracle* to get partial keys $(r_{\mathrm{ID}}, z_{\mathrm{ID}})$ about ID. Pick $y \in Z_p^*$ at random and compute $Y = yP$. Then, add $\langle \mathrm{ID}, (r_{\mathrm{ID}}, z_{\mathrm{ID}})\rangle$ to **PrivateKeyList** and $\langle \mathrm{ID}, (r_{\mathrm{ID}}, Y), coin\rangle$ to **PublicKeyList** respectively, return $PK_{\mathrm{ID}} = (r_{\mathrm{ID}}, Y)$ as an answer.
4) Otherwise (if $coin = 1$), pick $\alpha, y \in Z_p^*$ at random and compute $r_{\mathrm{ID}} = \alpha P$, $Y = yP$, add $\langle \mathrm{ID}, (y, *), \alpha\rangle$ to **PrivateKeyList** (where * denotes the arbitrary value), and $\langle \mathrm{ID}, (r_{\mathrm{ID}}, Y), coin\rangle$ to **PublicKeyList**, return $PK_{\mathrm{ID}} = (r_{\mathrm{ID}}, Y)$ as an answer.

Private-Key-Request-Oracle. $\mathcal{B}$ maintains a **PrivateKeyList** of tuples $\langle \mathrm{ID}, (y, z_{\mathrm{ID}}), \alpha\rangle$. On receiving a query ID, $\mathcal{B}$ responds as follows:

1) Perform *Public-Key-Request-Oracle* on ID to get a tuple $\langle \mathrm{ID}, (r_{\mathrm{ID}}, Y), coin\rangle$ from **PublicKeyList**.
2) If $coin = 0$, search a tuple $\langle \mathrm{ID}, (y, z_{\mathrm{ID}}), \alpha\rangle$ in **PrivateKeyList** and return $SK_{\mathrm{ID}} = (y, z_{\mathrm{ID}})$ as an answer.
3) Otherwise, return "*Abort*" and terminate this algorithm.

Replace-Public-Key-Oracle. $\mathcal{A}_I$ may replace any public key with a new value of its choice and $\mathcal{B}$ records all the changes.

Decapsulation-Oracle. On receiving a query $\langle \mathrm{ID}, PK_{\mathrm{ID}}, c\rangle$, where $PK_{\mathrm{ID}} = (r_{\mathrm{ID}}, Y)$. $\mathcal{B}$ responds as follows:

1) Search a tuple $\langle \mathrm{ID}, (r_{\mathrm{ID}}, Y), coin\rangle$ in **PublicKeyList**.
2) If such a tuple exists and $coin = 0$.
   a. Search **PrivateKeyList** for a tuple $\langle \mathrm{ID}, (y, z_{\mathrm{ID}})\rangle$.
   b. Compute $K = H_2(\mathrm{ID} \parallel PK_{\mathrm{ID}} \parallel c \parallel (y + z_{\mathrm{ID}})c \parallel yc)$.
3) Else, if such a tuple exists and $coin = 1$.
   a. Perform H_1 **queries** to get a tuple $\langle \mathrm{ID}, (r_{\mathrm{ID}}, Y), v\rangle$.
   b. If there exists $\langle (\mathrm{ID}, T), R\rangle \in L_2$ such that $R = H_2(\mathrm{ID} \parallel T)$, return R as the session key and "*Reject*" otherwise.
4) Else, if such a tuple does not exist (which means that the public key of a target user is replaced by $\mathcal{A}_I$), run the same algorithm in (3).

Challenge Phase. Once $\mathcal{A}_I$ decides that *Phase 1* is over, it outputs a challenge identity ID^*. On receiving a challenge query ID^*, $\mathcal{B}$ responds as follows:

1) Run *Public-Key-Request-Oracle* on ID^* to get a tuple $\langle \mathrm{ID}^*, (r_{\mathrm{ID}^*}, Y^*), coin\rangle$ in **PublicKeyList**.
2) If $coin = 0$, return "*Abort*" and terminate.
3) Otherwise, do the following:
   a. Search a tuple $\langle \mathrm{ID}^*, (y^*, *), \alpha\rangle$ in **PrivateKeyList**. (In this case, we know that $r_{\mathrm{ID}^*} = \alpha^* P$, $Y^* = y^* P$).
   b. Set $c^* = aP$, $c_1^* = a(Y^* + r_{\mathrm{ID}^*} + XH_1(\mathrm{ID}^* \parallel r_{\mathrm{ID}^*} \parallel Y^*))$ and $c_2^* = aY^*$. Note that $\mathcal{B}$ does not know "a".
   c. Compute $\Gamma = ar_{\mathrm{ID}^*}$ and $v^* = H_1(\mathrm{ID}^* \parallel r_{\mathrm{ID}^*} \parallel Y^*)$.
   d. Pick $K_0 \in_R \mathcal{K}$, where $\mathcal{K}$ is the key space.
   e. Compute $K_1 = H_2(\mathrm{ID}^* \parallel (r_{\mathrm{ID}^*}, Y^*) \parallel c^* \parallel c_1^* \parallel c_2^*)$.
4) Choose a bit $\beta \in_R \{0, 1\}$ and return (c^*, K_β) to $\mathcal{A}_I$.

Phase 2. $\mathcal{A}_I$ repeats the queries in *Phase 1*. However, there is no *Partial-Key-Extract-Oracle* or *Private-Key-Request-Oracle* query on ID^* is allowed. Also, no *Decapsulation-Oracle* query should be made on the encapsulation c^* for ID^*.

Guess. $\mathcal{A}_I$ outputs a guess β' for β, and wins the game if $\beta' = \beta$. Then, $\mathcal{B}$ will be able to solve the CDH problem by computing $(c^* \cdot z_{\mathrm{ID}^*} - \Gamma)/v^*$.

Analysis. We denote the event that ID^* has been queried to H_1 as $\mathbf{Ask}H_1^*$. Also, by $\mathbf{Ask}H_2^*$, we denote the event that $\langle(\mathrm{ID}^*, T^*), R^*\rangle$ has been queried to H_2. Provided that the event $\mathbf{Ask}H_2^*$ happens, $\mathcal{B}$ will solve the CDH problem by picking a tuple $\langle(\mathrm{ID}^*, T^*), R^*\rangle$ in L_2 and computing $(c^* \cdot z_{\mathrm{ID}^*} - \Gamma)/v^*$ with probability at least $1/q_{H_2}$. Hence, we have $\varepsilon' \geq (1/q_{H_2})\Pr[\mathbf{Ask}H_2^*]$.

If $\mathcal{B}$ does not abort in the **Game-I**, the simulations of *Partial-Key-Extract-Oracle*, *Public-Key-Request-Oracle*, *Private-Key-Request-Oracle* and the target encapsulation is identically distributed in our construction. Also, $\mathcal{B}$'s responses to all hash queries are uniformly and independently distributed as in the real attack, and all responses to $\mathcal{A}_I$ can pass the validity test unless $\mathcal{B}$ aborts. Thus, we find that when a public key PK_{ID} has not been replaced or produced under $coin = 1$, the simulation is perfect as $\mathcal{B}$ knowing the corresponding private key SK_{ID}. Otherwise, a simulation error may occur in *Decapsulation-Oracle*, and let **DecErr** denote this event. Suppose that ID, $PK_{\mathrm{ID}} = (r_{\mathrm{ID}}, Y)$ and c have been issued as a valid decapsulation query. Even if K is a valid session key, there is a possibility that K can be produced without querying $\langle(\mathrm{ID}, T), R\rangle$ to H_2. Let **Valid** be an event that K is a valid session key, $\mathbf{Ask}H_1$ and $\mathbf{Ask}H_2$ be events that $(\mathrm{ID}, r_{\mathrm{ID}}, Y)$ has been queried to H_1 and (ID, T) to H_2 respectively. Since **DecErr** is an event that $\mathbf{Valid}|\neg\mathbf{Ask}H_2$ happens during the entire simulation and q_D *Decapsulation-Oracle* queries are operated, we have $\Pr[\mathbf{DecErr}]=q_D\Pr[\mathbf{Valid}|\neg\mathbf{Ask}H_2]$, where $\Pr[\mathbf{Valid}|\neg\mathbf{Ask}H_2] \leq \Pr[\mathbf{Valid}\wedge\mathbf{Ask}H_1|\neg\mathbf{Ask}H_2]$ + $\Pr[\mathbf{Valid}\wedge\neg\mathbf{Ask}H_1|\neg\mathbf{Ask}H_2] \leq \Pr[\mathbf{Ask}H_1|\neg\mathbf{Ask}H_2]$ + $\Pr[\mathbf{Valid}|\neg\mathbf{Ask}H_1 \wedge \neg\mathbf{Ask}H_2] \leq (q_{H_1}/(2^{l_0}p^2)) + (1/(2^{l_0}p^5))$.

Let the event $(\mathbf{Ask}H_2^* \vee \mathbf{DecErr})|\neg\mathbf{Abort}$ be denoted by $\mathbf{E}$, where **Abort** is an event that $\mathcal{B}$ aborts during the simulation. The probability $\neg\mathbf{Abort}$ that happens is given by $\delta^{q_{prv}}(1-\delta)$ which is maximized at $\delta = 1 - 1/(q_{prv}+1)$. Hence, we have $\Pr[\neg\mathbf{Abort}] \leq 1/(e(q_{prv}+1))$, where e denotes the base of the natural logarithm.

If $\mathbf{E}$ does not happen, it is clear that $\mathcal{A}_I$ does not gain any advantage greater than 1/2 to guess β due to the randomness of the output of the random oracle H_2. Namely, we have $\Pr[\beta' = \beta \mid \neg\mathbf{E}] \leq 1/2$.

By definition of ε, we have $\varepsilon < |\Pr[\beta' = \beta] - (1/2)| = |\Pr[\beta' = \beta|\neg\mathbf{E}]\Pr[\neg\mathbf{E}] + \Pr[\beta' = \beta|\mathbf{E}]\Pr[\mathbf{E}] - (1/2)| \leq |(1/2)\Pr[\neg\mathbf{E}] + \Pr[\mathbf{E}] - (1/2)| = |(1/2)(1 - \Pr[\mathbf{E}]) + \Pr[\mathbf{E}] - (1/2)| = (1/2)\Pr[\mathbf{E}] \leq (\Pr[\mathbf{Ask}H_2^*] + \Pr[\mathbf{DecErr}])/(2\Pr[\neg\mathbf{Abort}]) \leq (e(q_{prv}+1)/2)(q_{H_2}\varepsilon' + (q_D q_{H_1}/(2^{l_0}p^2)) + (q_D/(2^{l_0}p^5)))$. Consequently, we obtain

$$\varepsilon' > \frac{1}{q_{H_2}}\left(\frac{2\varepsilon}{e(q_{prv}+1)} - \frac{q_D q_{H_1}}{2^{l_0}p^2} - \frac{q_D}{2^{l_0}p^5}\right).$$

□

The following lemma shows that our CLE scheme is secure against the Type-II adversary.

Lemma 2. *This CL-KEM is $(q_H, q_{par}, q_{pri}, q_{pub}, q_D, \varepsilon)$-IND-CCA secure against the Type-II adversary $\mathcal{A}$ in the random oracle model, then there exists an algorithm $\mathcal{B}$ that solves the CDH problem with the following advantage*

$$\varepsilon' > \frac{1}{q_{H_2}}\left(\frac{2\varepsilon}{e(q_{prv}+1)} - \frac{q_D q_{H_1}}{2^{l_0}p^2} - \frac{q_D}{2^{l_0}p^5}\right).$$

Proof. Assuming there exists an algorithm $\mathcal{A}_{II}$ who impersonates an "insider" adversary. Suppose that there is another PPT algorithm $\mathcal{B}$ can solve the CDH problem in the instance of (p, P, aP, bP) with probability at least ε' by interacting with $\mathcal{A}_{II}$. To solve this problem, $\mathcal{B}$ needs to simulate a challenger to run each algorithm of **Game-II** for $\mathcal{A}_{II}$ as follows:

Setup. Algorithm $\mathcal{B}$ picks the master secret key $x \in Z_p^*$ randomly and computes $X = xP$. Then $\mathcal{B}$ gives the system parameters $params = \{p, G, P, X, H_1, H_2\}$ to $\mathcal{A}_{II}$, where H_1 and H_2 are random oracles. Adversary $\mathcal{A}_{II}$ queries these two random oracles at any time during its attack. $\mathcal{B}$ responds as follows:

H_1 **Queries.** $\mathcal{B}$ maintains a list of tuples $\langle(\mathrm{ID}, r_{\mathrm{ID}}, Y), v\rangle$ in H_1-**List** L_1. On receiving a query $(\mathrm{ID}, r_{\mathrm{ID}}, Y)$ to H_1:

1) If $\langle(\mathrm{ID}, r_{\mathrm{ID}}, Y), v\rangle$ already appears on the list L_1, responds v as an answer.
2) Otherwise, pick $v \in Z_p^*$ randomly, add $\langle(\mathrm{ID}, r_{\mathrm{ID}}, Y), v\rangle$ to L_1 and return v as an answer.

H_2 **Queries.** $\mathcal{B}$ maintains a list of tuples $\langle(\mathrm{ID}, T), R\rangle$ in H_2-**List** L_2, where $T \in G^{*5}$. On receiving a query (ID, T) to H_2:

1) If $\langle(\mathrm{ID}, T), R\rangle$ exists in the list L_2, return R as an answer.
2) Otherwise, choose $R \in \{0,1\}^*$ uniformly at random, add $\langle(\mathrm{ID}, T), R\rangle$ to L_2 and return R as an answer.

Phase 1. $\mathcal{A}_{II}$ issues the following oracle queries.

Public-Key-Request-Oracle. $\mathcal{B}$ maintains a **PublicKeyList** of tuples $\langle\mathrm{ID}, (r_{\mathrm{ID}}, Y), coin\rangle$. On receiving a query ID, $\mathcal{B}$ responds as follows:

1) If $\langle\mathrm{ID}, (r_{\mathrm{ID}}, Y), coin\rangle$ exists in **PublicKeyList**, return $PK_{\mathrm{ID}} = (r_{\mathrm{ID}}, Y)$ as an answer.
2) Otherwise, pick $coin \in \{0,1\}$ at random so that $\Pr[coin = 0] = \delta$ (δ is the same as it in the proof of **Lemma 1**).
3) If $coin = 0$, choose $y, \alpha \in Z_p^*$ at random and compute $Y = yP$, $r_{\mathrm{ID}} = \alpha P$ and $z_{\mathrm{ID}} = \alpha + xH_1(\mathrm{ID}\|r_{\mathrm{ID}}\|Y)$. Then, add $\langle\mathrm{ID}, (y, z_{\mathrm{ID}}), \alpha\rangle$ to PrivateKeyList and $\langle\mathrm{ID}, (r_{\mathrm{ID}}, Y), coin\rangle$ to PublicKeyList respectively, return $PK_{\mathrm{ID}} = (r_{\mathrm{ID}}, Y)$ as an answer.

4) Otherwise (if $coin = 1$), pick $\alpha, y \in Z_p^*$ at random and compute $r_{\mathrm{ID}} = \alpha aP$, $Y = yP$ and $z_{\mathrm{ID}} = \alpha + bxH_1(\mathrm{ID}\|r_{\mathrm{ID}}\|Y)$. Then, add $\langle \mathrm{ID}, (y, *), \alpha \rangle$ to **PrivateKeyList** (where * denotes the arbitrary value), and $\langle \mathrm{ID}, (r_{\mathrm{ID}}, Y), coin \rangle$ to **PublicKeyList**, return $PK_{\mathrm{ID}} = (r_{\mathrm{ID}}, Y)$ as an answer.

Private-Key-Request-Oracle. $\mathcal{B}$ maintains a **PrivateKeyList** of tuples $\langle \mathrm{ID}, (y, z_{\mathrm{ID}}), \alpha \rangle$. On receiving a query ID, $\mathcal{B}$ responds as follows:

1) Perform *Public-Key-Request-Oracle* on ID to get a tuple $\langle \mathrm{ID}, (r_{\mathrm{ID}}, Y), coin \rangle$ from **PublicKeyList**.
2) If $coin = 0$, search **PrivateKeyList** for a tuple $\langle \mathrm{ID}, (y, z_{\mathrm{ID}}), \alpha \rangle$ and return $SK_{\mathrm{ID}} = (y, z_{\mathrm{ID}})$ as an answer.
3) Otherwise, return "*Abort*" and terminate.

Decapsulation-Oracle. On receiving a query $\langle$ID, PK_{ID}, $c\rangle$, where $PK_{\mathrm{ID}} = (r_{\mathrm{ID}}, Y)$. $\mathcal{B}$ responds as follows:

1) Search a tuple $\langle \mathrm{ID}, (r_{\mathrm{ID}}, Y), coin \rangle$ in **PublicKeyList**. If such a tuple exists and $coin = 0$, search a tuple $\langle \mathrm{ID}, (y, z_{\mathrm{ID}}) \rangle$ in **PrivateKeyList** (Note that $\langle \mathrm{ID}, (r_{\mathrm{ID}}, Y), coin \rangle$ must exist in **PublicKeyList**. While $coin = 0$, the tuple $\langle \mathrm{ID}, (y, z_{\mathrm{ID}}), \alpha \rangle$ exists in **PrivateKeyList**). Then, set $SK_{\mathrm{ID}} = (y, z_{\mathrm{ID}})$ and run the algorithm of *Decap*. Finally, return the results of the *Decap*.
2) Otherwise (if $coin = 1$), run H_1 **queries** to access a tuple $\langle (\mathrm{ID}, r_{\mathrm{ID}}, Y), v \rangle$. If there exists $\langle (\mathrm{ID}, T), R \rangle \in L_2$ such that $R = H_2(\mathrm{ID}\|T)$, return R as the session key and "*Reject*" otherwise.

Challenge Phase. Once $\mathcal{A}_{II}$ decides that *Phase 1* is over, it outputs a challenge identity ID*. On receiving a challenge query ID*, $\mathcal{B}$ responds as follows:

1) Taking ID* as input, $\mathcal{B}$ runs *Public-Key-Request-Oracle* and gets a tuple $\langle \mathrm{ID}^*, (r_{\mathrm{ID}^*}, Y^*), coin \rangle$ from **PublicKeyList**.
2) If $coin = 0$, return "*Abort*" and terminate.
3) Otherwise, do the following:
   a. Search for a tuple $\langle \mathrm{ID}^*, (y^*, z_{\mathrm{ID}^*}), \alpha^* \rangle$ from **PrivateKeyList** (In this case, we know that $r_{\mathrm{ID}^*} = \alpha^* aP$, $Y^* = y^* P$).
   b. Set $c^* = aP$, $c_1^* = a(Y^* + r_{\mathrm{ID}^*} + XH_1(\mathrm{ID}^* \| r_{\mathrm{ID}^*} \| Y^*))$ and $c_2^* = aY^*$. Also, note that $\mathcal{B}$ does not know "a". Then compute $v^* = H_1(\mathrm{ID}^* \| r_{\mathrm{ID}^*} \| Y^*)$.
   c. Pick $K_0 \in_R \mathcal{K}$, where $\mathcal{K}$ is the key space.
   d. Compute $K_1 = H_2(\mathrm{ID}^* \| (r_{\mathrm{ID}^*}, Y^*) \| c^* \| c_1^* \| c_2^*)$.
4) Choose a bit $\beta \in_R \{0, 1\}$ and return (c^*, K_β) to $\mathcal{A}_{II}$.

Phase 2. $\mathcal{A}_{II}$ repeats the same methods as in *Phase 1*. Moreover, no private key extraction on ID* is allowed and no *Decapsulation-Oracle* query should be made on the encapsulation c^* for ID*.

Guess. $\mathcal{A}_{II}$ outputs a guess β' for β, and wins the game if $\beta' = \beta$. Then, $\mathcal{B}$ enables to solve the CDH problem by computing $(c^* \cdot z_{\mathrm{ID}^*} - r_{\mathrm{ID}^*})/(x \cdot v^*)$.

Analysis. Similar to Analysis in the proof of **Lemma 1**.

Consequently, we obtain

$$\varepsilon' > \frac{1}{q_{H_2}}\left(\frac{2\varepsilon}{e(q_{prv}+1)} - \frac{q_D q_{H_1}}{2^{l_0} p^2} - \frac{q_D}{2^{l_0} p^5}\right).$$

□

In conclusion, based on these two lemmas, we complete the proof of **Theorem 1**.

4 Comparisons

In this section, we compare our CL-KEM with previous protocols [7, 10, 12] on the computation complexity of encapsulation (**Enc**) and decapsulation (**Dec**), the bandwidth of the encapsulation (**Bandwidth**) and the running time (**Time**) of one-round *Encap-Decap* of each scheme. Without considering the addition of two points, hash function and exclusive-OR operations, we denote the cost of a bilinear pairing by P, the cost of an exponentiation by E , and the cost of a scalar multiplication in the additive cyclic group by S.

This CL-KEM is tested on a laptop with the Intel Core i5-2400 at a frequency of 3.10 GHz processor, 3GB memory and Ubuntu-12.04 operation system, using the pairing based cryptography (PBC) library (version 0.5.13 [13]). The implementation takes use of a 160-bit elliptic curve group based on the supersingular curve $y^2 = x^3 + x$ over a 512-bit finite field with embedding degree 2. Then, the average running time of each operation is obtained and demonstrated in Table 2.

Table 2: Cryptographic operation time

Pairing	Exponentiation	Scalar multiplication
3.93 ms	3.35 ms	3.28 ms

As to communication cost, we analyze it in terms of bandwidth of transmitting encapsulation. Suppose that the output of one way Hash function is 160-bit, and the elements of multiplicative group is 1024-bit (e.g., parameters in RSA). In our protocol, one encapsulation contains one point, thus the bandwidth of our protocol is $160/8 = 20$ bytes. In [7, 10], each encapsulation contains two exponentiations, thus the bandwidths of [7, 10] are

$(1024 \times 2)/8 = 256$ bytes respectively. At last, in Lippold et al.'s scheme [12], the encapsulation contains two exponentiations and one hash value, the bandwidth of it is $(1024 \times 2 + 160)/8 = 276$ bytes. The detailed results are listed in Table 3, and the bandwidth of our scheme is the smallest one.

Table 3: Comparison of the related schemes

Schemes	Enc	Dec	Bandwidth	Time
[10]	4E	2E	256 bytes	20.10 ms
[12]	5E	3P+6E	276 bytes	48.64 ms
[7]	4E	2E	256 bytes	20.10 ms
Ours	4S	2S	20 bytes	19.68 ms

The computation and communication cost in this scheme is less than others, which shows that our scheme enables to provide an efficient method to protect the confidential of the session key between patient and doctor in e-healthcare system.

5 Conclusions

We have proposed an efficient certificateless key encapsulation mechanism for e-healthcare system to protect the confidentiality of the session key in the hybrid encryption scheme. In terms of security, we prove that this scheme is IND-CCA secure in the random oracle model assuming that CDH problem is intractable. Furthermore, our protocol promotes the trust hierarchy of the medical server to the Girault's trust Level 3. A thorough performance evaluation and experiments on PC indicate that the proposal is advantageous over the related schemes in efficiency. Thus, all these attributes render this scheme a promising approach in session key protection to e-healthcare system.

Acknowledgments

This work was supported by National Natural Science Foundation of China (Grant Nos. 11171200, 11426148) and Fundamental Research Funds for the Central Universities (Grant No. GK201402006).

References

[1] M. Abe, R. Gennaro, and K. Kurosawa, "Tag-kem/dem: A new framework for hybrid encryption," *Journal of Cryptology*, vol. 21, no. 1, pp. 97–130, 2008.

[2] S. S. Al-Riyami and K. G. Paterson, "Certificateless public key cryptography," in *Advances in Cryptology (ASIACRYRT03)*, pp. 452–473, Taipei, Taiwan, Nov. 2003.

[3] K. Bentahar, P. Farshim, J. M. Lee, and N. P. Smart, "Generic constructions of identity-based and certificateless KEMs," *Journal of Cryptology*, vol. 21, no. 1, pp. 178–199, 2008.

[4] J. Daemen and V. Rijmen, *Advanced Encryption Standard (AES)*, Technical Report 197, Dec. 2001.

[5] A. Dent, "A designers guide to KEMs," in *Cryptography and Coding*, pp. 133–151, Cirencester, UK, Dec. 2003.

[6] M. Girault, "Self-certificated public keys," in *Advances in Cryptology (EUROCRYPTO91)*, pp. 34–46, Brighton, UK, Apr. 2010.

[7] D. Hofheinz and E. Kiltz, "Secure hybrid encryption from weakened key encapsulation," in *Advances in Cryptology (CRYPTO07)*, pp. 553–571, California, USA, Aug. 2007.

[8] Q. Huang and D. S. Wong, "Generic certificateless key encapsulation mechanism," in *Information Security and Privacy*, pp. 215–229, Townsville, Australia, July 2007.

[9] E. Kiltz, "Chosen-ciphertext secure key-encapsulation based on gap hashed diffie-hellman," in *Public Key Cryptography (PKC'07)*, pp. 282–297, Beijing, China, Apr. 2007.

[10] K. Kurosawa and Y. Desmedt, "A new paradigm of hybrid encryption scheme," in *Advances in Cryptology (CRYPTO04)*, pp. 426–442, California, USA, Aug. 2004.

[11] Congress Public Law, *Health Insurance Portability and Accountability Act of 1996*, Technical Report 104, June 1996.

[12] G. Lippold, C. Boyd, and J. M. G. Nieto, "Efficient certificateless KEM in the standard model," in *Information, Security and Cryptology*, pp. 34–46, Seoul, Korea, Dec. 2010.

[13] B. Lynn, *The Pairing-based Cryptography Library*, PBC Library, May 2015. (http://crypto.stanford.edu/pbc/)

[14] J. H. Oh, K. K. Lee, and S. J. Moon, "How to solve key escrow and identity revocation in identity based encryption scheme," in *Proceedings of 1st International Conference on Information System Security*, pp. 290–303, Kolkata, India, Dec. 2005.

[15] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Advances in Cryptology-CRYPTO84*, pp. 47–53, California, USA, Dec. 1985.

[16] W. Tuchman and C. Meyer, *Date Encryption Standard (DES)*, Technical Report 46, July 1977.

[17] M. K. Watfa, *E-healthcare Systems and Wireless Communications: Current and Future Challenges*, Technical Report 27, Sep. 2012.

Shi Hui-Xian: received the B.S. and Ph.D degrees in Department of Mathematics and Information Science from Shaanxi Normal University, Xi'an, China, in 2007 and 2013, respectively. Now she is a post-doctoral in Department of Computer Science in Shaanxi normal University. Her present research interests include model

checking, fuzzy logic and uncertainty reasoning.

Guo Rui: received the BS degree in Math and Applied Math from Henan University of Science and Technology, the MS degree in Applied Math from Shaanxi Normal University. He is currently a candidate for Ph.D in the Department of State Key Laboratory of Networking and Switch Technology, Beijing University of Posts and Telecommunications. His present research interests include cryptography, information security and applied mathematics.

A Study of DWT-SVD Based Multiple Watermarking Scheme for Medical Images

Natarajan Mohananthini and Govindarajan Yamuna
(Corresponding author: Natarajan Mohananthini)
Department of Electrical Engineering, Annamalai University
Annamalainagar, Tamilnadu 608 002, India
(Email: mohananthini@yahoo.co.in)
(Received Dec. 12, 2014; revised and accepted Mar. 2 & Apr. 27, 2015)

Abstract

Medical images may contain sensitive information and additional security measures are essential to preserve the patient's privacy. The multiple watermarking is projected to increase the security of medical images and to preserve the privacy of patients. In this paper a multiple medical image watermarking scheme based on discrete wavelet transform (DWT) and singular value decomposition (SVD) is presented. In the proposed method, three watermarks are embedded into different channel (R, G and B) of color images such as the first watermark is patient identification, the second watermark is patient diagnosis information and the third watermark is doctor signature image. From the experimental results, the proposed method is robust to common image processing attacks and good performance in terms of imperceptibility on different types of medical images.

Keywords: Discrete wavelet transform, medical images, multiple watermarking, singular value decomposition

1 Introduction

Digital image watermarking is the process of embedding information into digital image such that the information can later be extracted for a variety of purposes [24]. Now-a-days medical image watermarking is a popular research area and the important applications of watermarking. The uses of advanced electronic and digital equipments in health care services are increased in the electronic and digital data such as, Computed Tomography (CT), Magnetic Resonance Imaging (MRI), X-ray images and Ultrasonic image.

The medical information records are focused, which for a complex set of clinical examinations, diagnosis observations and other findings information in its Electronic Patient Records (EPR) [6]. The three mandatory security characteristics are confidentiality, availability and integrity. Confidentiality means that only the authorized users have access to the information. Availability means the ability of an information system to be used in the normal scheduled conditions of access. Integrity means the information has not been modified by non-authorized user.

In medical image watermarking, the medical images are embedded with hidden information that may be used to assert ownership, increase the security, and verify the numerical integrity of medical images [2]. Zheng et al. [27] reviewed the algorithms for rotation, scaling and translation (RST) invariant image watermarking. There are mainly two categories of RST invariant image watermarking algorithms. One is to rectify the RST transformed image before conducting watermark detection. Another is to embed and detect watermark in an RST invariant or semi-invariant domain. In order to help readers understand, their first introduced the fundamental theories and techniques used in the existing RST invariant image watermarking algorithms and then discussed in detail the work principles, embedding process, and detection process of the typical RST invariant image watermarking algorithms.

Coatrieux et al. [1] focused on the complementary role of watermarking with respect to medical information security (integrity, authenticity) and management. Their reviewed sample cases where watermarking has been deployed and concluded that watermarking has found a niche role in healthcare systems, as an instrument for protection of medical information, for secure sharing and handling of medical images.

2 A Brief Literature Survey

A brief literature survey of image watermarking and some recent researches is presented here. Yamuna and Sivakumar [26] proposed a novel watermarking scheme for copyright protection in digital images. The watermarking is performed in wavelet domain using bi-orthogonal wavelet transform and their approach is non-blind, it requires

original image for extracting the watermark. The experimental results demonstrate that the watermark with their proposed algorithm satisfied imperceptibility. The watermarking algorithm is based on Discrete Wavelet Transform (DWT) and Singular Value Decomposition (SVD) in [13]. A new audio signal framing, DWT matrix formation and embedding methods are proposed and successfully implemented to improve the quality of watermarked audio. The significance of their features makes the system robust to common signal processing operations.

Mehta et al. [11] studied the performance of three different watermarking algorithms (DWT, SVD and DWT-SVD based watermarking algorithms). They have created two different watermarks one is QR Code watermark image, which is capable of carrying large information in small space and other one is a normal text image watermark. Experimental results show that DWT based method is suitable for medical applications where embedding time and imperceptibility are prime concerns while SVD based methods are suitable for medical applications where robustness and capacity are the main concerns. Khan et al. [9] presented a hybrid digital image watermarking based on Discrete Wavelet Transform (DWT), Discrete Cosine Transform (DCT) and Singular Value Decomposition (SVD). To embed the watermark in high band that facilities to add more information, gives more invisibility and robustness against some attacks. Sleit et al. [20] presented a semi-blind hybrid watermarking technique based on singular value decomposition (SVD) and discrete wavelet transformation (DWT). Their proposed technique decomposes the host image using DWT and combines the singular values (SVs) of the watermark and the selected sub-bands. Experimental results show that their proposed technique is able to resist a variety of attacks.

Navas et al. [18] proposed a method of non-blind transform domain watermarking based on DWT-DCT-SVD. The DCT coefficients of the DWT coefficients are used to embed the watermarking information. Their method of watermarking is found to be robust and the visual watermark is recoverable without only reasonable amount of distortion even in the case of attacks. Thus their method can be used to embed copyright information in the form of a visual watermark or simple text. Experiments conducted on four types of medical images proved that their modifications are visually imperceptible while it has a good robustness against some common attacks such as compression, filtering, and noise.

Dhaliwal et al. [3] presented a comparative study of single watermarking to multiple watermarking over a color image. Li et al. [10] proposed a novel multiple watermarking algorithm which embedded two watermarks into original image in different frequency by using bandelet transform. Experimental results demonstrate that their watermarking algorithm based on bandelet transform has a good performance both in invisibility and robustness. Radharani et al. [19] presented the concept of multiple watermarking is used to hide both copyright and authentication information into a color image. Experimental results indicate that their proposed watermarking scheme is highly robust and does not degrade the original signal. A watermarking technique is proposed in [14], to directly embed multiple watermarks into a single color image. The watermark embedding process the multiple watermarks are embedded into original image and the extraction processes recover the watermarks from the watermarked image. The experimental results have shown their scheme has preferable performance of imperceptibility. A digital image multiple successive watermarking scheme based on wavelet transform is proposed in [15]. In their method, the PSNR and image quality are degraded with every one new watermark embedded into image, and the watermarked images are tested for non-geometric attacks such as less robustness of salt and pepper noise and Gaussian noise and more robustness of median filtering, sharpening, smoothing and JPEG compression.

Successive and segmented watermarking techniques are proposed in [17]. The embedding and extraction process using multi-resolution analysis of wavelet transform for color images. The successive watermarking technique, the watermarks are embedded one after the other and the segmented watermarking technique, one watermark is embedded into odd-numbered rows and another watermark is embedded into even-numbered rows. Their segmented watermarking vividly shows better visual quality on watermarked image when compared with successive watermarking. The different embedding methods of single and multiple watermarking are compared using discrete wavelet transform in [16]. The different embedding methods are additive, multiplicative and hybrid watermarking with importance on its robustness versus the imperceptibility of the watermark. The objective quality metrics are demonstrated that, their additive embedding method achieves superior performance against watermark attacks on multiple watermarking technique.

Woo et al. [25] proposed a multiple digital image watermarking method which is suitable for privacy control and tamper detection in medical images. The annotation watermark can be detected in a blind manner, that is the original un-watermark image is not required to detect the annotated watermark. The effectiveness of the fragile part in tamper detection has been proved some general image manipulation attacks. Giakoumaki et al. [5] presented the perspectives of digital watermarking in health information systems, and proposes a wavelet-based multiple watermarking schemes that address the issues of medical data protection, archiving, and retrieval, as well as of origin and data authentication. The experimental results demonstrate the efficiency and transparency of their watermarking scheme, which conforms to the strict limitations that apply to regions of diagnostic significance.

Giakoumaki et al. [4] discussed the perspectives of digital watermarking in a range of medical data management and distribution. Their scheme imperceptibly embeds in medical images multiple watermarks conveying patient's personal and examination data, keywords for information

retrieval, the physician's digital signature for authentication, and a reference message for data integrity control. Experimental results indicate the efficiency and transparency of their scheme, which conforms to the strict requirements that apply to regions of diagnostic significance. Kallel et al. [8] applied a multiple watermarking technique in the wavelet field to preserve the traceability and the record of the medical image diagnosis.

Rathi et al. [21] focused on the study of medical image watermarking methods for protecting and authenticating medical data. The medical images can be transferred securely by embedding watermarks in Region of Non Interest (RONI) allowing verification of the legitimate changes at the receiving end without affecting Region of Interest (ROI). The experimental results show the satisfactory performance of their system to authenticate the medical images preserving ROI. Irany et al. [7] proposed a high capacity reversible multiple watermarking scheme for medical images based on integer-to-integer wavelet transform and histogram shifting. The novelty of their proposed scheme is that it uses a scalable location map and incorporates efficient stopping conditions on both wavelet levels and different frequency sub bands of each level to achieve high capacity payload embedding, high perceptual quality, and multiple watermarking capabilities.

Memon et al. [12] proposed a multiple watermarking scheme, embeds robust watermark in region of non interest (RONI) for achieving security and confidentiality, while integrity control is achieved by inserting fragile watermark in region of interest ROI. ROI in the medical image is important from diagnosis point of view so it must be preserved. The image visual quality as well as tamper localization has been evaluated. Sujatha et al. [22] proposed an innovative watermarking scheme, in which low frequency subband of wavelet domain and the rescaled version of original image are utilized in the watermark generation process. A watermark hiding scheme for copyright protection of sensitive images is proposed in [23].

In analysis of various literatures the medical image watermarking is found to be imperceptibility, robust, high security and verify the integrity of medical images. In this paper a study of discrete wavelet transform and singular value decomposition based multiple watermarking schemes for medical images. The multiple information's are embedded into the medical image for patient's privacy.

In the extraction process multiple information are recovered from the watermarked image. Experimental results demonstrate that, the proposed method achieves high security, imperceptibility and robustness against attacks for three different types of medical images.

3 Proposed Scheme

The proposed scheme focuses on medical image watermarking methods for protecting and authenticating the medical image. Medical images information related to the patient's health condition is stored separately either in digital documents or images. The embedded information of medical images is exchanged from hospitals to required area. Also, as this exchange of medical embedded information done through unsecured open networks leads to the condition of changes to occur in medical images and creates a threat which results in undesirable outcome. Considering this fact, increase the security of medical images due to easy reproduction and used for further diagnosis and treatment. The block diagram of proposed medical image watermarking is shown in Figure 1.

In the proposed scheme, there are two significant phases: watermark embedding and watermark extraction. The flowcharts for watermark embedding and extraction process are shown in Figure 2 and Figure 3. The Discrete Wavelet Transform (DWT) and Singular Value Decomposition (SVD) are used in watermark embedding and extraction process. DWT is a mathematical tool for hierarchical decomposing of an image into two level subbands. SVD is an effective numerical tool to analyze the matrices (USVT) which is of the same size as the original matrix. The watermark embedding and extraction process are discussed as follows,

3.1 Watermark Embedding Process

1) The original image, first watermark, second watermark and third watermark are separated into three components of Red (R), Green (G) and Blue (B).

2) The R, G and B component of original image are decomposed by two levels using discrete wavelet transforms.

3) The SVD is applied to R, G and B component of LL2 sub-bands of original image and watermark images.

4) The singular value of first watermark of R component, second watermark of G component and third watermark of B component are embedded into singular value of R, G and B component of original image, by using the following equations:

$$\begin{aligned} IR_W(i,j) &= SIR(i,j) + \alpha \times SW1(i,j); \\ IG_W(i,j) &= SIG(i,j) + \alpha \times SW2(i,j); \\ IB_W(i,j) &= SIB(i,j) + \alpha \times SW3(i,j). \end{aligned}$$

Where, $IR_W(i,j)$, $IG_W(i,j)$, and $IB_W(i,j)$ denote red, green, and blue component of watermarked image, respectively. $SIR(i,j)$, $SIG(i,j)$, and $SIB(i,j)$ denote singular values of red, green, and blue component of original image, respectively. $SW1(i,j)$, $SW2(i,j)$, and $SW3(i,j)$ denote singular values of first, second, and third watermark, respectively. α denotes a scaling factor.

5) The inverse SVD is applied and inverse wavelet transform is performed to get the R, G and B component of watermarked image.

6) R, G and B component of watermarked image are merged to get the watermarked image.

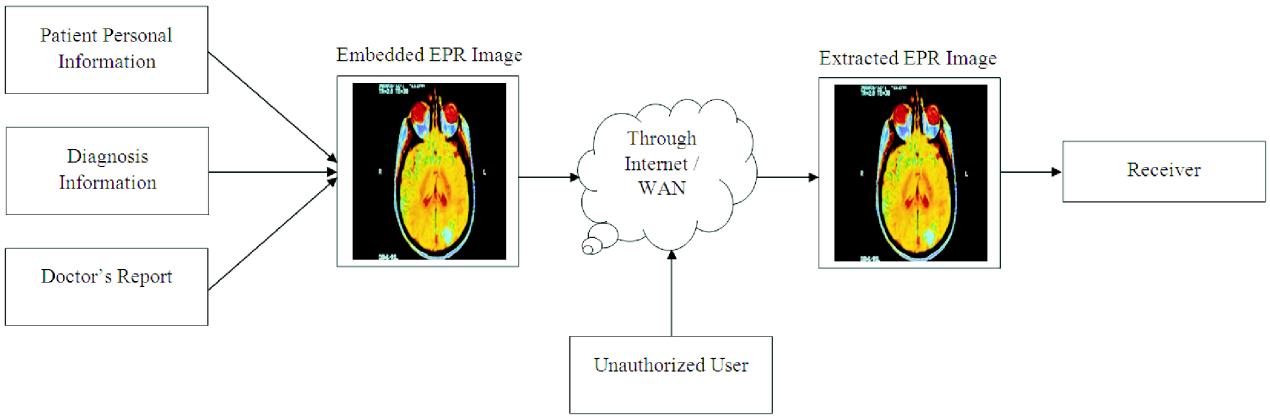


Figure 1: Block diagram of proposed medical image watermarking

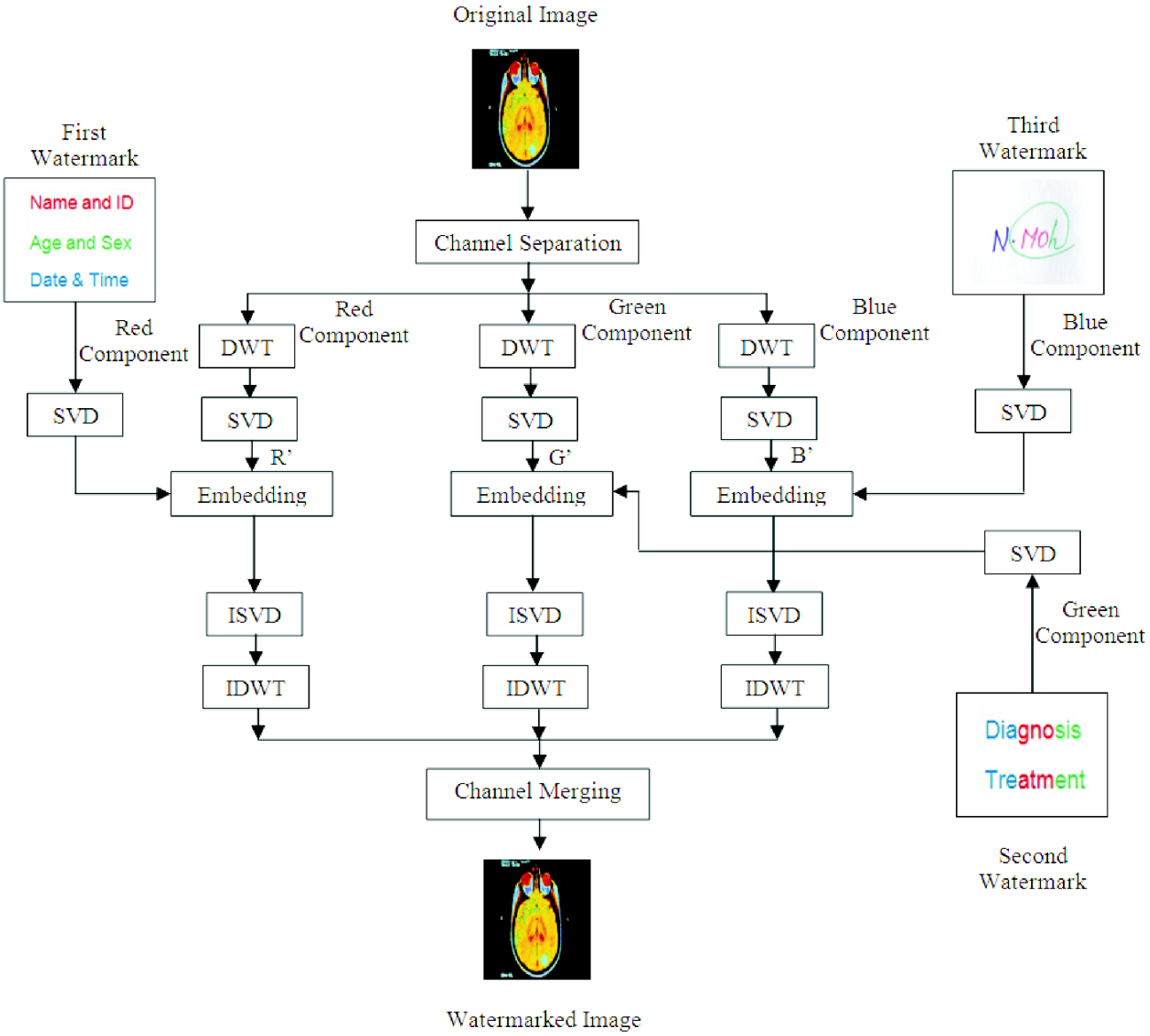


Figure 2: Flowchart for watermark embedding process

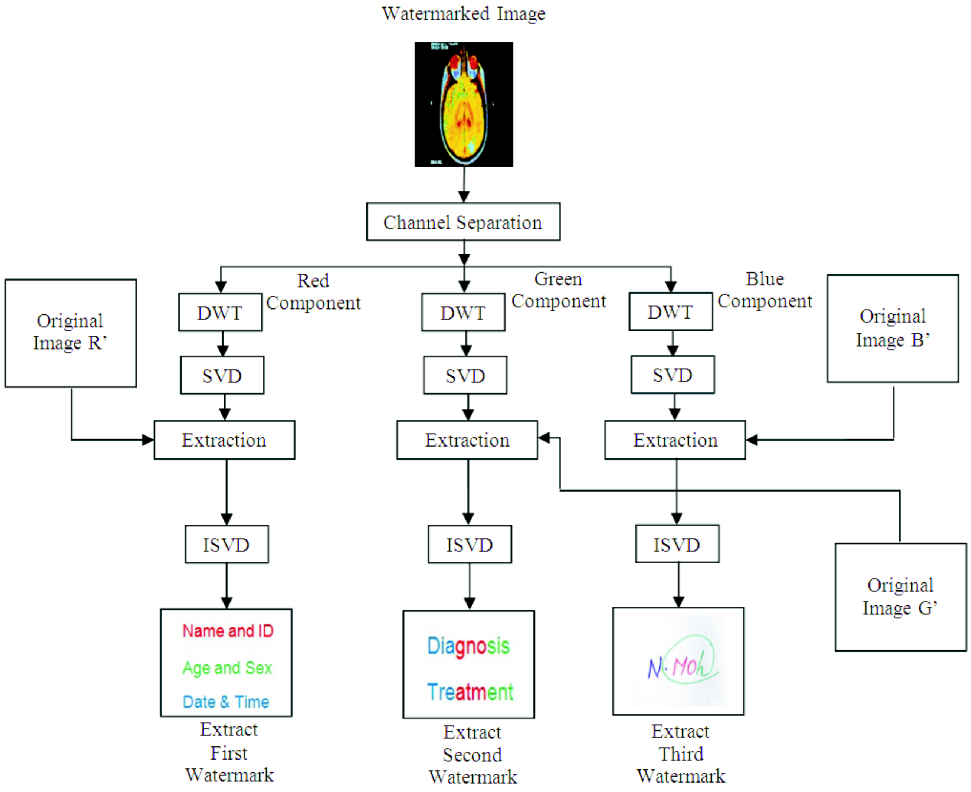


Figure 3: Flowchart for watermark extraction process

3.2 Watermark Extraction Process

1) The watermarked image and original image is separated into three components of Red (R), Green (G) and Blue (B).

2) The R, G, B component of watermarked image and original image is decomposed by two levels by using discrete wavelet transforms.

3) The SVD is applied to R, G and B component of LL2 sub- bands of watermarked image and original image.

4) The singular values of first, second and third watermark can be extracted as,

$$SW1(i,j) = \frac{IR_W(i,j) - SIR(i,j)}{\alpha};$$
$$SW2(i,j) = \frac{IG_W(i,j) - SIG(i,j)}{\alpha};$$
$$SW3(i,j) = \frac{IB_W(i,j) - SIB(i,j)}{\alpha}.$$

5) The inverse SVD is applied to extract first, second and third watermark.

4 Results and Performance Analysis

In this paper, a robust multiple watermarking scheme is proposed based on wavelet domain for medical images. The different type of medical images were used in the watermarking experiment. Figure 4 shows the different type of test images such as Magnetic Resonance Imaging (MRI), Computed Tomography (CT) and X-ray images. The watermarks are used in medical image watermarking to address the issues of medical information security. The watermarks used in the proposed scheme.

- Patient's identification number;
- Patient name;
- Patient age;
- Patient sex;
- Patients diagnosis information;
- Patient treatment information;
- Doctor signature.

These listed watermarks are used in the proposed watermarking scheme helps in addressing different problems and extracting the original image without any distortion.

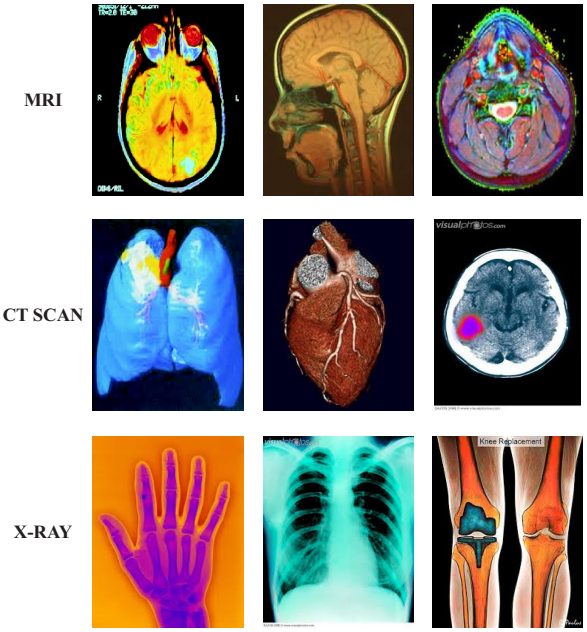


Figure 4: Different medical images

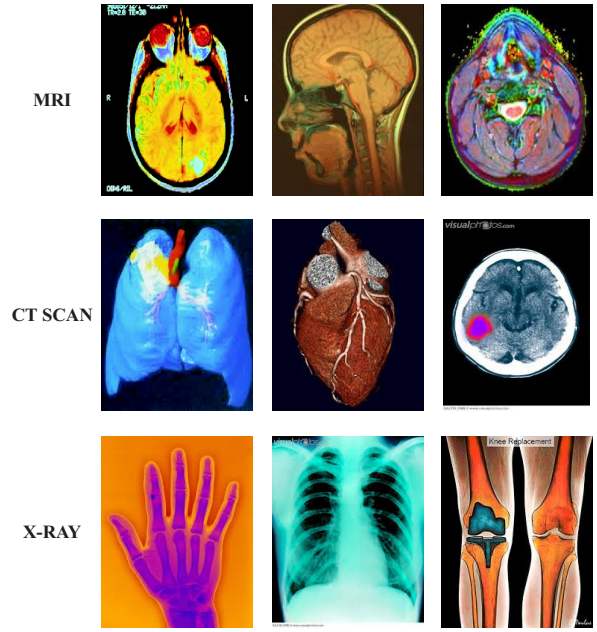


Figure 6: Watermarked images

Figure 5: (a) First watermark (b) Second watermark (c) Third watermark

Patient's identification number, name, age, sex, date and time are used in first watermark, patients diagnosis and treatment information are used in second watermark and doctor signature are used in third watermark as shown in Figure 5.

4.1 Imperceptibility Assessment

The performance of watermarking technique can be evaluated using the two common quantitative indices such as peak signal to noise (PSNR) and the normalized correlation (NC) are used. Figure 6 illustrates the watermarked images of MRI, CT scan and X-ray images. Table 1 shows the PSNR and NC values of multiple watermarking with different medical images. Peak Signal to Noise Ratio (PSNR) is used to measure quality of watermarked image, it is given by

$$PSNR(dB) = 10log_{10}\frac{255^2}{MSE}.$$

The Mean Square Error (MSE) between a watermarked image and cover image.

$$MSE = \frac{1}{N}\sum_{j=0}^{N}(I_w - I)^2,$$

where I_w is watermarked image and I is cover image.

Normalized Cross Correlation is used to measure the quality of watermark after recovery. The NC between the embedded watermark W (i, j) and the extracted watermark W (i, j) is defined as

$$NC = \frac{\sum_{i=1}^{H}\sum_{j=1}^{L}W(i,j)\times W'(i,j)}{\sum_{i=1}^{H}\sum_{j=1}^{L}[W(i,j)]^2}.$$

4.2 Comparison to Existing Method

To prove the effectiveness of proposed scheme, the imperceptibility (PSNR) value is compared with existing scheme [11]. Their existing scheme DWT SVD based watermarking algorithm and two watermarks are embedded into medical image. The PSNR values are listed in Table 2, and it is evident that the imperceptibility performance of the proposed scheme is superior to existing scheme for the medical image. The medical images shown in Figure 7 for compared to existing method.

4.3 Robustness Assessment

To prove the robustness, the watermarked images are tested for the different attacks such as salt and pepper noise, Gaussian noise, median filtering, Gaussian blur, translation, rotation, JPEG compression, histogram equalization, sharpening, smoothing, and Intensity transformation.

Table 1: PSNR and NC values of multiple watermarking with different medical images

Images	Watermarked Image (PSNR)	Extracted watermark		
		First	Second	Third
MRI	24.083	1	1	1
	23.423	1	1	1
	24.091	1	1	1
CT	29.420	1	1	1
	28.318	1	1	1
	23.961	0.997	0.998	0.999
X-ray	31.852	1	1	1
	28.023	1	1	1
	28.371	1	1	1

Table 2: Comparison to existing method

Images	Existing scheme PSNR (dB) [11]		Proposed scheme PSNR (dB)
	Watermark1	Watermark2	
1	21.25	21.02	26.4906
2	21.81	21.67	23.0892
3	21.97	21.69	24.2355
4	22.03	21.82	28.1566
5	26.93	27.02	29.1094

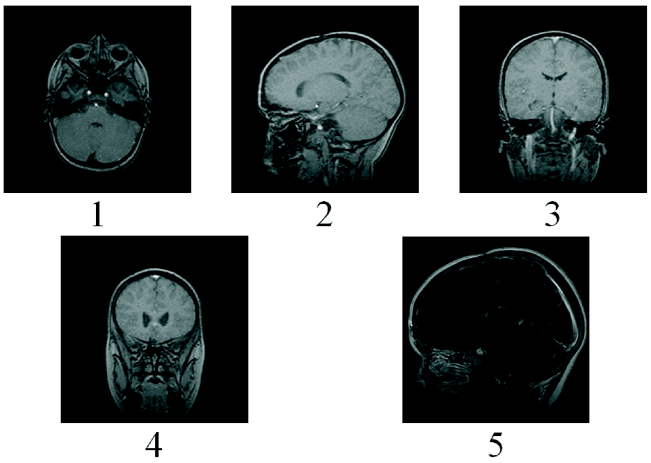


Figure 7: Medical images

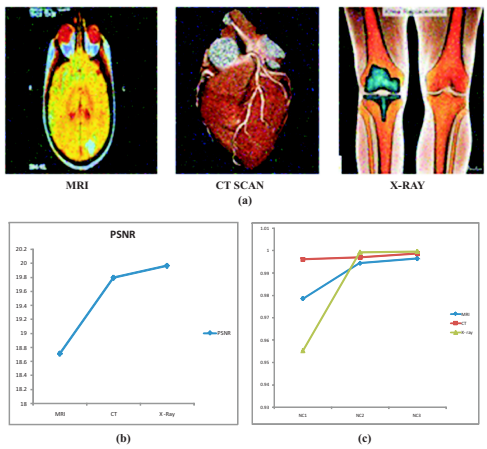


Figure 8: (a) Watermarked image with salt and pepper noise attacks (b) PSNR values (c) NC values

4.3.1 Salt and Pepper Noise

To test the robustness against adding noise, the watermarked image is degraded by adding salt and pepper noise at the density ranging from 0.01 to 0.1. Here the watermarked image is corrupted with salt and pepper noise at the density of 5%. The noise is usually quantified by the percentage of pixels which are corrupted. Figure 8 (a) shows the watermarked image, Figure 8 (b) and (c) plot of PSNR and NC values for medical images with salt and pepper noise attacks.

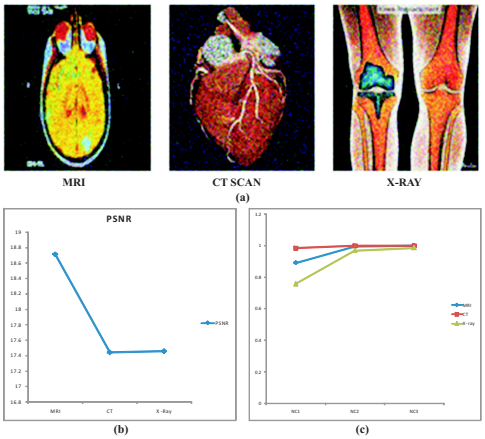


Figure 9: (a) Watermarked image with Gaussian noise attacks (b) PSNR values (c) NC values

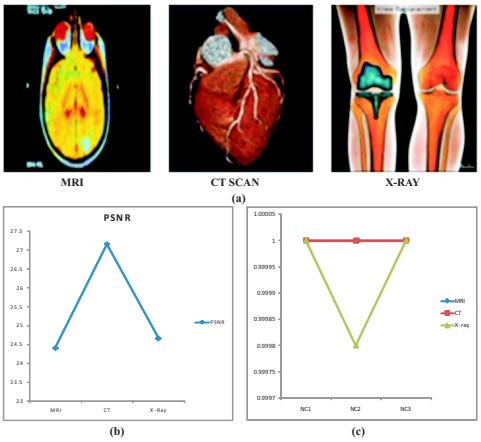


Figure 10: (a) Watermarked image with median filtering attacks (b) PSNR values (c) NC values

4.3.2 Gaussian Noise

The watermarked image corrupted with Gaussian noise of zero mean and varying the variance of the noise range is 0.5. Figure 9 (a) shows the watermarked image,

4.3.3 Median Filtering

Median filtering is a nonlinear operation often used in image processing to reduce high frequency noise in an image. For median filtering is 3×3 mask consisting of 0.05 intensity values is used to reduce noise in image. Median filtering is very widely used in digital image processing because, under certain conditions, it preserves edges while removing noise. Figure 10 (a) shows the watermarked image, Figure 10 (b) and (c) plot of PSNR and NC values for medical images with median filtering attacks.

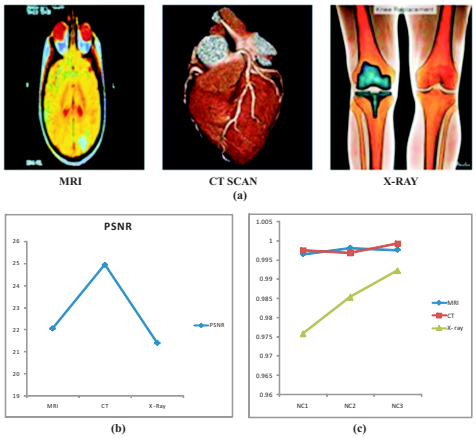


Figure 11: (a) Watermarked image with Gaussian blurring attacks (b) PSNR values (c) NC values

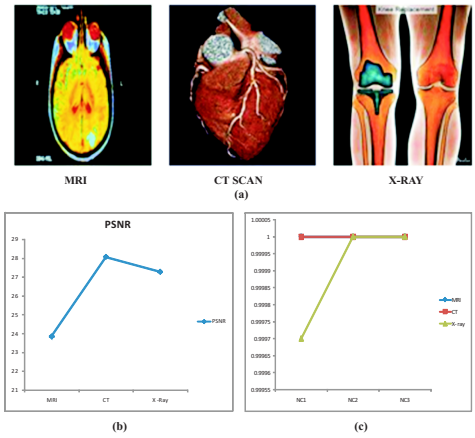


Figure 12: (a) Watermarked image with JPEG compression attacks (b) PSNR values (c) NC values

4.3.4 Gaussian Blur

A Gaussian blur is also known as Gaussian smoothing. It is the result of blurring an image by a Gaussian function. It is a widely used effect in graphics software, typically to reduce image noise. Figure 11 (a) shows the watermarked image, Figure 11 (b) and (c) plot of PSNR and NC values for medical images with Gaussian blurring attacks.

4.3.5 JPEG compression with Quality of 50

The JPEG is one of the most used image compression technique, and is often an unintentional attack. The watermarked images are compressed using different quality factor ranging from 0 to 100. JPEG compression is used in a number of image file formats. The watermarked images are compressed with quality factor 50. Figure 12 (a) shows the watermarked image, Figure 12 (b) and (c) plot of PSNR and NC values for medical images with JPEG compression attacks.

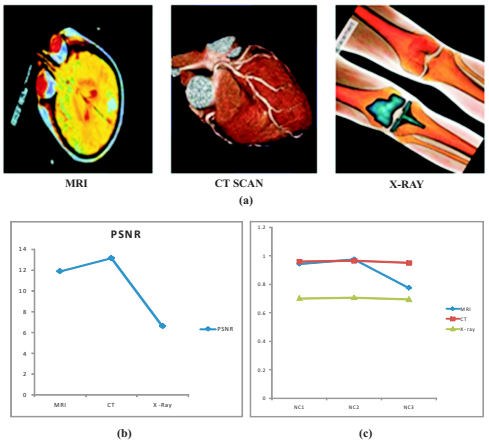


Figure 13: (a) Watermarked image with rotation attacks (b) PSNR values (c) NC values

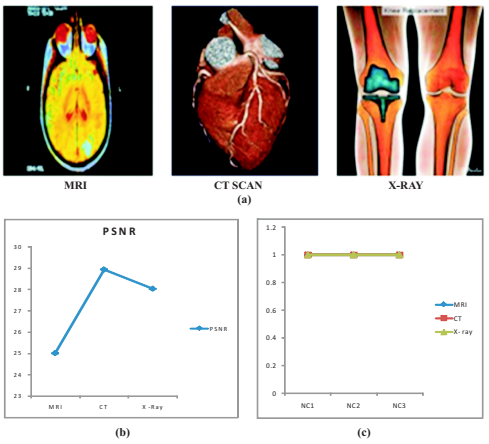


Figure 15: (a) Watermarked image with smoothing attacks (b) PSNR values (c) NC values

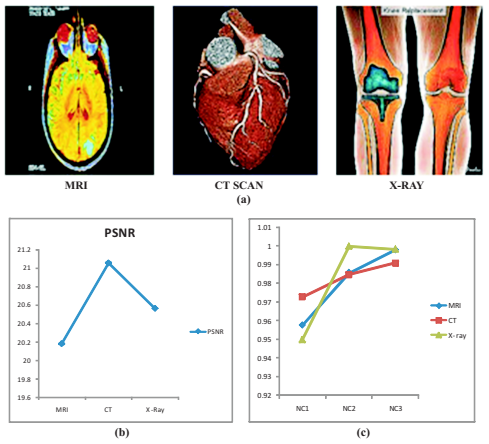


Figure 14: (a) Watermarked image with sharpening attacks (b) PSNR values (c) NC values

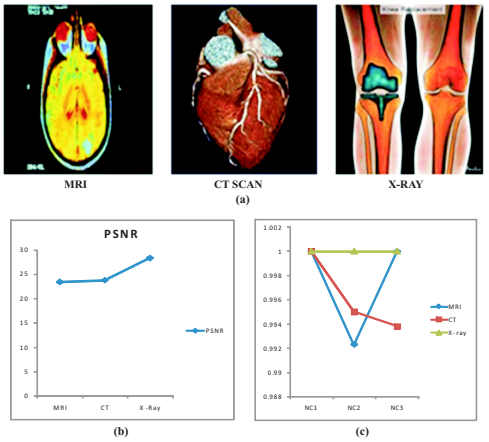


Figure 16: (a) Watermarked image with intensity transformation attacks (b) PSNR values (c) NC values

4.3.6 Rotation

The rotation is used to realign horizontal features of an image. Rotation is tested by rotating the image in counter-clockwise direction and then back to the original position through bilinear interpolation before watermark detection. Rotation is tested by rotating the image in 60 degrees direction. Figure 13 (a) shows the watermarked image, Figure 13 (b) and (c) plot of PSNR and NC values for medical images with rotation attacks.

4.3.7 Sharpening

Sharpening operations are used to enhance the subjective quality. A sharp image includes small components, the fine detail, down to the limit of vision. Thus, it is the size of the finest details that also contributes to our perception of sharpness. Figure 14 (a) shows the watermarked image, Figure 14 (b) and (c) plot of PSNR and NC values for medical images with sharpening attacks.

4.3.8 Smoothing

In smoothing, the data points of an image are modified, so individual points (presumably because of noise) are reduced. The individual points that are lower than the adjacent points are increased leading to a smoother image. Figure 15 (a) shows the watermarked image, Figure 15 (b) and (c) plot of PSNR and NC values for medical images with smoothing attacks.

4.3.9 Intensity Transformation

Intensity Transformation attack is adjusting the intensity range of image. Figure 16 (a) shows the watermarked image, Figure 16 (b) and (c) plot of PSNR and NC values for medical images with intensity transformation attacks.

4.3.10 Row Column Blanking

In row-column blanking attack, a set of rows and columns are deleted. In row-column blanking attack, a set of rows

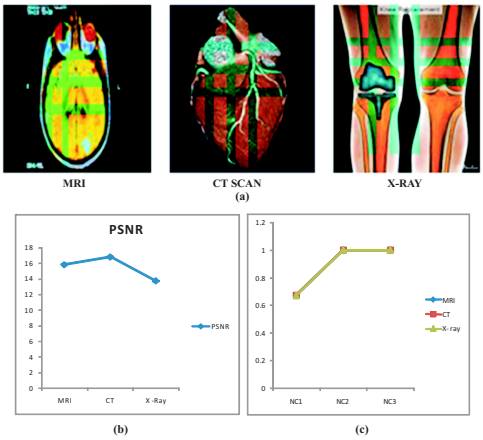


Figure 17: (a) Watermarked image with row column blanking attacks (b) PSNR values (c) NC values

and columns are deleted from 50 to 60, 80 to 90 and 110 to 120. Figure 17 (a) shows the watermarked image, Figure 17 (b) and (c) plot of PSNR and NC values for medical images with row column blanking attacks.

When compared with MRI and X-ray images, the CT images gives more peak signal to noise ratio (PSNR) and normalized correlation (NC) values for various attacks such as, median filtering, Gaussian blur, rotation, JPEG compression, smoothing and row column blanking. Third watermark achieves more robustness for various attacks such as, salt and pepper noise, Gaussian noise, median filtering, Gaussian blur, JPEG compression, sharpening, smoothing, and row column blanking on multiple watermarks.

5 Conclusions

This paper presents a multiple digital image watermarking scheme based on Discrete Wavelet Transform (DWT) and Singular Value Decomposition (SVD) for medical images. The proposed watermarking scheme, multiple watermarks are used to helps in addressing different problems such as, robustness against attacks and high security of medical images for patients privacy. In channel separation and merging of watermarking algorithm the blue component of watermark achieves more robustness, when compared with red and green component of watermark. The DWT and SVD based watermarking algorithm achieves imperceptibility, robustness against attacks. Thus the performance measures calculation shows that our proposed method is higher when compared to the existing method.

References

[1] G. Coatrieux, L. Lecornu, Ch. Roux, Fellow, and B. Sankur, "A review of image watermarking applications in healthcare," in *28th Annual International Conference of the IEEE on Engineering in Medicine and Biology Society*, pp. 4691–4694, 2006.

[2] G. Coatrieux, H. Maitre, B. Sankur, Y. Rolland, and R. Collorec, "Relevance of watermarking in medical imaging," in *IEEE International Conference on ITAB*, pp. 250–255, USA, 2000.

[3] K. S. Dhaliwal and R. Kaur, "Comparative study of single watermarking to multiple watermarking over a color image," *International Journal of Latest Trends in Engineering and Technology*, vol. 2, no. 2, pp. 43–48, 2013.

[4] A. Giakoumaki, S. Pavlopoulos, and D. Koutsouris, "Multiple image watermarking applied to health information management," *IEEE Transaction on Information Technology in Biomedicine*, vol. 10, no. 4, pp. 722–732, 2006.

[5] A. Giakoumaki, S. Pavlopoulos, and D. Koutsouris, "A multiple watermarking scheme applied to medical image management," in *26th Annual International Conference of the IEEE in Engineering in Medicine and Biology Society*, vol. 2, pp. 3241–3244, 2004.

[6] L. C. Huang, L. Y. Tseng, M. S. Hwang, "The study of data hiding in medical images," *International Journal of Network Security*, vol. 14, no. 6, pp. 301–309, 2012.

[7] B. M. Irany, X. C. Guo, and D. Hatzinakos, "A high capacity reversible multiple watermarking scheme for medical images," in *17th International Conference on Digital Signal Processing*, pp. 1–6, 2011.

[8] M. Kallel, M. S. Bouhlel, and J. C. Lapayre, "Use of multi-watermarking schema to maintain awareness in a teleneurology diagnosis platform," *Radio Engineering*, vol. 19, no. 1, pp. 68–73, 2010.

[9] M. I. Khan, Md. M. Rahman, and Md. I. H. Sarker, "Digital watermarking for image authentication based on combined DCT, DWT and SVD transformation," *International Journal of Computer Science Issues*, vol. 10, no. 1, 2013.

[10] Y. Li and X. Wang, "A novel multiple watermarking algorithm based on bandelet transform," *IEEE Global Congress on Intelligent Systems*, vol. 4, pp. 238–242, 2009.

[11] S. Mehta, R. Nallusamy, R. V. Marawar, and B. Prabhakaran, "A study of DWT and SVD based watermarking algorithms for patient privacy in medical images," in *IEEE International Conference on Healthcare Informatics*, pp. 287–296, 2013.

[12] N. A. Memon, S. A. M. Gilani, and S. Qayoom, "Multiple watermarking of medical images for content authentication and recovery," in *13th IEEE International Multitopic Conference*, pp. 1–6, 2009.

[13] J. Mishra, M. V. Patil, and J. S. Chitode, "An effective audio watermarking using DWT-SVD," *International Journal of Computer Applications*, vol. 70, no. 8, pp. 6–11, 2013.

[14] N. Mohananthini and G. Yamuna, "Color image multiple watermarking scheme based on discrete wavelet transform," in *International Conference on Science,*

Engineering and Management (ICSEM13), pp. 783–787, 2013.

[15] N. Mohananthini and G. Yamuna, "Multiple successive watermarking scheme based on wavelet transform," *International Journal of Emerging Trends and Technology in Computer Science*, vol. 2, no. 2, pp. 416–420, 2013.

[16] N. Mohananthini and G. Yamuna, "Performance comparison of single and multiple watermarking techniques," *International Journal of Computer Network and Information Security*, vol. 6, no. 7, pp. 28–34, 2014.

[17] N. Mohananthini, G. Yamuna, and R. Vivek, "Comparison of successive and segmented watermarking techniques for colour images," in *National Conference on Emerging Trends in Information and Communication Technology (NCETICT'13)*, pp. 13–16, 2013.

[18] K. A. Navas, M. C. Ajay, M. Lekshmi, T. S. Archana, and M. Sasikumar, "DWT-DCT-SVD based watermarking," in *Third International Conference on Communication Systems Software and Middleware and Workshops*, pp. 271–274, 2008.

[19] S. Radharani and M. L. Valarmathi, "Multiple watermarking scheme for image authentication and copyright protection using wavelet based texture properties and visual cryptography," *International Journal of Computer Applications*, vol. 23, no. 3, pp. 29–36, 2011.

[20] A. Sleit, S. Abusharkh, R. Etoom, and Y. Khero, "An enhanced semi-blind DWT - SVD based water marking technique for digital images," *The Imaging Science Journal*, vol. 60, no. 1, pp. 29–38, 2012.

[21] C. Sonika, Rathi, and S. Vandana Inamdar, "Medical images authentication through watermarking preserving ROI," *Health Informatics - An International Journal*, vol. 1, no. 1, pp. 27–42, 2012.

[22] S. S. Sujatha and M. Mohamed Sathik, "A novel DWT based blind watermarking for image authentication," *International Journal of Network Security*, vol. 14, no. 4, pp. 223–228, 2012.

[23] B. Surekha and G. N. Swamy, "Sensitive digital image watermarking for copyright protection," *International Journal of Network Security*, vol. 15, no. 2, pp. 113–121, 2013.

[24] A. Z. Tirkel, G. A. Rankin, R. M. Van Schyndel, W. J. Ho, N. R. A. Mee, and C. F. Osborne, "Electronic water mark," *Proceedings of DICTA'93*, pp. 666–673, 1993.

[25] C. S. Woo, J. Du, and B. Pham, "Multiple watermark method for privacy control and tamper detection in medical images," in *APRS Workshop on Digital Image Computing Pattern Recognition and Imaging for Medical Applications*, pp. 59–64, 2005.

[26] G. Yamuna and D. Sivakumar, "A novel digital image watermarking scheme using biorthogonal wavelets," *ICTACT Journal on Image and Video processing*, vol. 1, no. 4, pp. 236–240, 2011.

[27] D. Zheng, Y. Liu, J. Zhao, and A. Saddik, "A survey of RST invariant image watermarking algorithms," *ACM Computing Surveys*, vol. 39, no. 2, pp. 91–96, 2007.

Mohananthini Natarajan received her B.E. (Electrical and Electronics) Degree from Annai Mathammal Sheela Engineering College, Namakkal, Tamil Nadu, India in 2007. She received her M.E. (Applied Electronics) Degree from Anna University, Chennai, Tamilnadu, India in 2009. She has published technical papers in national and international conferences and journals. Her current research areas are digital image processing, information security and optimization.

Yamuna Govindarajan received her B.E. (Electronics and Communication) Degree from the Regional Engineering College, Trichy, Tamil Nadu, India in 1987. She received her M.E. (Power Systems) Degree from Annamalai University in the year 1991. She received her Ph.D Degree in Electrical Engineering from the Annamalai University in the year 2010. She has published many technical papers in national and international conferences and journals. Currently, she is working as Professor in the Department of Electrical and Electronics Engineering, Annamalai University, Tamil Nadu, India. Her areas of interest are digital signal processing, digital image processing and information security.

Leveraging P2P Interactions for Efficient Location Privacy in Database-driven Dynamic Spectrum Access

Erald Troja[1], Spiridon Bakiras[2]
(Corresponding author: Erald Troja)

The Graduate Center, City University of New York[1]
365 Fifth Avenue, New York, NY, 10016, USA
John Jay College, City University of New York[2]
524 West 59th Street, New York, New York 10019, USA
(Email: etroja@gc.cuny.edu, sbakiras@jjay.cuny.edu)
(Received Sept. 11, 2014; revised and accepted Jan. 16 & May 15, 2015)

Abstract

Dynamic spectrum access (DSA) is a novel communication paradigm that enables wireless clients to utilize statically allocated radio channels that are currently idle. Specifically, in the database-driven DSA model, clients learn their geographic location through a GPS device and use this location to retrieve a list of available channels from a centralized white-space database. To mitigate the potential privacy threats associated with location-based queries, existing work has proposed the use of private information retrieval (PIR) protocols when querying the database. Nevertheless, PIR protocols are very expensive and may lead to significant costs for highly mobile clients. In this paper, we propose a novel method that allows wireless users to collaborate in a peer-to-peer (P2P) manner, in order to share their *cached* channel availability information that is obtained from previous queries. To preserve location privacy against other users, we leverage an *anonymous veto* protocol that anonymizes the exchange of information among a group of users. Our experimental results with a real-life dataset show that our methods reduce the number of PIR queries by 50% to 60%, while incurring low computational and communication costs.

Keywords: Computer-communications networks, database applications, database management, distributed systems, GIS, spatial databases

1 Introduction

Radio spectrum is governed by federal agencies, mainly through a static sharing strategy. However, with the exponential increase of mobile devices capable of Internet connectivity, static spectrum sharing has led to a spectrum shortage. To this end, dynamic spectrum access (DSA) is a novel communication paradigm that enables wireless clients to utilize statically allocated radio channels when not in use by their licensed owners. DSA is accomplished through cognitive radio (CR), an intelligent wireless communication system that is aware of its operating spectral environment [33].

Currently, there exist two main approaches by which CR nodes acquire their spectral knowledge. On one end lies the distributed and cooperative sensing method, where nodes perform sheer power detection and coordination [18, 45, 47, 48]. On the other end lies the database-driven model, where nodes learn their spectral surroundings by querying a centralized white-space database (WSDB). In this latter model, mobile clients no longer bear the task of fusing spectral knowledge regarding the surrounding geographical area. Instead, they simply send their GPS coordinates to the database and receive the centrally fused repository report for that area.

On May 2012, the FCC issued a ruling [7] that obsoletes the distributed and cooperative sensing methods for white-space TV bands, thus requiring all CR nodes to implement the database-driven approach. Nevertheless, this approach suffers from severe location privacy leakage. According to the FCC specifications [8], a mobile TV band device (TVBD) must access the WSDB for a list of available channels, every time it is activated from a power-off state. Furthermore, a mobile TVBD must issue a new query to the WSDB whenever it moves further than 100m from its previous location. Given that the GPS coordinates are part of the query, the WSDB operator can easily build a detailed history of a mobile TVBD's trajectories. These trajectories would allow the WSDB to infer sensitive information about the mobile user, such as home location, health condition, lifestyle habits, political

and religious affiliations, etc.

To mitigate the potential privacy threats associated with location-based queries, existing work has proposed the use of private information retrieval (PIR) protocols when querying the database [10]. A PIR protocol enables a user to retrieve a record from a database server, while maintaining the identity of the record secret from the server. However, PIR protocols are very expensive and may lead to significant costs in the case of highly mobile clients that issue numerous queries throughout their trajectories. For example, the trivial PIR case is to download the entire database (e.g., once per day), which clearly preserves privacy but incurs an overwhelming communication cost.

Typical PIR protocols offer a trade-off between computational and communication complexity. Computational complexity has an adverse impact mostly at the server side, whereas communication complexity affects the end-user as well (especially in the case of wireless devices). For instance, the scheme by Trostle and Parrish [40] that is applied in previous work [10] is computationally efficient, but its communication cost is equal to a large percentage of the database size. On the other extreme, Gentry and Ramzan's protocol [11] is considered to attain the best communication complexity, but incurs a high computational cost due to its heavy use of cryptographic operations [35].

In this paper, we argue that any location privacy method for the database-driven DSA model is bounded by the limitations of the underlying PIR protocol. As such, it is desirable to identify new mechanisms for users to acquire the necessary spectral knowledge. Our intuition is that, in a white-space TV band network, mobile TVBD users will gradually develop a trajectory-specific spectrum knowledge *cache*, through a series of PIR requests. In the extreme case, some users might opt to download the entire WSDB (trivial PIR case) before initiating their travel[1]. Therefore, we propose that mobile users that are within communication range interact in a peer-to-peer (P2P) manner, in order to exchange their cached spectrum knowledge for the surrounding area.

However, a user's spectrum knowledge cache is a summary of his/her recent trajectory, and some users may be unwilling to share that information due to privacy concerns. To this end, we leverage the *anonymous veto network* (AV-net) protocol of Hao and Zieliński [17] that anonymizes the exchange of information among a group of users. Our experimental results with Microsoft's GeoLife trajectory dataset [49] show that our methods reduce the number of PIR queries by 50% to 60%, while incurring low computational and communication costs for the mobile clients.

The rest of this paper is organized as follows. Section 2 presents a literature review on location privacy and Section 3 provides the necessary background on the cryptographic primitives utilized in our methods. Section 4 describes the details of our P2P protocol and Section 5 presents the results of the experimental evaluation. Finally, Section 6 concludes our work.

[1]However, due to its overwhelming communication cost, the trivial PIR case may be infeasible for most users.

2 Related Work

Most existing protocols on location privacy build upon the notion of k-anonymity [38] or l-diversity [31]. A spatial query is considered to be k-anonymous if it is indistinguishable from at least $k-1$ other queries spawning from the same area, usually called the spatial cloaking region (SCR). The SCR is chosen such that it encapsulates the querying user as well as at least $k-1$ other users. To compute the SCR, existing k-anonymity algorithms typically extend the SCR around the query point until it encapsulates $k-1$ other users [16, 34, 44].

On the other hand, l-diversity based methods, such as the ones proposed in [42, 43], extend the SCR until $l-1$ different locations are included in the query. Both k-anonymity and l-diversity offer some degree of location privacy, but they are susceptible to semantic location information leakage. For example, if the SCR only contains casinos, the server can infer (to a certain degree) that the mobile user is interested in gambling. To this end, the work of Lee et al. [27] attempts to provide location privacy using location semantics.

The k-anonymity and l-diversity based approaches, as well as collaborative location privacy protection methods [9, 41], often rely on third party trusted *anonymizers*, an expensive and scarcely available option. Ghinita et al. [12] propose the first privacy-preserving protocol that does not require an anonymizer. They focus on nearest neighbor queries and introduce a method that achieves perfect location privacy via the PIR protocol of Kushilevitz and Ostrovsky [26].

Other protocols on location privacy revolve around the notion of data perturbation, location hiding, and the introduction of data point dummies. Meyerowitz et al. [32] introduce a data perturbation technique to protect personal location data against untrusted location based service (LBS) servers. In their work, they develop CacheCloak, a protocol that enables real time anonymization of location data. CacheCloak relies on a trusted anonymizing server to generate mobility prediction from historical data, and then submit intersecting predicted paths simultaneously to the LBS. Reliance on a trusted server is a very expensive and strong assumption that we would like to avoid in our proposed methods. Also, the intuition behind CacheCloak is to obscure the user's path by surrounding parts of it with other user's paths, effectively creating a k-anonymous region.

Huang at al. [20] study the problem of location privacy preservation with respect to an LBS that threatens a user's location privacy by tracking transmitting frames. The authors argue that correlation attacks between a node's old and new address are not sufficient. They suggest the concept of a *silent period*, defined as the transi-

tion period between the use of new and old pseudonyms, during which a node is not allowed to disclose neither the old nor the new address.

Furthermore, Huang at al. [21] extend their previous work [20] and study the problem of location privacy with respect to a user's communication with network access points. They mainly focus on the issue of how location privacy enhancements affect the perceived Quality of Service (QoS). The authors propose a silent cascade method to enhance a user's location privacy by trading end-to-end delay for anonymity. They abstract silent cascade as a mix-network model and evaluate its performance. In our setting, however, we are concerned with the effectiveness (computation and communication cost) of the location privacy-preserving protocol itself.

Kido et al. [25] suggest a location privacy-preserving method that uses the notion of *dummy* data (false positives), in order to hide the user's true location from the LBS. The authors argue that, after sending their GPS coordinates to the LBS, users can not delete or modify their disclosed location. In other words, users cannot prevent the service providers from analyzing motion patterns using stored location data. In their proposed method, users send their true location data along with several false ones (dummies) to the service provider, who subsequently creates a reply message for each received data point. Users then simply extract the correct information from the reply messages. However, it is clear that this scheme is essentially a k-anonymity based approach.

Similarly to Kido et al. [25], Lu at al. [30] introduce PAD, a method that injects dummy locations in the query, which are generated according to either a virtual grid or a circle. The virtual grid or circle cover the user's actual location, and their spatial extents are controlled by appropriate generating algorithms. However, PAD relies on a server-side front-end, in order to be integrated into existing client/server mobile service systems. Even though PAD takes into account the number of location points in the query, as well as the area of the region covered by those points, it can be effectively reduced to a pure k-anonymity based technique.

Other techniques such as routing anonymization and privacy-preserving wireless broadcast networks have been suggested. The authors in [2] suggest wireless anonymous routing (WAR) as the main approach of achieving anonymity in a wireless broadcast network. Ref. [50] and [28] propose lightweight ad hoc routing protocols in order to preserve location privacy of the mobile nodes. Lastly, the authors in [22] provide evidence that such anonymization and location privacy-preservation techniques can be applied even in radio frequency identification networks (RFID). Such techniques are orthogonal to our proposed methods and can be applied in an optional and complementary fashion in order to provide local network addressing anonymity as well as geo-location privacy.

Until recently, location privacy work in the dynamic spectrum access domain has mainly focused on the collaborative spectrum sensing model. In particular, most existing solutions attempt to protect the location privacy of mobile users that submit sensing reports to a fusion center [19, 29, 37]. The collaborative sensing and reporting approach was embraced as a superior method compared to the centralized database approach. This is no longer the case, though, at least in the white-space TV band realm.

Due to the recency of the FCC's ruling (May 2012), location privacy research in database-driven DSA networks is still in its infancy. The state-of-the-art protocol is due to Gao et al. [10], which builds upon a modified version of Trostle and Parrish's PIR scheme [40]. They assume a fixed grid of $n \times n$ cells, where each cell contains a bitmap that represents the channel availability information (typically 32 bits). Nevertheless, their scheme incurs a high communication cost of $(2n + 3) \cdot \log p$ bits, where p is a 2048-bit modulus. For example, if $n = 5000$, the amount of data transmitted to retrieve the bitmap of a single cell is 2.5 MB, which is approximately 2.6% of the whole database size. For highly mobile clients, the cost of this approach can exceed the cost of the trivial PIR case.

3 Preliminaries

In this section, we give a brief description of the cryptographic primitives incorporated in our methods. Section 3.1 provides some background on anonymous communication and Section 3.2 introduces the 2-round anonymous veto network (AV-net) of Hao and Zieliński [17]. Section 3.3 presents the threat model of our approach.

3.1 Anonymous Communication

Research on anonymous communication has evolved due to the *dining cryptographers* problem, introduced by Chaum in 1988 [5]. Essentially, a dining cryptographers network (DC-net) allows groups of $n > 2$ participating users to contribute their boolean bits towards a boolean-OR calculation of some statement, while preserving the privacy of the individual inputs. DC-nets have many weaknesses and are considered impractical due to complex key setup, message collisions, and vulnerability to disruptions. Alternatively, circuit evaluation techniques, such as the ones proposed in [13, 46], can also be used towards the secure computation of a boolean-OR function. However, as pointed out by Brandt [4], the circuit evaluation technique is expensive and impractical.

A similar problem is the anonymous veto network (AV-net), which allows groups of $n > 2$ participating users to vote against a given statement. In the setting of a white-space TV bands database, where channel availability can be represented via a boolean bit, a sample statement might be: "none of the group members knows that the channel is free." If any of the users in the group anonymously vetoes the statement, it means that "at least one of the users in the group knows that the channel is free."

Unlike DC-nets, AV-net protocols do not require secret channels in order to exchange messages. Furthermore, they have no message collisions and are very resistant to disruptions. Nevertheless, all existing AV-net protocols assume the existence of an *authenticated broadcast channel*, which is easily implemented using digital signatures [5]. Several such anonymous veto protocol designs exist in the literature [4, 15, 17, 24]. In our work, we leverage the 2-round AV-net protocol of Hao and Zieliński [17], because it is more efficient in terms of number of rounds, computation, and communication cost.

3.2 The 2-Round AV-net Protocol

Setup. All users participating in the protocol agree on two public parameters, namely G and g. G is a finite cyclic group of prime order q in which the Decision Diffie-Hellman (DDH) problem is hard [3], and g is a generator of G. These values are fixed and used in all protocol invocations. Subsequently, each participant P_i, $i \in \{1, 2, \ldots, k\}$, selects a random secret value $x_i \in_R \mathbb{Z}_q$.

Round 1. In the first round, every participant P_i broadcasts g^{x_i}. When the first round completes, each participant P_i computes

$$g^{y_i} = \prod_{j=1}^{i-1} g^{x_j} \Big/ \prod_{j=i+1}^{k} g^{x_j}$$

Round 2. In round 2, every participant P_i broadcasts a value $g^{c_i y_i}$ where c_i is either x_i or a random value $r_i \in_R \mathbb{Z}_q$, depending on whether participant P_i vetoes the statement or not.

$$g^{c_i y_i} = \begin{cases} g^{r_i y_i} & \text{if } P_i \text{ sends '1' (veto),} \\ g^{x_i y_i} & \text{if } P_i \text{ sends '0' (no veto).} \end{cases}$$

In order to test the final result, all participants compute $\prod_i g^{c_i y_i}$. If nobody vetoed the statement, then $\prod_i g^{c_i y_i} = \prod_i g^{x_i y_i} = 1$, since $\sum_i x_i y_i = 0$. If, however, one or more participants vetoes the statement by sending a '1', we have $\prod_i g^{c_i y_i} \neq 1$.

3.3 Threat Model and Security

In this work, we are not concerned with privacy against the WSDB operator. We assume that mobile users, when needed, query the WSDB through a standard PIR protocol. Instead, in our methods, the adversary is one or more users in the group that executes the AV-net protocol, or any eavesdropper that monitors the exchange of messages over the wireless channel. The adversary runs in polynomial time, and its goal is to identify a user that vetoes a certain statement.

Note that, in the case of malicious adversaries, the protocol described above necessitates zero-knowledge proof (ZKP) schemes, such as Schnorr's signature [36]. In particular, during each round, users must demonstrate knowledge of their own secret values, such as x_i and c_i. Nevertheless, in our work, we assume the *honest-but-curious* adversarial model, i.e., all users follow the protocol correctly but try to gain an advantage by examining the communication transcript. As a result, we do not implement zero-knowledge proofs.

Our methods inherit the security of the underlying AV-net protocol [17]. As such, they are *semantically secure* [14], i.e., it is infeasible to derive any information about a mobile client's input, given its published values and the public parameters. The security is based on the DDH assumption. Therefore, an eavesdropper is unable to determine whether a user has vetoed a statement. Our methods are also secure against *partial* collusions, i.e., when some participants collude (by revealing their secret values) to determine the input of a certain user. As explained in [17], only a *full* collusion against a single user can compromise security, i.e., when $k-1$ users reveal their values to identify whether the k-th user vetoed a statement.

4 P2P Protocol

In this section, we present the P2P protocol that allows a group of users to share anonymously their cached spectrum information. Section 4.1 describes the system architecture, and Section 4.2 explains the protocol initiation process. Section 4.3 presents the criteria for mobile nodes to participate in this protocol, and Section 4.4 describes the group formation mechanism. Finally, Section 4.5 introduces the details of the AV-net protocol invocation.

4.1 System Architecture

Similar to previous work [10], we assume a fixed grid of $n \times n$ cells, where mobile users can communicate through white-space TV bands, while maintaining their location privacy. According to the FCC specifications [8], each cell is 100m×100m in size, and users may need to query the WSDB whenever they move into a cell with no prior spectrum availability knowledge. The dimensions of the grid (i.e., n) can be made arbitrarily large, which has a direct effect on the database size.

Note that, mobile TVBDs are allowed to communicate only in the frequency ranges 512-608 MHz (TV channels 21-36) and 614-698 MHz (TV channels 38-51), i.e., there are a total of 31 possible white-space TV band channels that can be accessed in a DSA manner. Therefore, we represent the daily channel availability as 32 bits (per cell), where bit 0 represents a busy channel and bit 1 represents an idle channel. As an example, when $n = 5000$, the WSDB is 100 MB in size. The trivial PIR case is impractical in this setting, since it involves downloading the entire WSDB. This would take approximately 35 mins on 3G networks[2] [39].

[2]Furthermore, communication over a cellular network is a priced resource that should be avoided.

In our model, we assume an out-of-band common control channel (CCC) through a dedicated transceiver. This enables mobile users to exchange concurrently both control and data messages. Out-of-band CCC coordination can be realized over the 802.11 protocol in *ad-hoc* mode or through any of the methods proposed in [1, 6]. We emphasize that 802.11 is not a viable protocol for long range communications, hence it is only used to implement the out-of-band CCC for communications within a 100m×100m cell.

The FCC's white-space TV band DSA specifications state that "A mode II personal/portable device may load channel availability information for multiple locations around, i.e., in the vicinity of, its current location and use that information in its operation." Accordingly, in our methods, we assume a PIR protocol that retrieves channel information for multiple cells with a single query[3]. As a proof of concept, we consider a fixed grouping of the available cells into 4×4 blocks. Therefore, we assume that each PIR query retrieves the 16-cell block that contains the user's current cell.

Figure 1a shows an example of this approach. The black colored cells signify the locations where a new PIR query is issued, due to lack of spectrum availability knowledge. The alternating white and grey colored cells identify the different blocks, with the block *id* shown in the lower-left corner of the block. Note that, even though we assume a specific method for querying the WSDB, our protocol is *orthogonal* to the underlying PIR query/reply structure. Any WSDB indexing method is a viable candidate for our protocol, but for the PIR reply to be of some utility to the client, the retrieved cells should be spatially close to the user's location.

As illustrated in Figure 1a, each of the three mobile TVBDs gradually builds a spectrum knowledge *cache* containing channel availability information from their respective trajectories. When the users eventually meet at the diagonally striped cell, it may be beneficial to all of them to exchange their cached information. To maximize utility for all participating users, the sharing of spectrum information involves the area surrounding the current location (as users may continue their trajectories towards any direction). In particular, the TVBD nodes agree on the number of surrounding rings (AR) that they wish to explore during the protocol invocation. (Table 1 summarizes the symbols used in the remainder of this paper, along with the values tested in the experimental evaluation.) In the example of Figure 1b, $AR = 3$, and the explored region is shown in a darker shade.

To illustrate the location privacy leakage from a *plaintext* exchange of spectrum availability information (i.e., without the invocation of the AV-net protocol), consider the example of Figure 1b. We can infer that u_1 arrived at the current cell through block 9, while u_3 visited blocks 13 and 14. On the other hand, u_2's trajectory contains some uncertainty, as u_2 may have arrived at the current cell through blocks 2, 7, or 12. Furthermore, if two users participate in the same group multiple times (at different locations), they can derive more information about each other's movement patterns.

[3]All existing PIR schemes can retrieve multiple records from a database.

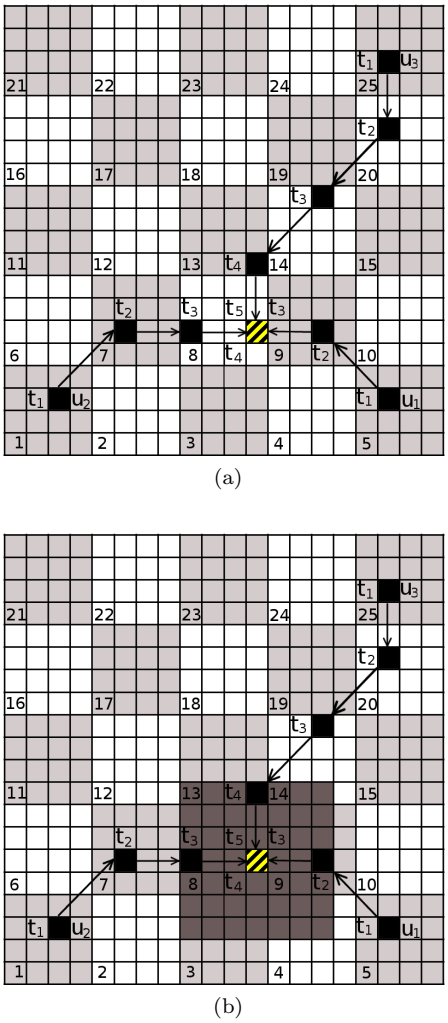


Figure 1: (a) Three mobile users querying a WSDB via PIR, and intersecting at the diagonally striped cell (b) Three mobile users invoking the AV-net protocol for the region identified by the darker shaded cells

We assume that intersecting users remain within communication range for ample periods of time (e.g., 1-2 minutes). However, they do not need to reside in the same cell continuously. The three conditions that control a successful invocation of our protocol are (i) protocol *initiation*, (ii) protocol *participation*, and (iii) successful *group formation*. Group formation is dependent on at least three users willing to engage in the P2P protocol, such that at least one of the engaging users is an *initiator*. We examine each condition separately in the following sections.

4.2 Protocol Initiation

Ideally, a mobile TVBD would like to maintain DSA connectivity throughout its trajectory, without any disrup-

Table 1: Summary of symbols

Symbol	Description	Range
GS	Group size	3-10
BS	Number of cells in a PIR block	16
AR	Ring(s) explored through AV-net invocations	1-3
AP	AV-net participation probability (fixed)	0-1
PI	AV-net participation probability increment (TCP)	0.05-0.2
K	AV-net initiation threshold	0.2-0.8
AK	Actual knowledge of the AR area	0-1

tions. As such, whenever the TVBD moves into a new cell, it measures the ratio of knowledge (AK) in the surrounding area. If that ratio falls under a system-defined threshold K, it initiates the protocol that triggers the group formation algorithm (described later). Algorithm 1 shows the detailed protocol initiation procedure. If there is no channel availability information for the current cell, the user always initiates the protocol (Lines 6–8), because it needs to identify free channels. On the other hand, if the current cell does exist in its cache, it computes the ratio AK for the present position (Lines 9–15). Specifically, each surrounding ring is assigned an identical aggregate weight (equal to $1/AR$), which is split equally among the individual cells. As a result, cells in the inner rings carry more weight than those in the outer rings, and lack of knowledge in the inner rings is more likely to initiate the protocol.

Algorithm 1 Protocol initiation Algorithm

```
1: procedure INITIATE-PROTOCOL
2:
3:     bool initiate = false;
4:     double AK = 0.0;
5:
6:     if no spectrum information for current cell then
7:         initiate = true;
8:     else
9:         for i = 1 to AR do
10:            for all cells c_j in ring i do
11:                if no spectrum information for c_j then
12:                    AK = AK + 1.0/(AR · 8 · i);
13:                end if
14:            end for
15:        end for
16:        if AK ≤ K then
17:            initiate = true;
18:        end if
19:    end if
20:    return initiate;
21: end procedure
```

4.3 Protocol Participation

Participation is defined as the selfless event, where one or more users in the group decide to participate in the AV-net protocol for the purpose of disseminating (and also collecting) channel information about the surrounding area. In order to avoid meaningless (due to repetition) AV-net protocol invocations that could lead to battery drainage, we propose the following three probabilistic AV-net participation methods.

Fixed Probability. This is the simplest approach where, whenever a protocol is initiated, a nearby TVBD always chooses to participate with probability AP. Larger AP values produce a greedy behavior that is optimal in terms of PIR query savings. On the other hand, this may also lead to numerous AV-net invocations in close (spatial) proximity, which are redundant in terms of gained knowledge.

TCP-like Approach. In the second method, we borrow from TCP Reno's congestion control mechanism [23]. In particular, a mobile user starts with a participation probability $AP = 1.0$. At each successful AV-net participation, AP is cut by half. Otherwise, if there is a protocol initiation but the TVBD does not participate, AP is incremented by PI units. This technique is expected to be the most conservative one, due to its aggressive back-off behavior.

Weighted Sliding Window. The final method is based on the weighted sliding window (SW) projection. We experimented with different window sizes, and decided to utilize a model with five entries, such that $W_1 = 0.5$, $W_2 = 0.25$, $W_3 = 0.15$, $W_4 = 0.07$, $W_5 = 0.03$, and $\sum_i W_i = 1$. (W_1 corresponds to the most recent entry.) The current window snapshot is stored as a 5-bit array, where '0' represents participation and '1' represents non-participation. In order to determine the probability of participation, a mobile user first checks its window and sums up past events for which it did not participate. For example, if the current SW snapshot is $(0, 1, 0, 1, 0)$, the user will participate with probability $0.25 + 0.07 = 0.32$. The weighted SW allows us to weight recent historical data more heavily than older ones, when determining the projected probability. This fits well with the intended participation model, in which more recent participation should lead to lower participation probability in the near future.

4.4 Group Formation

When Algorithm 1 (protocol initiation) returns *true*, the underlying TVBD initiates an invocation of the AV-net protocol. This is done by broadcasting its interest in the lowest out-of-band CCC channel. Assuming 801.11 as our out-of-band CCC implementation, any potential initiators will broadcast their unique MAC addresses, their current cell id, their *rendezvous* channel id[4], and an initiation flag over 802.11 channel 1. Users that are already engaged in an AV-net invocation/transmission will not hear such broadcast. We assume standard 802.11 MAC contention mechanisms are in place. We coin as "root" the first mobile user that successfully broadcasts the AV-net initiation control packet, regarding a specific cell id. Any other users (including other potential initiators situated in the same cell) that receive the first successful broadcast from a root node, and whose cell id *matches* the broadcast cell id, will use a simple three-way handshake group formation protocol.

[4]Assuming there is a free channel in the out-of-band CCC range.

Mobile users that decide to participate (based on the methods described earlier) or had attempted to initiate an AV-net invocation themselves, will first switch to the rendezvous channel. They will announce to the root user, through broadcast communication, their willingness to engage. We coin as "children" any of the users that have successfully rendezvoused in the channel id specified by the root user. The three-way handshake broadcast MAC protocol is summarized in Algorithm 2.

Algorithm 2 Group formation algorithm

```
1: procedure THREE-WAY-HANDSHAKE
2:
3:     [all children broadcast] Send Request To Join Group
4:
5:     while group size < GS do
6:         [root] randomly pick a child
7:         [root] send Clear to Join Group
8:         [root] increment group size counter
9:     end while
10:
11:    for all other children who sent a Request to Join do
12:        [root] send Reject to Join Group
13:    end for
14:    [all who received Clear to Join Group] Send Confirm To
   Join Group
15:
16:    [root] send ABORT if group size counter ≤ 3
17: end procedure
```

4.5 AV-net Protocol

When a group is formed, the nodes therein execute the AV-net protocol (as described in Section 3.2) for each bit of information that they want to share. However, to avoid excessive network delays due to the 2-round nature of the AV-net protocol, we group all individual invocations into two aggregate rounds, as shown in Algorithm 3. Specifically, the users first agree on the the specific order in which the cell information is transmitted, and then each user broadcasts its aggregate data to the rest of the group. The broadcast order can be arranged based on the unique MAC addresses of the TVBDs.

Algorithm 3 AV-net protocol

```
1: procedure AV-NET(G, g)
2:
3:     for all users i in the group do
4:         for all bits b in the explored area do
5:             compute g^{y_{i_b}};
6:         end for
7:     end for
8:
9:     for all users i in the group do
10:        for all bits b in the explored area do
11:            compute g^{c_{i_b} y_{i_b}};
12:        end for
13:        broadcast all exponentiations for user i;
14:    end for
15:
16:    for all users i in the group do
17:        for all bits b in the explored area do
18:            compute r_b = Π_i g^{c_{i_b} y_{i_b}};
19:            if r_b ≠ 1 then
20:                mark the corresponding channel as free;
21:            end if
22:        end for
23:    end for
24: end procedure
```

Lines 3–8 (Algorithm 3) correspond the the first round of the AV-net protocol, i.e., each node broadcasts a unique key for every bit of information in the surrounding AR rings. In the example of Figure 1b, where $AR = 3$, each node computes and broadcasts $32 \cdot (1 + \sum_{i=1}^{3} 8 \cdot i) = 1568$ modular exponentiation results. In Round 2 of the protocol (Lines 9–23), users publish their spectrum knowledge by choosing the appropriate values for c_{i_b} (as explained in Section 3.2). Specifically, if the underlying channel if free, the user vetoes that particular statement. Note that, in our running example, this step also involves the computation and broadcast of 1568 modular exponentiations. The result extraction phase of the algorithm (Lines 16–23) necessitates only GS modular multiplications per bit, and it is optional, i.e., it can be computed only when the user moves into the corresponding cell.

5 Experimental Evaluation

In this section we evaluate experimentally the performance of our methods. Section 5.1 describes the experimental setup and Section 5.2 presents our results.

5.1 Experimental Setup

We developed our experiments in Java SDK, running on a Ubuntu 10.4 LTS machine. To simulate the mobile TVBD users, we utilized Microsoft's GeoLife GPS Trajectories[5], which is an excellent dataset containing real-life trajectories from users traveling around Beijing, China. The GeoLife dataset [49] was collected as part of the Microsoft Research Asia GeoLife project, by monitoring numerous users for a period of over five years (from Apr. 2007 to Aug. 2012). A GPS trajectory from this dataset is represented as a sequence of time-stamped points, each containing information regarding the user's latitude, longitude, and altitude.

The dataset includes 17,621 trajectories, with a total distance of 1,292,951 kilometers, and a total duration of 50,176 hours. These trajectories were recorded by different GPS loggers and GPS-enabled phones, and have a variety of sampling rates. More specifically, 91.5 percent of the trajectories are logged in a dense representation, e.g., every 1–5 seconds or every 5–10 meters per point. We randomly selected 2774 intersecting trajectories, each simulating a unique user. For each trajectory, we measure (i) the average number of PIR queries issued by the user, and (ii) the average number of AV-net invocations that the user participates into.

In addition to the simulation results, we also implemented the basic cryptographic operations of the AV-net protocol on an iPhone 5, running iOS 7.1. Specifically, we cross compiled the GMP[6] multiple precision arithmetic library for the ARM architecture, and built a benchmark app to measure the cost of these operations on a handheld

[5] http://research.microsoft.com/en-us/projects/GeoLife/
[6] http://gmplib.org

device. We generated a cyclic group G of prime order q, where q is a 160-bit number. The group modulus was chosen as a 64-byte prime. Table 2 shows the cost of these operations.

Table 2: Cost of cryptographic primitives

Operation	Cost
Modular multiplication	0.004 ms
Modular exponentiation	0.518 ms

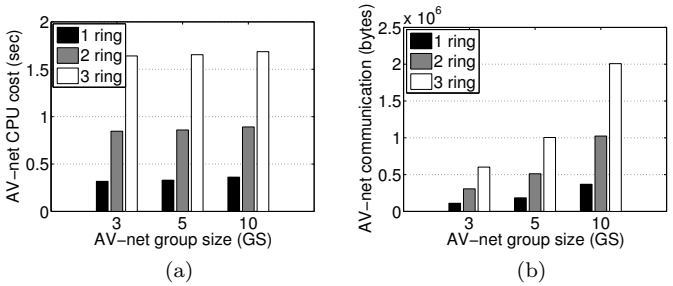
5.2 Results

Figure 2a illustrates the projected CPU time needed to run the AV-net protocol (Algorithm 3) on a handheld device. This cost is dominated by the expensive modular exponentiation operations and is, thus, unaffected by the group size GS. The major factor that determines this cost is the number of surrounding rings (AR) that are explored during a protocol invocation, since each cell contributes 32 modular exponentiations. Nevertheless, even for a value of $AR = 3$, the total CPU time is around 1.65 sec, which is an acceptable cost.

Furthermore, this cost can easily be reduced by 50%, using offline computations. Observe that, during the first round of the AV-net protocol, each node computes and publishes a large number of modular exponentiations. These values do not require any input from the other participating nodes and may, thus, be pre-computed offline. Specifically, a large pool of values (e.g., several hundred thousands) can be computed either at the mobile device during night time (when charging), or at a desktop machine for faster computations. The storage space required to maintain these values is insignificant compared to the storage capabilities of modern handheld devices.

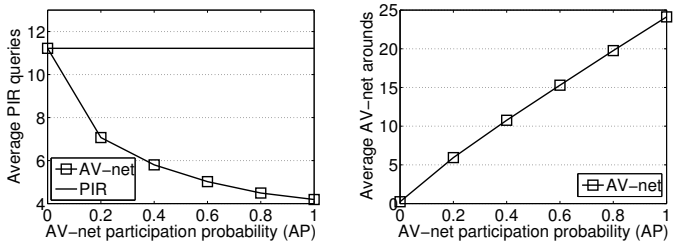
Figure 2b shows the total number of bytes that are broadcast during an AV-net protocol invocation. Clearly, the communication cost is linear in GS, as each group member needs to broadcast its own input to the protocol. We believe that $GS = 5$ is a very reasonable value for anonymity purposes, in which case the communication cost remains below 1 MB. While this cost might appear significant, we stress that, AV-net broadcasts occur over the 802.11 CCC band and do not involve the cellular network infrastructure.

Figure 3 investigates the effect of the fixed AV-net participation probability (AP) on the performance of our methods. For this experiment, we set $AR = 2$, $GS = 5$, and $K = 0.5$. The curve labeled "PIR" (Figure 3a) corresponds to the PIR-only approach, i.e., when users do not leverage our P2P protocol. When $AP = 0.5$, we observe a 50% reduction in the amount of PIR queries that are sent to the WSDB provider. Larger values naturally lead to better performance (over 60% reduction), but they increase considerably the number of AV-net invocations per user (Figure 3b). Nevertheless, as we have explained previously, PIR queries are much more expensive compared to the AV-net protocol.



(a) (b)

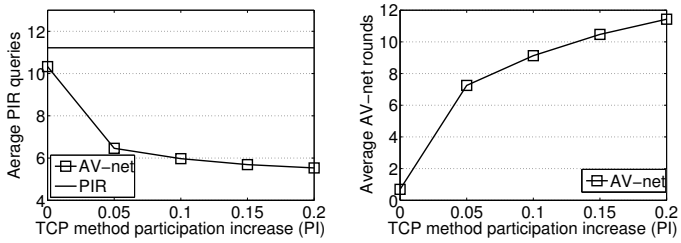
Figure 2: Cost of AV-net protocol on handheld devices (a) CPU cost (b) Communication cost



(a) (b)

Figure 3: Effect of varying the AV-net participation probability (a) Average number of PIR queries (b) Average number of AV-net invocations

Figure 4 shows the effect of the participation probability increment (PI) for the TCP-like approach ($AR = 2$, $GS = 5$, $K = 0.5$). Lower values of PI discourage users from participating in AV-net protocols and, thus, incur less cost compared to the fixed probability method (Figure 4b). However, as evident in Figure 4a, the TCP-like approach can still reduce the number of PIR queries by up to 50%.

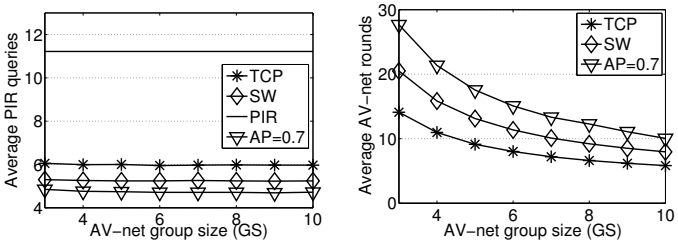


(a) (b)

Figure 4: Effect of varying the AV-net participation probability increment (TCP) (a) Average number of PIR queries (b) Average number of AV-net invocations

Figure 5 demonstrates the effect of the group size (GS) on the different methods ($AR = 2$, $K = 0.5$, $PI = 0.1$). As Figure 5a implies, larger groups do not contribute more information during the P2P data exchange. Therefore, the average number of PIR queries remains fairly constant. Nevertheless, users may still opt for larger

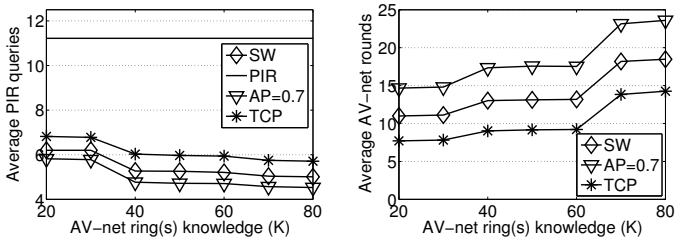
groups, in order to gain more privacy. On the other hand, a larger group size reduces the number of AV-net invocations (Figure 5b), because some groups may fail to form due to insufficient number of members. Among the three participation algorithms, the sliding window (SW) approach strikes a good balance between PIR savings (53%) and AV-net overhead (13 rounds, for $GS = 5$).



(a) (b)

Figure 5: Effect of varying the AV-net group size (a) Average number of PIR queries (b) Average number of AV-net invocations

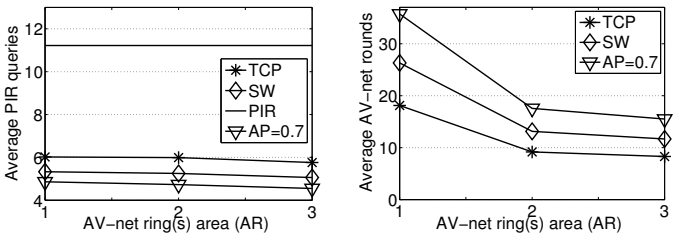
Figure 6 depicts the effect of the protocol initiation threshold (K) on the different methods ($AR = 2$, $GS = 5$, $PI = 0.1$). Recall that, this threshold represents a lower bound on the amount of spectrum knowledge that a mobile user must possess (regarding the surrounding area), in order to defer an AV-net protocol initiation. As evident in this figure, a knowledge of around 40%-50% is sufficient in terms of overall performance. Larger values to not offer much in terms of PIR reduction, but instead lead to unnecessary AV-net rounds. Similar to Figure 5, the SW participation method has the best performance.



(a) (b)

Figure 6: Effect of varying the AV-net initiation threshold (a) Average number of PIR queries (b) Average number of AV-net invocations

Finally, Figure 7 illustrates the effect of the number of surrounding rings (AR) that are explored during an AV-net protocol invocation ($K = 0.5$, $GS = 5$, $PI = 0.1$). The first observation, is that the number of PIR queries remains almost constant (Figure 7a). The reason is that, as shown in Figure 7b, exploring one ring at a time merely results in more AV-net rounds, since users invoke a new AV-net protocol once they move further away from their current position. However, the overall PIR reduction is not affected, because users still get most of their spectrum knowledge from the P2P protocol. A value of $AR = 2$ seems like the best choice, given that the number of AV-net rounds does not decrease significantly from 2 to 3 rings.



(a) (b)

Figure 7: Effect of varying the AV-net exploration area (a) Average number of PIR queries (b) Average number of AV-net invocations

6 Conclusions

Database-driven dynamic spectrum access is the standard mode of operation for cognitive radios in the white-space TV bands. This method requires mobile devices to periodically send their location to a centralized white-space database, in order to receive channel availability information in their surrounding area. Nevertheless, location-dependent queries pose a serious privacy threat, as they may reveal sensitive information about an individual. To mitigate this threat, previous work has proposed the use of private information retrieval (PIR) protocols when querying the database. In this work, we argue that PIR queries are very expensive and should be avoided, to the extent possible. To this end, we propose a novel approach that allows mobile users to share anonymously their cached channel availability information that is obtained from previous queries. Our experiments with a real-life dataset, indicate that our methods reduce the number of PIR queries by 50% to 60%. Furthermore, they are efficient in terms of both computational and communication cost.

Acknowledgments

This research has been funded by the NSF CAREER Award IIS-0845262.

References

[1] I. F. Akyildiz, W. Y. Lee, and K. R. Chowdhury, "CRAHNs: Cognitive radio ad hoc networks," *Ad Hoc Networks*, vol. 7, no. 5, pp. 810–836, 2009.

[2] M. Blaze, J. Ioannidis, A. D. Keromytis, T. G. Malkin, and A. Rubin, "Anonymity in wireless

broadcast networks," *International Journal of Network Security*, vol. 8, no. 1, pp. 37–51, 2009.

[3] D. Boneh, "The decision Diffie-Hellman problem," in *Algorithmic Number Theory*, pp. 48–63. 1998.

[4] F. Brandt, "Efficient cryptographic protocol design based on distributed ElGamal encryption," in *Information Security and Cryptology (ICISC'06)*, pp. 32–47. 2006.

[5] D. Chaum, "The dining cryptographers problem: Unconditional sender and recipient untraceability," *Journal of Cryptology*, vol. 1, no. 1, pp. 65–75, 1988.

[6] C. Cormio and K. R. Chowdhury, "A survey on MAC protocols for cognitive radio networks," *Ad Hoc Networks*, vol. 7, no. 7, pp. 1315–1329, 2009.

[7] FCC, "Third memorandum opinion and order," pp. 12–36, 2012.

[8] FCC, "Television band devices," p. 11, 2013.

[9] J. Freudiger, M. H. Manshaei, J. P. Hubaux, and D. C. Parkes, "On non-cooperative location privacy: A game-theoretic analysis," in *Proceedings of the 16th ACM conference on Computer and communications security (CCS'09)*, pp. 324–337, 2009.

[10] Z. Gao, H. Zhu, Y. Liu, M. Li, and Z. Cao, "Location privacy in database-driven cognitive radio networks: Attacks and countermeasures," in *IEEE INFOCOM'13*, pp. 2751–2759, 2013.

[11] C. Gentry and Z. Ramzan, "Single-database private information retrieval with constant communication rate," in *Proceedings of 32nd International Colloquium Automata, Languages and Programming (ICALP'05)*, pp. 803–815. 2005.

[12] G. Ghinita, P. Kalnis, A. Khoshgozaran, C. Shahabi, and K. L. Tan, "Private queries in location based services: Anonymizers are not necessary," in *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, pp. 121–132, 2008.

[13] O. Goldreich, S. Micali, and A. Wigderson, "How to play any mental game," in *Proceedings of the Nineteenth Annual ACM Symposium on Theory of Computing*, pp. 218–229, 1987.

[14] S. Goldwasser and S. Micali, "Probabilistic encryption & how to play mental poker keeping secret all partial information," in *Proceedings of the Fourth Annual ACM Symposium on Theory of Computing*, pp. 365–377, 1982.

[15] J. Groth, "Efficient maximal privacy in boardroom voting and anonymous broadcast," in *Financial Cryptography*, pp. 90–104, 2004.

[16] M. Gruteser and D. Grunwald, "Anonymous usage of location-based services through spatial and temporal cloaking," in *Proceedings of the 1st International Conference on Mobile Systems, Applications and Services*, pp. 31–42, 2003.

[17] F. Hao and P. Zieliński, "A 2-round anonymous veto protocol," in *Security Protocols*, pp. 202–211, 2009.

[18] S. Haykin, D. J. Thomson, and J. H. Reed, "Spectrum sensing for cognitive radio," *Proceedings of the IEEE*, vol. 97, no. 5, pp. 849–877, 2009.

[19] W. He, X. Liu, H. Nguyen, K. Nahrstedt, and T. T. Abdelzaher, "PDA: Privacy-preserving data aggregation in wireless sensor networks," in *IEEE INFOCOM'07*, pp. 2045–2053, 2007.

[20] L. Huang, K. Matsuura, H. Yamane, and K. Sezaki, "Enhancing wireless location privacy using silent period," in *IEEE Wireless Communications and Networking Conference*, vol. 2, pp. 1187–1192, 2005.

[21] L. Huang, H. Yamane, K. Matsuura, and K. Sezaki, "Silent cascade: Enhancing location privacy without communication qos degradation," in *Security in Pervasive Computing*, pp. 165–180, 2006.

[22] M. S. Hwang, C. H. Wei, and C. Y. Lee, "Privacy and security requirements for rfid applications," *Journal of Computers*, vol. 20, no. 3, pp. 55–60, 2009.

[23] IETF, "Tcp congestion control," RFC 2581, 2001. (https://tools.ietf.org/html/rfc2581)

[24] A. Kiayias and M. Yung, "Non-interactive zero-sharing with applications to private distributed decision making," in *Financial Cryptography*, pp. 303–320, 2003.

[25] H. Kido, Y. Yanagisawa, and T. Satoh, "An anonymous communication technique using dummies for location-based services," in *Proceedings of IEEE International Conference on Pervasive Services (ICPS'05)*, pp. 88–97, 2005.

[26] E. Kushilevitz and R. Ostrovsky, "Replication is not needed: Single database, computationally-private information retrieval," in *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, pp. 364–373, 1997.

[27] B. Lee, J. Oh, H. Yu, and J. Kim, "Protecting location privacy using location semantics," in *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1289–1297, 2011.

[28] C. T. Li and M. S. Hwang, "A lightweight anonymous routing protocol without public key en/decryptions for wireless ad hoc networks," *Information Sciences*, vol. 181, no. 23, pp. 5333–5347, 2011.

[29] S. Li, H. Zhu, Z. Gao, X. Guan, K. Xing, and X. Shen, "Location privacy preservation in collaborative spectrum sensing," in *IEEE INFOCOM*, pp. 729–737, 2012.

[30] H. Lu, C. S. Jensen, and M. L. Yiu, "PAD: privacy-area aware, dummy-based location privacy in mobile services," in *Proceedings of the Seventh ACM International Workshop on Data Engineering for Wireless and Mobile Access*, pp. 16–23, 2008.

[31] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam, "l-diversity: Privacy beyond k-anonymity," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 1, no. 1, 2007.

[32] J. Meyerowitz and R. Choudhury, "Hiding stars with fireworks: location privacy through camouflage," in *Proceedings of the 15th Annual ACM International Conference on Mobile Computing and Networking*, pp. 345–356, 2009.

[33] J. Mitola III, "Cognitive radio: An integrated agent architecture for software defined radio," *Doctoral Dissertation, KTH, Stockholm, Sweden*, May 2000.

[34] M. F. Mokbel, C. Y. Chow, and W. G. Aref, "The new Casper: query processing for location services without compromising privacy," in *Proceedings of the 32nd International Conference on Very Large Data Bases (VLDB'06)*, pp. 763–774, 2006.

[35] S. Papadopoulos, S. Bakiras, and D. Papadias, "pCloud: A distributed system for practical PIR," *IEEE Transactions on Dependable and Secure Computing*, vol. 9, no. 1, pp. 115–127, 2012.

[36] C. P. Schnorr, "Efficient signature generation by smart cards," *Journal of Cryptology*, vol. 4, no. 3, pp. 161–174, 1991.

[37] J. Shi, R. Zhang, Y. Liu, and Y. Zhang, "Prisense: privacy-preserving data aggregation in people-centric urban sensing systems," in *IEEE INFOCOM*, pp. 1–9, 2010.

[38] L. Sweeney, "k-anonymity: A model for protecting privacy," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 05, pp. 557–570, 2002.

[39] W. L. Tan, F. Lam, and W. C. Lau, "An empirical study on the capacity and performance of 3G networks," *IEEE Transactions on Mobile Computing*, vol. 7, no. 6, pp. 737–750, 2008.

[40] J. Trostle and A. Parrish, "Efficient computationally private information retrieval from anonymity or trapdoor groups," in *Information Security*, pp. 114–128. 2011.

[41] K. Vu, R. Zheng, and J. Gao, "Efficient algorithms for k-anonymous location privacy in participatory sensing," in *IEEE INFOCOM*, pp. 2399–2407, 2012.

[42] T. Wang and L. Liu, "Privacy-aware mobile services over road networks," *Proceedings of the VLDB Endowment*, vol. 2, no. 1, pp. 1042–1053, 2009.

[43] T. Xu and Y. Cai, "Exploring historical location data for anonymity preservation in location-based services," in *IEEE INFOCOM*, pp. 547–555, 2008.

[44] T. Xu and Y. Cai, "Feeling-based location privacy protection for location-based services," in *Proceedings of the 16th ACM Conference on Computer and Communications Security*, pp. 348–357, 2009.

[45] Z. Yang, G. Cheng, W. Liu, W. Yuan, and W. Cheng, "Local coordination based routing and spectrum assignment in multi-hop cognitive radio networks," *Mobile Networks and Applications*, vol. 13, pp. 67–81, 2008.

[46] A. C. C. Yao, "How to generate and exchange secrets," in *IEEE 27th Annual Symposium on Foundations of Computer Science*, pp. 162–167, 1986.

[47] W. Zhang, R. K. Mallik, and K. Letaief, "Optimization of cooperative spectrum sensing with energy detection in cognitive radio networks," *IEEE Transactions on Wireless Communications*, vol. 8, no. 12, pp. 5761–5766, 2009.

[48] Q. Zhao, S. Geirhofer, L. Tong, and B. M. Sadler, "Optimal dynamic spectrum access via periodic channel sensing," in *IEEE Wireless Communications and Networking Conference (WCNC'07)*, pp. 33–37, 2007.

[49] Y. Zheng, L. Wang, R. Zhang, X. Xie, and W. Y. Ma, "GeoLife: Managing and understanding your past life over maps," in *IEEE 9th International Conference on Mobile Data Management (MDM'08)*, pp. 211–212, 2008.

[50] Z. Zhi and Y. K. Choong, "Anonymizing geographic ad hoc routing for preserving location privacy," in *25th IEEE International Conference on Distributed Computing Systems Workshops*, pp. 646–651, 2005.

Erald Troja is currently a Ph.D. student at the Graduate Center of the City University of New York. He received his bachelor degree in Computer Science from Brooklyn College, NY, in 1999, his M.S in Information Systems and Business Administration from Brooklyn College, NY, in 2009 and M.S in Computer Science from the Graduate Center, CUNY in 2012. His major research areas include dynamic spectrum access, cognitive radio networks and security and privacy in location based services.

Spiridon Bakiras received the BS degree in Electrical and Computer Engineering from the National Technical University of Athens in 1993, the MS degree in Telematics from the University of Surrey in 1994, and the PhD degree in Electrical Engineering from the University of Southern California in 2000. Currently, he is an associate professor in the Department of Mathematics and Computer Science at John Jay College, City University of New York. Before that, he held teaching and research positions at the University of Hong Kong and the Hong Kong University of Science and Technology. His current research interests include database security and privacy, mobile computing, and spatiotemporal databases. He is a member of the ACM and a recipient of the US National Science Foundation (NSF)CAREER award.

Provably Secure Identity-Based Aggregate Signcryption Scheme in Random Oracles

Jayaprakash Kar
Information Security Research Group, Faculty of Computing & Information Technology
Department of Information Systems, King Abdulaziz University
P.O. Box 80221, Jeddah 21589, Kingdom of Saudi Arabia
(Email: jayaprakashkar@yahoo.com)
(Received Jan. 26, 2013; revised and accepted Jan. 5 & Feb. 6, 2014)

Abstract

This article proposes a provably secure aggregate signcryption scheme in random oracles. Security of the scheme is based on computational infeasibility of solving Decisional Bilinear Diffie-Hellman Problem and Discrete Logarithm Problems. Confidentiality and authenticity are two fundamental security requirements of Public key Cryptography. These are achieved by encryption scheme and digital signatures respectively. Signcryption is a cryptographic protocol that carries out signature and encryption simultaneously in a single logical step. An aggregate signcryption scheme can be constructed of the aggregation of individual signcryption. The aggregation is done taking n distinct signcryptions on n messages signed by n distinct users.

Keywords: Aggregate signature, BDHP, bilinear pairing, random oracle model

1 Introduction

In 1997, Zheng [22] introduced signcryption where signature and encryption are performed simultaneously in one logical step at lower computational costs and communication overheads than those required by the traditional sign-then-encrypt approach. Due to its advantages, there have been many signcryption schemes proposed after Zheng's publication. Baek *et al.* [1] shows Zheng's original schemes is provably secure in formal security model. Authentication, Confidentiality, non-repudiation and integrity are the strong security goals for many cryptographic applications. Applications must often contain at least two cryptographic primitives: signature, and encryption, which will definitely increase the corresponding computation and implementation complexity and even will be infeasible in some resources-constrained environments. To implement on low processor devices, Han *et al.* [6] introduced generalized signcryption scheme. It is feasible to implement joint encryption and signature functions in a single primitive.

2 Previous Works

Zheng [22] devised the principle of signcryption where both these encryption and signature are gained in a single logical step. Identity based cryptography was introduced by Shamir [15] in 1984 without obtaining the certificates for their public keys. In alternate, public keys are constructed taking user's IP address, telephone no, email addresses, social security numbers that distinctively identifies a user [9]. Trusted Third Party called Certificated Authority (CA) or Private Key Generator (PKG) generates the private key correspond to public key. Identity-based cryptography is supposed to provide a more suitable to traditional Public Key Infrastructure(PKI). Several practical identity-based signature schemes were proposed since 1984 with some vulnerability.

In 2001, Boneh and Franklin [2] first introduced fully practical identity based encryption scheme. Subsequently, many ID-based signcryption schemes have been proposed [7, 8, 10, 20, 21]. Yu *et al.* [18] proposed the first Identity based signcryption scheme in the standard model. But it was proved, that are insecure [16, 17, 19]. Also later on the schemes [18, 19] have proven these are insecure.

In 2002, Malone-Lee [12] proposed an efficient IBSC scheme by joining the function of of identity-based cryptography and signcryption. But this scheme is not semantically secure due to the visibility of the signature in the signcrypted message. This is proven by Libert and Quisquater [11]. Subsequently, Libert and Quisquater also proposed three different types of IBSC schemes which suit either forward security or public verifiability. Therefore to design an efficient signcryption scheme that proves both forward security and public verifiability was a great challenge in research community. To provide both the forward security and public verifiability, Chow *et al.* [5] constructed an Identity based Signcryption scheme. Boyen [3] proposed an IBSC scheme that provides ciphertext unlinkability and anonymity along with public verifiability and forward security. The improved version of this scheme was proposed by Chen and Malone-Lee [4] Barreto

et al. [13] which is provably secure and more efficient.

3 Preliminaries

3.1 Notation

Definition 1 [Bilinearity.] *Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be two cyclic group under the operation addition and multiplication. Both the groups are of same prime order p. Let e be an admissible bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ with the following properties:*

- **Bilinearity***: Let $P, Q \in \mathbb{G}$ and $a, b \in \mathbb{Z}_q^*$, $e(aP, bQ) = e(P, Q)^{ab}$, i.e for $P, Q, R \in \mathbb{G}_1, e(P + Q, R) = e(P, R)e(Q, R)$.*
- **Non-degenerate***: If P is a generator of $\mathbb{G}_1$, then $e(P, P)$ is generator of $\mathbb{G}_2$. $\exists\ P, Q \in \mathbb{G}$ such that $e(P, Q) \neq 1_{\mathbb{G}_2}$.*
- **Computability***: $\exists$ algorithm that compute $e(P, Q)$ in efficient way $\forall\ P, Q \in \mathbb{G}_1$.*

3.2 Mathematical Assumptions

Definition 2 [Decision Diffie-Hellman Problem (DDHP).] *Decide whether $c \equiv ab \bmod q$, for $a, b, c \in \mathbb{Z}_q^*$, given P, aP, bP, cP.*

Definition 3 [Computational Diffie-Hellman Problem (CDHP).] *Given P, aP, bP compute abP, for $a, b \in \mathbb{Z}_q^*$.*

Definition 4 [Bilinear Diffie-Hellman Problem.] *Let the algorithm $\mathcal{G}(k)$ generates 5 tuples $(q, \mathbb{G}_1, \mathbb{G}_2, e, P)$. Where $a, b, c \in \mathbb{Z}_q^*$. The problem in the group $\mathbb{G}$ is defined as: Given (P, aP, bP, cP) with $a, b, c \in \mathbb{Z}_q^*$, compute $e(P, P)^{abc} \in \mathbb{G}_T$. The (t, ϵ)-BDH assumption holds in $\mathbb{G}$ if $\nexists$ algorithm $\mathcal{A}$ running in time at most t such that*

$$\mathbf{Adv}_{\mathbb{G}}^{BDH}(\mathcal{A}) = Pr[\mathcal{A}(P, aP, bP, cP) = e(P, P)^{abc}] \geq \epsilon.$$

The probability is to find out taking all possible choices of (a, b, c) and is measured over the internal random operation of $\mathcal{A}$ and random choices of $a, b, c \in \mathbb{Z}_q^*$. Let us assume that BDHP is computationally infeasible to solve. Let the magnitude of q is $2k$, where k denotes a security parameter. There does not exist a polynomial time (in k) algorithm which has a non-negligible advantage in solving the BDHP, for all values of sufficiently large k. Following are the two variations of BDHP [9].

Definition 5 [Decisional Diffie-Hellman Problem.] *Let the probability is to find out taking all choices of (a, b, c, h). $\mathcal{G}(k)$ generates 5-tuples $(q, \mathbb{G}, \mathbb{G}_T, e, P)$. The problem is defined in the group $\mathbb{G}$ is given (P, aP, bP, cP, r) with some $a, b, c \in \mathbb{Z}_q^*$, if $r = e(P, P)^{abc}$ return* **yes***, otherwise* **no***. Where $a, b, c, r \in \mathbb{Z}_q^*$. The DBDHP in The (t, ϵ)-HDDH assumption holds in $\mathcal{G}$ if $\nexists$ an algorithm $\mathcal{A}$ with running time at most t such that*

$$\mathbf{Adv}_{\mathbb{G}}^{DBDH}(\mathcal{A}) = |Pr[\mathcal{A}(P, aP, bP, cP, e(P, P)^{abc})) = 1] - Pr[\mathcal{A}(P, aP, bP, cP, r) = 1]| \geq \epsilon.$$

Definition 6 [Hash Decisional Diffie-Hellman Problem.] *Let $\mathcal{G}(k)$ generates 5-tuple $(q, \mathbb{G}, \mathbb{G}_T, e, g)$. $\mathcal{H} : \{0,1\}^* \rightarrow \{0,1\}^l$ is a hash function, whether l is a security parameter, and let $a, b \in \mathbb{Z}_q^*, h \in \{0,1\}^l$, HDDH problem in $\mathbb{G}$ is defined as: Given (P, aP, bP, cP, h), decide whether it is a hash Diffie-Hellman tuple $((P, aP, bP, cP\mathcal{H}(e(P, P)^{abc}))$. Return 1, if it is correct, otherwise return 0. The (t, ϵ)-HDDH assumption holds in $\mathcal{G}$ if $\nexists$ algorithm $\mathcal{A}$ running in time at most t such that*

$$\mathbf{Adv}_{\mathbb{G}}^{HDDH}(\mathcal{A}) = |Pr[\mathcal{A}(P, aP, bP, cP\mathcal{H}(e(P, P)^{abc})) = 1] - Pr[\mathcal{A}(P, aP, bP, cP, h) = 1]| \geq \epsilon,$$

where the probability is taken over all possible choices of (a, b, h).

4 Framework of Aggregate Signcryption

An ID-based Aggregate Signcryption scheme (IDASC) comprises following probabilistic polynomial time solvable algorithms:

- **Setup**: $(param, msk) \leftarrow \text{Set}(1^k)$ takes $k \in \mathbb{N}$ the security parameter and generates $mask$ master secret key and $param$ global public parameters.
- **Key Extract**: $(<S_{ID_i}, d_i>, P_{pub}, q_i) \leftarrow Ext(1^k, param, msk, ID_i)$ takes $param$ global parameters, msk master secret key, k security parameter and identity of the sender ID_i to generate a private key $<S_{ID_i}, d_i>$ and public key P_{pub} and q_i.
- **Signcrypt**: $\sigma_i \leftarrow Signcrypt(1^k, param, m_i, X_i, d_i, ID_i, ID_B)$ takes k security parameter, $param$ global parameter and $(m_i, X_i, d_i, S_{ID_i}, ID_i, ID_B)$ to generate signcrypt σ_i. Let $\mathcal{M}$, $\mathcal{W}$ and $\mathcal{R}$ are space of message, space of signcrypted message and the space of sender respectively. Any member can be identified as U by its identity ID_U, where $U \in \mathcal{R}$.

  For any message $m_i \in \mathcal{M}, 1 \leq i \leq n, n \in \mathbb{Z}^+$.
- **Aggregate**: $\sigma \leftarrow Aggregate(\{\sigma_i, ID_i\}_{i=\ 1...n})$ The algorithm take the set of all signcryption $\{\sigma_i\}_{i=1...n}$ and the corresponding identity ID_i outputs the final aggregate signcryption σ.
- **UnSigncrypt**: $(\{m_i\}_{i=1...n}, Z_{agg}) \leftarrow UnSigncrypt(1^k, param, \sigma_{agg}, S_{ID_B}, d_B, ID_B)$ takes k a security parameter, $param$ the global parameters, σ_{agg} aggregate signcryption, S_{ID_B} receiver's secret and d_B to generate the plaintext m_i and signature Z_{agg}.
- **Verify**: $(Valid/\perp) \leftarrow Verify(1^k, param, \{m_i\}_{i=1...n}, Z_{agg}, S_{ID_B}, d_B)$. The algorithm takes k a security parameter, $param$ global parameters, m the message, Z_{agg} the signature and the

private key $<ID_B, d_B>$ outputs $Valid$ or $\perp$ for invalid signature.

5 Security Notions

Security of signcryption comprises two distinct techniques: providing authenticity and confidentiality or privacy. The two security goals can be provided by digital signature and encryption respectively. Under chosen ciphertext, we can say the indistinguishability of ciphertext with signature (signcrypt) or under chosen message attack, existential unforgeability of signcrypt. To achieve high level security, we concentrate on the above two forms of security.

Definition 7 [Confidentiality.] *An Identity-based signcryption scheme is said to be semantically secure or has indistinguishability against adaptive chosen ciphertext attack (IND-IDASC-CCA2) is there does not exist an adversary of polynomial bounded (PPT) with non-negligible advantage in the following game.*

1) **Initial**: **Setup** *is run by the challenger $\mathcal{C}$ taking the input of security parameter k. It returns the system parameter $param$ and master secret key msk. Master secret key msk is kept secret and send the system parameter $param$ to the adversary $\mathcal{A}$. The adversary $\mathcal{A}$ submits queries of polynomial bounded number of times to the oracles given to $\mathcal{A}$ by $\mathcal{C}$. In the first phase, execution of the queries are scheduled below:*

   - **Extraction oracle**: $<S_{ID_i}, d_i> \leftarrow Ext(mask, ID_i)$. *$\mathcal{A}$ submits ID_i extraction oracle and corresponding to the identity ID_i, get $<S_{ID_i}, d_i>$ as the private key pairs.*
   - **Signcryption oracle**: *$\mathcal{A}$ submits a message m_i, signer identity ID_i, and receiver identity ID_r to the challenger $\mathcal{C}$. $\mathcal{C}$ computes private key $<S_{ID_i}, d_i>$ for ID_i and runs the algorithm $Signcrypt(m_i, d_i, ID_i, ID_B)$ to obtain the signcryption σ_i. Finally $\mathcal{C}$ returns σ_i to $\mathcal{A}$.*
   - **UnSigncryption oracle**: *$\mathcal{A}$ submits the receiver identity $ID_B \notin \{ID_i\}_{i=1...n}$ to $\mathcal{C}$. $\mathcal{C}$ produces pair $<S_{ID_B}, d_B>$ as private key by submitting queries to the Key Extraction oracle. $\mathcal{C}$ unsigncrypts using the private key pairs $<S_{ID_B}, d_B>$ and returns the output to $\mathcal{A}$. If σ is an invalid signcrypted ciphertext returns a symbol $\perp$ for rejection from $\{ID_i\}_{i=1...n}$ to ID_B. $\mathcal{A}$ submits adaptively the queries to the oracle.*

2) *Let messages m_{i0}, m_{i1} are chosen by $\mathcal{A}$. Identities $\{ID_i\}_{i=1...n}$ and ID_B of sender and receiver on which $\mathcal{A}$ would like to be challenged. Two random bit $b \in \{0,1\}$ are chosen by the challenger $\mathcal{C}$ and computes the aggregate signcryption σ_{agg} by running $\sigma_i^* = Signcrypt(1^k, param, m_i, X_i, d_i, ID_i, ID_B)$ and aggregate algorithm $Aggregate(\{\sigma_i, ID_i\}_{i=1...n})$ and sends to $\mathcal{A}$.*

3) *Initially $\mathcal{A}$ performs polynomially bounded number of new queries with the restrictions that $\mathcal{A}$ cannot submit query to UnSigncryption oracle for the unsigncryption of σ_{agg}^* or the Keygen oracles for the private keys pairs of ID_B^*.*

4) *$\mathcal{A}$ returns a bit b' and if $b' = b$, then wins the game at the end of the game. The success probability is:*

$$Adv^{(IDASC-IND-CCA2)}(\mathcal{A}) = |Pr[b' = b] - \tfrac{1}{2}|,$$

where Adv denotes advantage for the adversary.

Definition 8 [Signature Unforgeability.] *An identity based aggregate signcryption scheme (IDASC) is said to be existentially signature unforgeable against adaptive chosen-messages attacks (EUF-IDASC-CMA) if no polynomial bounded adversary has a non-negligible advantage in the following game:*

1) *The algorithm* **Setup** *is run by the challenger $\mathcal{C}$ taking input k as security parameter and sends $param$ the system parameters to the adversary $\mathcal{A}$ and keeps secret $mask$ the master private key.*

2) *$\mathcal{A}$ performs polynomial bounded number of queries to the same oracles described in IDASC-IND-CCA2 game which are simulated by the challenger $\mathcal{C}$. The queries may be run in adaptive manner.*

   The adversary $\mathcal{A}$ returns a recipient identity ID_B and a ciphertext σ_i $\mathcal{A}$ submits a signcryption ciphertext σ_i and two identity ID_B^* and ID_i^*, $\mathcal{A}$ wins the game if the ciphertext σ_i is decrypted as a signed message (ID_i, m_i^*, V_i^*) having $ID_i \neq ID_B, ID_i \in \{ID_i\}_{i=1...n}$ result of the $UnSingncrypt(\sigma_{agg}, S_{ID_B}, d_B)$, otherwise returns the symbol $\perp$. Formally it can be defined as:

   - *$(\{m_i^*\}_{i=1...n}, Z_{agg}^*) \leftarrow UnSigncrypt(1^k, param, \sigma_{agg}^*, S_{ID_B^*}, d_B^*, ID_B^*)$ takes k security parameter, $param$ the global parameters, σ_{agg} aggregate signcryption, secret key of the receiver S_{ID_B} and d_B to generate the plaintext m_i and signature Z_{agg}. i.e. $\mathcal{A}$ submit a signcryption ciphertext σ_{agg}^*, global parameters $param$, k and identity ID_B^* returns $\{m_i^*\}_{i=1...n}, Z_{agg}^*$ such that $valid \leftarrow Verify(m_i^*, \sigma^*, \{ID_i^*\}_{i=1...n})$.*
   - *There will be no signcryption oracle decrypts to (m^*, σ^*) such that $valid \leftarrow Verify(m^*, \sigma^*, \{ID_i^*\}_{i=1...n})$.*
   - *No extra query was made on $\{ID_i^*\}_{i=1...n}$.*

   $\mathcal{A}$'s advantage is defined as

$$Adv_{\mathcal{A}}^{EUF-IDASC-CMS} = Pr[Verify(m_i^*, \sigma^*, \{ID_i^*\}_{i=1...n}) = Valid]$$

Definition 9 [Ciphertext Unforgeability.] *An ID-based aggregate signcryption scheme (IDASC) is said to be existentially ciphertext unforgeable against adaptive chosen-messages attacks (AUTH-IDASC-CMA) if no polynomial bounded adversary (PPT) has a non-negligible advantage in the following game:*

1) *The* **Setup** *algorithm is run by the challenger $\mathcal{C}$ taking the input k as security parameter and sends param the system parameters to the adversary $\mathcal{A}$ and keeps secret msk the master private key.*

2) *The adversary $\mathcal{A}$ performs polynomial bounded number of queries to the oracles provided to $\mathcal{A}$ by $\mathcal{C}$. The attack may be conducted in adaptive manner and allows as in queries described in (IND-IDASC-CCA2) game.*

3) *Forgery. The adversary $\mathcal{A}$ produces a new aggregate signcryption σ_{agg} from a set $\{ID_i\}_{i=1\ldots n}$ of n users on messages $m_i, \forall i = 1\ldots n$ to a final receiver $ID_B \notin \{ID_i\}_{i=1\ldots n}$, where the private keys of the users in $\{ID_i\}_{i=1\ldots n}$ was not queried in query phase and σ_i is not the output of a previous query to the Signcrypt queries. Outcome. The adversary $\mathcal{A}$ wins the game if $\perp$ is not returned by $UnSigncrypt(1^k, param, \sigma_{agg}, S_{ID_B}, d_B, ID_B)$.*

6 ID-based Aggregate Signcryption Scheme

The scheme comprises five randomized polynomials algorithms.

- **Setup.** The algorithm take k the security parameter. Groups $\mathbb{G}_1$ and $\mathbb{G}_2$ of prime order q are chosen by **PKG**. A generator P of $\mathbb{G}_1$, a bilinear map $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ and collision resistant hash function $\mathcal{H}_0 : \{0,1\}^* \to \mathbb{F}_q^*$, $\mathcal{H}_1 : \mathbb{G}_2 :\to \{0,1\}^l \times \mathbb{F}_q^*$, $\mathcal{H}_2 : \{0,1\}^l \times \{0,1\}^* \times \mathbb{G}_1 \times \mathbb{G}_1 \times \{0,1\}^* \times \mathbb{G}_1 \to \mathbb{F}_q^*$, $\mathcal{H}_3 : \{0,1\}^l \times \{0,1\}^* \times \mathbb{G}_1 \times \mathbb{F}_q^* \times \mathbb{G}_1 \times \{0,1\}^* \times \mathbb{G}_1 \to \mathbb{F}_q^*$. It chooses a master-key $s \in \mathbb{F}_q^*$ and computes $P_{pub} = sP$. System parameters are published by **PKG**.

$$\mathcal{P} = (\mathbb{G}_1, \mathbb{G}_2, n, \hat{e}, P, P_{pub}, \mathcal{H}_0, \mathcal{H}_1, \mathcal{H}_2, \mathcal{H}_3).$$

- **Extract.** The algorithm follows
  - Given an identity $ID_i \in \{0,1\}^*$, PKG computes $Q_{ID_i} = \mathcal{H}_0(ID_i)$ and the partial private key as $S_{ID_i} = s \cdot Q_{ID_i}$.
  - Chooses a random number $x_i \leftarrow_R \mathbb{F}_q^*$ and computes $X_i = x_i \cdot P$.
  - Computes $d_i = (x_i + sq_i) \bmod q$, for all $i = 1\ldots n$. corresponding public key $q_i = \mathcal{H}_0(ID_i \| X_i)$.
  - The PKG sends the corresponding private key $<S_{ID_i}, d_i>$ and public key $<X_i, q_i>$ through a secure channel to the users.

- **Signcrypt.** $(m_i, X_i, d_i, ID_i, ID_B)$: The algorithm works as follows
  - Chooses $r_i \leftarrow_R \mathbb{F}_q^*$ randomly and calculate $W_i = r_i \cdot P, w_i = \hat{e}(P_{pub}, Q_{ID_B})^{r_i}$.
  - Computes $h_{1i} = \mathcal{H}_1(w_i), h_{2i} = \mathcal{H}_2(m_i, ID_i, X_i, w_i, ID_B, X_B)$.
  - Computes $h_{3i} = \mathcal{H}_3(m_i, ID_i, X_i, w_i, ID_B, X_B, h_{2i})$.
  - Computes $v_i = (r_i h_{2i} + h_{3i} d_i) \bmod q$.
  - Computes $C_i = (m_i \| v_i) \oplus h_{2i}, Z_i = v_i \cdot P$.
  - Output $\sigma_i = <C_i, W_i, Z_i, X_i>$ is the signcryption of ID_i on message m_i.

- **Aggregate.** $(\{\sigma_i, ID_i\}_{i=1\ldots n})$: On input a set of signcryption $\sigma_i = <C_i, W_i, Z_i, X_i>$, $i = 1\ldots n$ and the corresponding identity ID_i such that $\forall i = 1\ldots n, \sigma_i$ are the signcryption of message m_i by ID_i.
  1) $Z_{agg} = \sum_{i=1}^{n} Z_i$, $Z_i = v_i \cdot P$, $i = 1\ldots n$;
  2) Output the final aggregate signcryption $\sigma_{agg} = <\{C_i, W_i, X_i, ID_i\}_{i=1\ldots n}, Z_{agg}>$.

  The aggregate can be computed by the sender or a trusted third party.

- **UnSigncrypt.** $(\sigma_{agg}, S_{ID_B}, d_B)$: To decrypt and verify the aggregate signcryption $\sigma_{agg} = <\{C_i, W_i, X_i, ID_i\}_{i=1\ldots n}, Z_{agg}>$, the receiver with identity ID_B use his private key $<S_{ID_B}, d_B>$ and follows the following steps.
  - Computes $C_i \oplus h_{1i} = m_i \| v_i$, where $h_{1i} = \mathcal{H}_1(w_i), w_i = \hat{e}(W_i, S_{ID_B})$.
  - $\forall i = 1\ldots n$, computes $h_{2i} = \mathcal{H}_2(m_i, ID_i, X_i, w_i, ID_B, X_B)$.
  - Verify the validity of the following equation

$$\begin{aligned}
w_i &= \hat{e}(W_i, S_{ID_B}) = \hat{e}(r_i P, S_{ID_B}) \\
&= \hat{e}(P, S_{ID_B})^{r_i} \\
&= \hat{e}(P, sQ_{ID_B})^{r_i} = \hat{e}(sP, Q_{ID_B})^{r_i} \\
&= \hat{e}(P_{pub}, Q_{ID_B})^{r_i}.
\end{aligned}$$

$$\begin{aligned}
Z_{agg} &= \sum_{i=1}^{n}(v_i \cdot P) = \sum_{i=1}^{n}(r_i h_{2i} + h_{3i} d_i) \cdot P \\
&= \sum_{i=1}^{n} h_{2i}(r_i \cdot P) + \sum_{i=1}^{n} h_{3i}(d_i \cdot P) \\
&= \sum_{i=1}^{n} h_{2i}(r_i \cdot P) + \sum_{i=1}^{n} h_{3i}(x_i + sq_i) \cdot P \\
&= \sum_{i=1}^{n}(h_{2i} W_i) + \sum_{i=1}^{n}(h_{3i} X_i) \\
&\qquad + P_{pub} \sum_{i=1}^{n}(h_{3i} q_i).
\end{aligned}$$

7 Proof of Correctness

7.1 Security Analysis

Our scheme is secure in IDASC-IDASC-CCC2, AUTH-IDASC-CMA and EUF-IDASC-CMA defined in Definitions 7, 8 and 9. We prove the following theorem as proved in [14].

Theorem 1 *In random oracle model, we assume the adversary $\mathcal{A}$ for $IND-IDASC-CCA2$ is able to distinguish two valid ciphertext during the game with a non-negligible advantage and run $Keygen$ queries, $Signcrypt$ queries, and $Unsigncrypt$ queries; then there exists a distinguisher $\mathcal{B}$ that can solve an instances of Decisional Bilinear Diffie-Hellman problem with a non-negligible advantage.*

Proof.

- **Setup**: The distinguisher $\mathcal{B}$ receives a random instance (P, aP, bP, cP, μ) of the Decisional Bilinear Diffie-Hellman problem and decide validity of $\mu = \hat{e}(P,P)^{abc}$. $\mathcal{B}$ executes $\mathcal{A}$ as a subroutine and proceeds $\mathcal{A}$s challenger in the IND-IDASC-CCA2 game. A lists L_0, L_1, L_2 and L_3 are set by $\mathcal{B}$. These are initial empty. $\mathcal{A}$ submits queries to the respective oracles $\mathcal{H}_0, \mathcal{H}_1, \mathcal{H}_2, \mathcal{H}_3$ and place the answers in the corresponding list.

- **Oracle Simulation**:

  1) $\mathcal{H}_0$**-Oracle**: At the beginning of the game $\mathcal{B}$ submits the system parameters to $\mathcal{A}$ by computing $P_{pub} = cP$ ($\mathcal{B}$ does not know c and act function of master-key). Then $\mathcal{B}$ chooses two distinct random numbers $i, j \in \{1 \dots q_{\mathcal{H}_0}\}$. $\mathcal{A}$ asks a polynomial bounded number of $\mathcal{H}_0$ requests on identities of his choice. At the i^{th} $\mathcal{H}_0$ request, $\mathcal{B}$ answers by $\mathcal{H}_0(ID_i) = aP$. At the j^{th}, he answers by $\mathcal{H}_0(ID_j) = bP$. Since aP and bP belong to a random instance of the DBDH problem, $\mathcal{A}$s view will not be modified by these changes. Hence, the private keys S_{ID_i} and S_{ID_j} (which are not computable by $\mathcal{B}$) are respectively acP and bcP. Thus the solution $\hat{e}(P,P)^{abc}$ of the BDH problem is given by $\hat{e}(Q_{ID_i}, S_{ID_j}) = \hat{e}(S_{ID_i}, Q_{ID_j})$. For requests $\mathcal{H}_0(ID_k)$ with $k \neq i, j$, $\mathcal{B}$ chooses $b_k \leftarrow_R \mathbb{F}_q^*$, puts the pair (ID_k, b_k) in list L_0 and answers $\mathcal{H}_0(ID_k) = b_k P$.

     Further on input $ID_i \in \{0,1\}^*$, $\mathcal{B}$ first checks the L_0-list $<ID_i, X_i, q_i, x_i>$, if $ID_i = ID_B>$, selects new random $\gamma_i \leftarrow_R \mathbb{F}_q^*$, sets $X_i = b \cdot P, q_i = \gamma_i$, add this tuple $<ID_i, X_i, q_i, *>$ to the L_0-list and returns q_i. Otherwise, $\mathcal{B}$ selects a new random $\gamma_i \leftarrow_R \mathbb{F}_q^*$, $x_i \leftarrow_R \mathbb{F}_q^*$, sets $X_i = x_i \cdot P, q_i = \gamma_i$, add this tuple $<ID_i, X_i, q_i, x_i>$ to the L_0-list and returns q_i.

  2) $\mathcal{H}_1$**-Oracle**: When a (m_i, ID_i, X_i, w_i, ID_B, X_B) is submitted in $\mathcal{H}_1$ query for the first time, $\mathcal{B}$ returns checks the L_1-list, whether the tuples $<w_i, h_{1i}>$ in L_1-list, $\mathcal{B}$ returns h_{1i}, otherwise, $\mathcal{B}$ chooses a new random $h_{1i} \leftarrow_R \mathbb{F}_q^*$, includes the tuples $<w_i, h_{1i}>$ to the L_1-list and return h_{1i}.

  3) $\mathcal{H}_2$**-Oracle**: On input (m_i, ID_i, X_i, w_i, ID_B, X_B), $\mathcal{B}$ first checks the L_2-List, whether the tuple $<m_i, ID_i, X_i, W_i, ID_B, X_B, h_{2i}>$ in the L_2-List, $\mathcal{B}$ returns h_{2i}, otherwise $\mathcal{B}$ chooses a new random $h_{2i} \leftarrow_R \mathbb{F}_q^*$, includes h_{2i} to the L_2-list and return h_{2i}.

  4) $\mathcal{H}_3$**-Oracle**: On input (m_i, ID_i, X_i, w_i, ID_B, X_B, h_{2i}), $\mathcal{B}$ first checks the L_3-List, whether the tuple $<m_i, ID_i, X_i, W_i, ID_B, X_B, h_{2i}>$ in the L_3-List, $\mathcal{B}$ returns h_{3i}, otherwise $\mathcal{B}$ chooses a new random $h_{3i} \leftarrow_R \mathbb{F}_q^*$, includes h_{3i} to the L_3-list and return h_{3i}.

  5) **Keygen-Oracle**: When $\mathcal{A}$ makes a $Keygen$ query with ID_i as the input, $\mathcal{B}$ checks the L_0-List to verify whether or not there is an entry for ID_i. If the L_0-List does not contain an entry for ID_i, return $\perp$. Otherwise, if $ID_i = ID_B$, $\mathcal{B}$ recovers the tuple $<ID_i, X_i, q_i, x_i>$ from the L_0-List and returns $<X_i, q_i, *, *>$, if $ID_i \neq \{ID_i\}_{i=1\dots n}$ $\mathcal{B}$ recovers the tuple $<ID_i, X_i, q_i, x_i>$ from the L_0-List and returns $<X_i, q_i, S_{ID_i}, d_i>$, where $S_{ID_i} = x_i(aP) = a(x_i P) = aX_i$ and $d_i \leftarrow_R \mathbb{F}_q^*$ is randomly selected.

  6) **Signcryption Oracle**: When $\mathcal{A}$ makes a Signcrypt query with ID_i as the input, $\mathcal{B}$ checks the L_0-List to verify whether or not there is an entry for ID_i. If the L_0-List does not contain an entry for ID_i returns$\perp$. Otherwise, $\mathcal{B}$ executes $Signcrypt(m_i, X_i, d_i, ID_i, ID_B)$ as usual and returns what the Signcrypt algorithm returns.

  7) **UnSigncryption Oracle**: When $\mathcal{A}$ makes an $Unsigncrypt$ query with $\sigma_{agg} = <\{C_i, W_i, X_i, ID_i\}_{i=1,\cdots,n}, Z_{agg}>$ and the receiver with identity ID_B, $\mathcal{B}$ first verifies whether or not there are entries for $ID_i, (ID_i \neq ID_B)$ and ID_B in L_0-List and there is an entry of the form $<ID_i, X_i, q_i, \gamma_i>$. If at least one of these conditions is not satisfied, $\mathcal{B}$ returns$\perp$. Otherwise, $\mathcal{B}$ executes $Unsigncrypt(\sigma_{agg}, S_{ID_B}, d_B)$ in the normal way and returns what the $Unsigncrypt$ algorithm returns.

After getting sufficient training, $\mathcal{A}$ submits two equal length of messages m_{i0} and m_{i1}. $\mathcal{A}$ randomly chooses a bit $b^* \leftarrow \{0,1\}$ and return ciphertext of the challenged signcryption running $Signcrypt(m_{ib^*}, X_i, d_i, ID_i, ID_B)$ and $Aggregate(\{\sigma_i^*, ID_i\}_{i=1\dots n})$, then returns σ_{agg}^* to $\mathcal{A}$.

- **Output**: $\mathcal{A}$ returns the presumed bit after submitting adequate number of queries. Then *mathcalB* solve BDH problem and returns '1'. Else, it returns '0'. Since the adversary is denied access to the Unsigncrypt oracle with the challenge signcryption, for $\mathcal{A}$ to find that m_i is not a valid ciphertext, $\mathcal{A}$ should have queried the $\mathcal{H}_1$ Oracle with $w_i = e(W_i, S_{ID_B})$. Here S_{ID_B} is the private key of the receiver, and it is $aX_B = (bP)a = abP$. Also, $\mathcal{B}$ has set $W_i = cP$. We have $w_i = e(W_i, S_{ID_B}) = e(cP, abP) = e(P, P)^{abc}$.

□

Theorem 2 *Assume Elliptic Curve Discrete Logarithm Problem is computationally infeasible to solve in* $\mathbb{G}_1$. *The proposed ASC is secure against any probabilistic polynomial time adversary* $\mathcal{A}$ *for AUTH-IDASC-CMA in the random oracle model.*

Proof. $\mathcal{B}$ receives a random instance $(P, W_{r_\alpha}) = r_\alpha P$ and $(P, d_\alpha P)$ of ECDLP as a challenge in the AUTH-IDASC-CMA game defined in Definition 2. His goal is to determine r_α and d_α. $\mathcal{B}$ will run $\mathcal{A}$ as a subroutine and act as $\mathcal{A}$s challenger in the AUTH-IDASC-CMA game. $\mathcal{A}$ can compute $d_\alpha P$ as $W_\alpha + (sP)q_\alpha$, $d_\alpha P = (x_\alpha + sq_\alpha)\cdot P = W_\alpha + (sP)q_\alpha$.

- $\mathcal{H}_0$ **Oracle**: For $\mathcal{H}_0$-queries on input $ID_i \in \{0,1\}^*$, $\mathcal{B}$ first checks the L_0-list $<ID_i, X_i, q_i, x_i>$, selects random $\gamma_i \leftarrow_R \mathbb{F}_q^*$, sets $X_i = x_i \cdot P, q_i = \gamma_i$, add this tuple $<ID_i, X_i, q_i, *>$ to the L_0-list and returns q_i.

- **Keygen Oracle**: When $\mathcal{A}$ submits a *Keygen* query with ID_i as the input, $\mathcal{B}$ checks the L_0-List to verify whether or not there is an entry for ID_i. If no entry for ID_i belongs to the L_0-List, return $\perp$. Otherwise, if $ID_i \in \{ID_i\}_{i=1\ldots n}$, $\mathcal{B}$ recovers the tuple $<ID_i, X_i, q_i, x_i>$ from the L_0-List and returns $<X_i, q_i, *, *>$, if $ID_i \neq \{ID_i\}_{i=1\ldots n}$ $\mathcal{B}$ recovers the tuple $<ID_i, X_i, q_i, x_i>$ from the L_0-List and returns $<X_i, q_i, S_{ID_i}, d_i>$, where $S_{ID_i} = x_i(aP) = a(x_iP) = aX_i$ and $d_i \leftarrow_R \mathbb{F}_q^*$ is randomly selected.

- **Forgery**: $\mathcal{A}$ chooses the corresponding senders identities set $\{ID_i\}_{i=1\ldots n}$ and receiver identity ID_B and returns a forged signcryption $\sigma_\alpha^* = <C_\alpha^*, W_\alpha^*, Z_\alpha^*, X_\alpha^*>$ on message m_α^* from $ID_\alpha \in \{ID_i\}_{i=1\ldots n}$ to $\mathcal{B}$. $\mathcal{B}$ retrieves the entry corresponding to ID_B in the L_0-List and uses s_B to execute $Unsigncrypt(\sigma_{agg}, S_{ID_B}, d_B)$. If σ_α^* is a valid signcryption from ID_α to receiver ID_B, that is, a message m_α^* is returned by the *Unsigncrypt* algorithm, then $\mathcal{B}$ applies the oracle replay technique to produce two valid signcryptions $\sigma_\alpha^{'} = <C_\alpha^{'}, W_\alpha^{'}, Z_\alpha^{'}, X_\alpha^{'}>$ and $\sigma_\alpha^{''} = <C_\alpha^{''}, W_\alpha^{''}, Z_\alpha^{''}, X_\alpha^{''}>$ on message m_α from the ID_α to receiver ID_B. $\mathcal{B}$ obtains the signatures as $v_\alpha^{'} = r_\alpha h_{2\alpha}^{'} + h_{3\alpha}^{'} d_\alpha$ and $v_\alpha^{''} = r_\alpha h_{2\alpha}^{''} + h_{3\alpha}^{''} d_\alpha$ with $h_{2\alpha}^{'} \neq h_{2\alpha}^{''}$ and $h_{3\alpha} \neq h_{3\alpha}^{''}$. The PPT algorithm $\mathcal{B}$ can computes r_α and d_α as

$$r_\alpha = \frac{v_\alpha^{'} h_{3\alpha}^{''} - v_\alpha^{''} h_{3\alpha}^{'}}{h_{2\alpha}^{'} h_{3\alpha}^{''} - h_{2\alpha}^{''} h_{3\alpha}^{'}}, h_{2\alpha}^{'} h_{3\alpha}^{''} - h_{2\alpha}^{''} h_{3\alpha}^{'} \neq 0.$$

$$d_\alpha = \frac{v_\alpha^{'} h_{2\alpha}^{''} - v_\alpha^{''} h_{2\alpha}^{'}}{h_{3\alpha}^{'} h_{2\alpha}^{''} - h_{3\alpha}^{''} h_{2\alpha}^{'}}, h_{3\alpha}^{'} h_{2\alpha}^{''} - h_{3\alpha}^{''} h_{2\alpha}^{'} \neq 0.$$

□

Theorem 3 *Assume Decisional Biliner Diffie-Hellman Problem is computationally infeasible to solve in* $\mathbb{G}_1$. *The proposed ASC is secure against any probabilistic polynomial time adversary* $\mathcal{A}$ *for EUF-IDASC-CMA in the random oracle model.*

Proof. $\mathcal{B}$ simulates the $\mathcal{A}$'s challenger in the EUF-IDASC-CMA game. $\mathcal{B}$ can perform queries as defined in Definition-9. we describe the process as follows.

Keygen Oracle: When $\mathcal{A}$ submits a *Keygen* query with ID_i as the input, $\mathcal{B}$ checks the L_0-List to verify whether or not there is an entry for ID_i. If no entry for ID_i belongs to the L_0-List, return $\perp$. Otherwise, if $ID_i = ID_\alpha$, $\mathcal{B}$ recovers $<ID_i, X_i, q_i, x_i>$ from the L_0-List and returns $<X_i, q_i, *, *>$, if $ID_i \neq ID_\alpha$ $\mathcal{B}$ recovers the tuple $<ID_i, X_i, q_i, x_i>$ from the L_0-List and returns $<X_i, q_i, S_{ID_i}, d_i>$, where $S_{ID_i} = x_i(sP)$ and $d_i \leftarrow_R \mathbb{F}_q^*$ is randomly selected.

Eventually, $\mathcal{A}$ returns a forgery, consisting of a ciphertext and a recipient identity ID_B. $\mathcal{B}$ decrypts the ciphertext for ID_B (by invoking its own decryption oracle), which causes the plaintext forgery (ID_i, m_i, V_i) to be revealed. Note that if $\mathcal{B}$ has made the correct guess, that is, $ID_i = ID_\alpha$, then $ID_B \neq ID_\alpha$ and the decryption works.

Let the valid signcryption σ_i is sent from ID_i to ID_B which is, a message m_i is generated by the *Unsigncrypt* algorithm. $\mathcal{B}$ submits the queries to the oracle by applying replay technique return two valid signed messages (ID_i, m_i, V_i) and (ID_i, m_i, V_i) on a message m_i from the ID_i to receiver ID_B. With the same random tape but with a different hash value, this is provided by running the truing machine again $\mathcal{B}$ obtains the signatures $v_\alpha^{'} = r_\alpha h_{2\alpha}^{'} + h_{3\alpha}^{'} d_\alpha$ and $v_\alpha^{''} = r_\alpha h_{2\alpha}^{''} + h_{3\alpha}^{''} d_\alpha$ with $h_{2\alpha}^{'} \neq h_{2\alpha}^{''}$ and $h_{3\alpha}^{'} \neq h_{3\alpha}^{''}$. □

8 Comparison

Let symbolize confidentiality (Con), unforgeability (Unf), public verifiability (PuV), forward security (FoS), ciphertext unlinkability (CiU) and ciphertext anonymity (CiA). "√" and "×" denotes Yes and No respectively. Table 1 shows the security comparison among IDASC and others.

Efficiency of aggregate sincryption scheme can be evaluated with respect to computational cost and ciphertext length [14]. To compute the computational cost, we consider scalar multiplications, exponentiations and pairing

Table 1: Security comparison

Schemes	$Conf$	Unf	PuV	Fos	CiU	CiA
Libert and Quisquater(I)	√	√	√	√	×	×
Libert and Quisquater(II)	√	√	√	√	×	×
Libert and Quisquater(III)	√	√	√	×	√	×
Malone-Lee	×	√	√	√	×	×
Barreto *et al.*	√	√	√	√	×	×
Boyen	√	√	√	√	√	√
Chow *et al.*	√	√	√	√	×	×
IDASC	√	√	√	√	√	√

Table 2: Comparison of computational cost

	$Signcrypt$			$UnSigncrypt$		
	$Pairing$	$Mul(\mathbb{G}_1)$	$Exp(\mathbb{G}_2)$	$Pairing$	$Mul(\mathbb{G}_1)$	$Exp(\mathbb{G}_2)$
Libert and Quisquater(I)	1(+1)	2	2	4		2
Libert and Quisquater(II)	1(+1)	2	2	4		2
Libert and Quisquater(III)	1	2	1	2	1	
Malone-Lee	1	3		4		1
Barreto *et al.*		2	1	2	1	1
Boyen	1	3	1	4	2	
Chow *et al.*	2	2		4	1	
IDASC	1	2		1	1	

Table 3: Comparison of ciphertext size

Scheme	Ciphertext size
Selvi *et al.* and Boneh *et al.*	$\mid M \mid + \mid \mathbb{Z}_q^* \mid +3 \mid \mathbb{G}_1 \mid$
Ren *et al.*	$\mid M \mid + \mid \mathbb{Z}_q^* \mid +4 \mid \mathbb{G}_1 \mid$
IDASC	$\mid M \mid + \mid \mathbb{Z}_q^* \mid +2 \mid \mathbb{G}_1 \mid$

operation are costly operation. Let scalar multiplication in $\mathbb{G}_1$ is denoted by $(Mul(\mathbb{G}_1))$, exponentiations in $\mathbb{G}_2$ is $(Exp(\mathbb{G}_2))$, and pairing operations $(Pairing)$. Tables 2 and 3 show the comparison among IDASC and others with computational cost and ciphertext size items, respectively.

9 Conclusion

Here we have proposed an efferent and secure aggregate signcryption scheme which is more efficient than the scheme proposed by Xun-Yi Ren *et* al. [14] with respect to the length of Ciphertext and secure than the other schemes summarized in the tables. We prove that the scheme in Random oracle model and proven that the scheme achieve the three strong security goals confidentiality, signature unforgeability and ciphertext unforgeability under the assumption, ECDLP and BDHP are computationally hard. Since our scheme is compact, fast and unforgeable, in real time application such as key transport, multi cast electronics commerce, authenticated e-mail, it can be applied.

References

[1] J. Baek, R. Steinfeld, and Y. Zheng, "Formal proofs for the security of signcryption," *Journal of Cryptology*, vol. 20, no. 2, pp. 203–235, 2007.

[2] D. Boneh and M. K. Franklin, "Identity-based encryption from the weil pairing," in *Proceedings of Advances in Cryptology (Crypto'01)*, LNCS 2139, pp. 213–229, Springer-Verlag, 2001.

[3] X. Boyen, "Multipurpose identity-based signcryption," in *Proceedings of Advances in Cryptology (Crypto'03)*, LNCS 2729, pp. 383–399, Springer-Verlag, 2003.

[4] L. Chen and J. Malone-Lee, "Improved identity-based signcryption," in *Proceedings of Public Key Cryptography (PKC'05)*, LNCS 3386, pp. 362–379, Springer-Verlag, 2005.

[5] S. S. M. Chow, S. M. Yiu, L. C. K. Hui, and K. P. Chow, "Efficient forward and provably secure ID-based signcryption scheme with public verifiability and public ciphertext authenticity," in *Security and Cryptology (ICISC'03)*, LNCS 2971, pp. 352–369, Springer-Verlag, 2004.

[6] Y. Han and X. Yang, "Elliptic curve based generalized signcryption scheme," Technical Report IACR Archive ePrint-2006/126, Sep. 2006.

[7] Z. P. Jin, Q. Y. Wen, and H. Z. Du, "An improved semantically-secure identity-based signcryption scheme," *Standard model Computers & Electrical Engineering*, vol. 36, no. 2, pp. 545–552, 2010.

[8] J. Kar. "An efficient signcryption scheme from q-diffe-hellman problems,". Technical Report IACR Archive ePrint-2012/483, Sep. 2012.

[9] J. Kar, "Provably secure identity based online/offline signature scheme for wireless sensor network," *International Journal of Network Security*, vol. 16, no. 1, pp. 29–39, 2014.

[10] F. Li, X. Xin, and Y. Hu, "ID-based signcryption scheme with (t, n) shared unsigncryption," *International Journal of Network Security*, vol. 3, no. 2, pp. 155–159, 2006.

[11] B. Libert and J. J. Quisquater, "A new identity based signcryption schemes from pairings," in *Proceedings of IEEE Information Theory Workshop*, vol. 435, pp. 155–158, Paris, France, June 2003.

[12] J. Malone-lee, "Identity-based signcryption," in *Proceedings of Public Key Cryptography*, LNCS 3386, pp. 362–379, Springer-Verlag, 2005.

[13] N. McCullagh P. S. L. M. Barreto, B. Libert and J. J. Quisquater, "Efficient and provably-secure identity-based signatures and signcryption from bilinear maps," in *Proceedings of Advances in Cryptology (Asiacrypt'05)*, LNCS 3788, pp. 383–399, Springer-Verlag, 2005.

[14] X. Y. Ren, Z. H. Qi, and Geng, "Provably secure aggregate signcryption scheme," *ETRI Journal*, vol. 34, no. 3, pp. 421–428, 2012.

[15] A. Shamir, "Identity-based cryptosystem and signature schemes," in *Proceedings of Advances in Cryptology (Crypto'84)*, LNCS 3386, pp. 47–53, Springer-Verlag, 1984.

[16] M. Toorani and A. A. B. Shirazi, "Cryptanalysis of an elliptic curve-based signcryption scheme," *International Journal of Network Security*, vol. 10, no. 1, pp. 51–56, 2010.

[17] X. Wang and H. F. Qian, "Attacks against two identity-based signcryption schemes," in *Proceeding of Second International Conference on Networks Security, Wireless Communications and Trusted Computing*, vol. 1, pp. 24–27, Paris, France, June 2010.

[18] Y. Sun Y. Yu, B. Yang and S. Zhu, "Identity based signcryption scheme without random oracles," *Elsevier-Computer Standards & Interfaces*, vol. 31, no. 1, pp. 56–62, 2009.

[19] R. Yanli and G. Dawu, "Efficient identity based signature/signcryption scheme in the standard model," in *Proceeding of First International Symposium on Data, Privacy, and E-Commerce*, vol. 3386, pp. 133–137, Paris, France, June 2007.

[20] G. Yu, X. X. Ma, and Y. Shen, "Provable secure identity based generalized signcryption scheme," *Theoretical Computer Science*, vol. 411, no. 40, pp. 3614–3624, 2010.

[21] B. Zhang and Q. L. Xu, "An ID-based anonymous signcryption scheme for multiple receivers secure in the standard model," in *Proceedings of Computer Science and Information Technology*, vol. 1294, pp. 15–27, Chengdu, China, June 2010.

[22] Y. Zheng, "Digital signcryption or how to achieve cost (signature & encryption) $\ll$ cost(signature) + cost (encryption)," in *Proceeding of Advances in Cryptology (Crypto'97)*, LNCS 1294, pp. 165–179, Springer-Verlag, 1997.

Jayaprakash Kar has received his M.Sc and M.Phil in Mathematics from Sambalpur University, M.Tech and Ph.D in Computer Science (Cryptographic Protocols) from Utkal University, India. Currently he is working as Assistant Professor in the Department of Information Systems, Faculty of Computing and Information Technology. He is actively associated with Information Security Research Group, King Abdulaziz University, Saudi Arabia. His current research interests is on development and design of provably secure cryptographic protocols and primitives using Elliptic Curve and Pairing based Cryptography includes digital signature, Signcryption Scheme, Key management problem of broadcast encryption, Deniable authentication protocols, Proxy Blind Signature scheme. He has 01 monograph, 04 book chapters and more than 32 Journal papers and Conference articles to his credit. Dr. Kar is member of advisory and editorial board of many peer reviewed Journals. He is life member of International Association for Cryptology Research (IACR), Cryptology Research Society of India, International Association of Computer Science & Information Technology (Singapore) and International Association of Engineers (United Sates).

Elliptic Curve Based Dynamic Contributory Group Key Agreement Protocol for Secure Group Communication over Ad-hoc Networks

Vankamamidi Srinivasa Naresh[1] and Nistala V.E.S. Murthy[2]
(Corresponding author: Vankamamidi Srinivasa Naresh)
Computer Science Department, S.V.K.P. and Dr. K.S. Raju, Arts and Science College[1]
Penugonda 534320, Andhrapradesh, India.
Computer Science and System Engineering Department, Andhra University[2]
Visakhapatnam 530003, Andhra Pradesh, India
(Email: vsnaresh111@gmail.com)
(Received Feb. 6, 2013; revised and accepted Jan. 3 & Feb. 6, 2014)

Abstract

The aim of this paper is to propose an efficient and simpler Contributory Group Key Agreement protocol (CGKA) based on Elliptic Curve Diffie Hellman (ECDH). In this CGKA protocol, a member acts as a group controller (GC) and forms two-party groups with remaining group members and generates an ECDH-style shared key per each two-party group. It then combines these keys into a single group key and acts as a normal group member. This paper also addresses a Dynamic Contributory Group Key Agreement protocol (DCGKA) by extending CGKA to dynamic groups. The proposed protocol has been compared with other popular DH and ECDH based group key distribution protocols and satisfactory results were obtained.

Keywords: Dynamic group key agreement, elliptic curve Diffie-Hellman, mobile ad-hoc networks (MANETS), secure group communication (SGC)

1 Introduction

Wireless networks are growing rapidly in the last few years and also secure and reliable communication is an increasingly active research area with growing popularity in group oriented and collaborative applications. In the light of advances in Mobile ad-hoc networks the need for mechanism of secure group communications is growing day by day. Providing SGC over ad-hoc mobile networks is a very difficult task because they are mostly without much infrastructure. This problem was overcome by using elliptic curve crypto (ECC) systems. ECC emerged as the cryptographic choice for ad-hoc networks and communication devices because it can provide high security with very smaller key sizes and also at low computational expenses. Recent studies [11] indicate that the execution of ECC operations in mobile ad-hoc networks is feasible with predictable improved performance.

Secure Group Communication (SGC) refers to a scenario in which a group of participants can send and receive messages to/from other group members in a way that outsiders are unable to glean any information even when they are able to intercept the messages. The vast majority of SGC protocols use the (Discrete Logarithm Problem) based or DLP-based Diffie-Hellman as the basic key agreement protocol [16]. Any DLP-based Diffie-Hellman key agreement protocol now-a-days depends on the discrete logarithm problem for its security. The key length for secure DLP-based Diffie-Hellman has increased over recent years, which has simultaneously placed a heavier processing load on applications using DLP-based Diffie-Hellman. However, the processing load is especially critical for ad-hoc networks, which have a relatively limited bandwidth, slower CPU speed, limited battery power and high bit-error rate wireless links.

Elliptic Curve Cryptography (ECC) is a public key cryptosystems based on elliptic curves [10, 12]. The attraction of ECC is that it appears to offer equal security for a far smaller key size, thereby reducing processing overheads. However, the methods for computing general elliptic curve discrete logarithms are much less efficient than those for factoring or computing conventional discrete logarithms, indicating that more computational time is required for ECC. Thus, the overall performance of ECDLP based applications need to be evaluated.

The recent works on performance evaluation of group Diffie-Hellman protocols can be found in [2] and [6]. In [2], the authors evaluated five notable group key agreement protocols: Centralized Group Key Distribution (CKD), Burmeister Desmedt (BD), Steer et al. (STR), etc. The

Group Diffie-Hellman (GDH) key distribution protocols were first presented in [14]. There are three different versions. GDH.2 involves fewer number of rounds and messages than GDH.1 and GDH.3. The GDH protocol consists of two stages: up flow and down flow. The up flow stage collects contributions from all group members. The down flow stage broadcasts the intermediate values to all group members for calculating the shared group key. The Group Elliptic Curve Diffie-Hellman (GECDH) protocol and Tree-based Group Elliptic Curve Diffie-Hellman Protocol based on ECDLP are analyzed in [15]. Also studied Group-DH technique for Multi Party Key in [4, 13]. However, few studies have been conducted in literature on the performance of DLP and ECDLP-based group Diffie-Hellman protocols.

All the group key generating techniques can be divided into two classes. In one class, a single member of the group generates the key [5, 17] and distributes it to remaining member. However, it requires a trusted key generator for reliability. In the other class there is a contributory key agreement [1, 5, 7, 9, 14, 17, 18], in which each member of the group contributes a share to generate the group key. This class provides key secrecy. In order to assure the secrecy of communication nodes of the network, the group usually computes the group key dynamically in the sense that the group key will be updated whenever a node joins or leaves the group.

In this paper, in the second part, we propose and evaluate the performance of ECDH-based dynamic contributory group key agreement protocol over ad-hoc networks with the following secure attributes:

- Key Secrecy: The key can computed only by the members of the group.
- Forward Secrecy: As soon as a member leaves the group, it is hard to compute the new key with the previous knowledge of the old key.
- Backward Secrecy: As soon as a new member joins the group, it is hard to compute old key with the knowledge of the new key.

The rest of the paper is organized as follows: Section 2 describes the background material necessary to understand the ECDLP-based protocols. Section 3 presents the proposed ECDH-based group schemes. Section 4 discusses Security analysis. Section 5 provides comparative analysis. Finally, Section 6 concludes the paper.

2 Preliminaries

2.1 List of Some Common Abbreviations and Notations

Table 1 is the abbreviations used in this paper. Table 2 is the notations used in this paper.

Table 1: Abbreviations

Abbreviations	Full Name
ECC	Elliptic Curve Cryptography
ECDH	Elliptic Curve Diffie-Hellman
ECDLP	Elliptic Curve Discrete Log Problem
ECDP	Elliptic Curve Domain Parameters
GC	Group Controller
CGKA	Contributory Group Key Agreement
DCGKA	Dynamic Contributory Group Key Agreement
GK	Group Key
NJGK	New Join Group Key
NLGK	New Leave Group Key

2.2 Background of Elliptic Curve Group

Let E be an elliptic Curve over F_p described in terms of Weierstrass equation

$$E(x,y): y^2 = x^3 + ax + b, a, b \in F_p,$$

and with the discriminant

$$\Delta = 4a^3 + 27b^2 \neq 0.$$

The set of rational points in E over $F_p$denoted by $E(F_p)$

$$E(F_p) = \{(x,y) \in F_p^2 : E(x,y) = 0\} \cup O,$$

where O is the point at infinite. $E(F_p)$ carries a group structure under point addition with the point at infinity acting as identity element. Scalar multiplication over $E(F_p)$ can be represented as follows. The k th multiple of a point P belongs to $E(F_p)$ computed as follows:

$$[k]P = P + P + ... + (ktimes).$$

Note: For integers j and k, we have

$$[j]([k]P) = [jk]P = [k]([j]P).$$

2.2.1 Elliptic Curve Domain Parameters (ECDP)

$ECDP(p, a, b, P, n, h)$ a set of information for communicating members to identify a certain elliptic curve group used in cryptography. Here p is a large prime number, a and b are the coefficients of the Weierstrass equation, P is the base point of $E(F_p)$, having order n, and Finally the co factor $h = \#E(F_p)/n$, where $\#E(F_p)$ is the number of points on an elliptic curve group. The base point P generates a cyclic group of order n. In other words, $E(F_p) = \langle P \rangle = \{P, [2]P, ..., [n-1]P, [n]P\}$.

2.2.2 Elliptic Curve Discrete Logarithm Problem (ECDLP)

Given the ECDP as described above and $Q \in < P > = E(F_p)$, ECDLP is to find an integer l, $0 \leq l \leq n-1$ such that $Q = [l]P$.

Table 2: Notations

Symbol	Comment
p	Large Prime Number
F_p	The finite field of p elements
E	An Elliptic Curve defined by Weierstrass equation
E(F_p)	An Elliptic Curve group over the finite field F_p
P,Q	Points on the Elliptic Curve $E(F_p)$
P+Q	The Sum of two points P and Q in $E(F_p)$
[k]P	The K-th multiple of a point P, i.e [k]P=P+P+.... + (k times)
x_P,y_P	The x and y coordinates of point P respectively
P	The Base Point is a generator of a sub group of E(F_p)
n	The order of base point P typically, n is a prime of bit length ≥ 224
m	Total number of members in the group
M_i	ith group member, $1 \leq i \leq m$
M_l	The group controller
x_i	The Private Key of member M_i. This is an integer belongs to $\{1, 2, ...n-1\}$
X_i	The Public key of member M_i
$x_{K_{li}}$	ECDH shared key between GC and M_i, for $1 \leq i \leq m$, $i \neq l$

2.2.3 Cryptographically Strong Elliptic Curve Domain Parameters over F_p

The ECDLP is currently considered to be intractable if at least the following condition holds.

- The Order n of the base points P must be prime of at least 224 bits.
- To avoid the elliptic curve to be anomalous the order n must be different from p.
- The ECDLP must not be reducible to DLP in a multiplicative group F_{p^r}, for a small integer r. Thus it is required that $p^r \neq 1 \bmod n$, for all $1 \leq r \leq 10^4$.
- The class number of the principle order belongs to the endomorphism ring of E should be at least 200.

2.2.4 Elliptic Curve Diffie-Hellman

Elliptic Curve Diffie-Hellman protocol (ECDH) is one of the key exchange protocols used to establish a shared key between two members. ECDH protocol is based on the additive elliptic curve group. First A and B agree on elliptic curve domain parameters and proceed as Table 3.

The secret key K is a point on the elliptic curve. If this secret key is to be used as a session key, a single integer must be derived. There are two categories of derivation: reversible and irreversible. If the session key is also required to be decoded as a point on elliptic curve, it is reversible. Otherwise, it is irreversible. The reversible derivation will result in a session key which doubles the length of the private key. In the irreversible derivation, we can simply use the x-coordinate or simple hash function of the x-coordinate as the session key and thus the session key may have a different length with the private key.

3 Proposed Protocols

3.1 Contributory Group Key Agreement Protocol (CGKA)

We propose a contributory group key agreement protocol to generate a group key among the group members. In this technique, an arbitrary group member acts as a group controller that publicly publishes cryptographically strong elliptic curve domain parameters (p,a,b,P,n,h) and proceeds as follows.

Let $M_1, M_2, ..., M_l, ..., M_m$ be the group members and let the group controller be M_l, where $1 \leq l \leq m$.

Step 1. Initially GC, M_l forms $(m-1)$ two-party groups with each of the remaining group members M_i and produces $(m-1)$ shared keys for $(m-1)$ two-party groups, as follows:

1) The group controller M_l, selects a private key $x_l \in \{1, 2, ..., n-1\}$ and generates a public key as,

$$X_l = [x_l]P.$$

2) Each group member M_i, where $i \neq l$, also selects a private key $x_i \in \{1, 2, ..., n-1\}$ and generates a public key as

$$X_i = [x_i]P, for, 1 \leq i \leq m, i \neq l.$$

3) The GC M_l, broadcast, X_l to the remaining group members and each M_i transmits X_i to the group controller, M_l

4) After exchanging the public keys, each member generates a ECDH-style shared key with GC as

$$K_{li} = [x_i]X_l = [x_i]([x_l]P) = [x_i x_l]P = (x_{K_{li}}, y_{K_{li}}),$$

for $1 \leq i \leq m, i \neq l$. Where $x_{K_{li}}, y_{K_{li}} \in F_p$ are x and y coordinates of K_{li}, respectively.
Similarly GC, M_l generates the same shared keys as

$$K_{li} = [x_l]X_i = [x_l]([x_i]P) = [x_l x_i]P = (x_{K_{li}}, y_{K_{li}}),$$

for $1 \leq i \leq m, i \neq l$. Where $x_{K_{li}}$, $y_{K_{li}} \in F_p$ are x and y coordinates of K_{li}, respectively.
Hence take $x_{K_{li}}$ be the $(m-1)$ shared keys between the GC, M_l and M_i, where $1 \leq i \leq m$, $i \neq l$, respectively.

Table 3: ECDH

Party-A	Communication	Party-B
Choose a random number $x \in \{1, 2, ...n-1\}$		Choose a random number $y \in \{1, 2, ...n-1\}$
Compute $[x]\,P$		Compute $[y]\,P$
Retrieve $[y]\,P$	$\overset{[x]P}{\rightarrow}$ $\overset{[y]P}{\leftarrow}$	Retrieve $[x]\,P$
Compute [x][y]P=[xy]P		Compute [y][x]P=[yx]P=[xy]P

Step 2. Now the group controller computes the $(m-1)$ public keys L_i as follows and send to M_i respectively.

$$L_i = [\prod_{j=1, j\neq i}^{m} x_{K_{li}}]P, for 1 \leq i \leq m, i \neq l, and\ j \neq l.$$

After retrieving L_i each member M_i of the group generates group key K as follows:

$$\begin{aligned} K &= [x_{K_{li}}]L_i \\ &= [x_{K_{li}}][\prod_{j=1, j\neq i}^{m} x_{K_{lj}}]P \\ &= [\prod_{i=1}^{m} x_{K_{li}}]P \\ &= (x_K, y_K)\,. \end{aligned}$$

Since the GC knows all the shared keys, it also generates the group key:

$$K = [\prod_{i=1}^{m} x_{K_{li}}]P = (x_K, y_K)\,.$$

Hence take x_K as group key among the group members.

3.2 Dynamic Contributory Group Key Agreement Protocol (DCGKA)

CGKA addresses group key agreement for static groups. However, it is often times necessary to either to add a new member (or) delete an existing group member of the initial group creation. Naturally, it is desirable to do so without executing entire protocol a new. To address this issue we extend CGKA to DCGKA by proposing join protocol and leave protocol.

3.2.1 Join Protocol

The main security requirement of the member addition is the secrecy of the previous group key with respect to outsider and new group members.

1) When a new member M_{m+1} wants to join the group, intimates the group controller and generates a ECDH-style key $x_{K_{lm+1}}$ with GC.

2) GC generates a random number $R^{'}_{m+1}$ and broadcasts $\left[x_{K_{lm+1}}.R_{m+1'}\right]P$ to all the previous members of the group, M_i on receiving they compute the new key:

$$NJGK = (x_K)x_{K_{lm+1}}R^{'}_{m+1}P = (\prod_{i=1}^{m+1} x_{K_{li}}.R^{'}_{m+1})P,$$

where x_K is the previous group key.

3) GC transmits $[(x_K)R^{'}_{m+1}]P$ to M_{m+1} and then M_{m+1} computes the new key as follows:

$$NJGK = (x_K)x_{K_{lm+1}}R^{'}_{m+1}P = (\prod_{i=1}^{m+1} x_{K_{li}}R^{'}_{m+1})P,$$

where x_K is the previous group key.

3.2.2 Leave Protocol

The main security requirement of member leaving is the secrecy of the subsequent (future) group key with respect to both outsiders and former group members.

1) When M_j wants to leave the group, intimates the GC and then GC, M_l generates a random number R'_j.

2) M_l sends $\left[R'_j x^{-1}_{K_{lj}}\right]P$ by encrypting with $x_{K_{li}}$ to the corresponding group member M_i, $i \neq j$, (i.e) except leaving member.

$$M_l \overset{E_{K_{li}}\left[R'_j x^{-1}_{K_{lj}}\right]P}{\longrightarrow} M_i, for 1 \leq i \leq m, i \neq j.$$

After receiving each member computes the new key as follows:

$$NLGK = (x_K)R'_j x^{-1}_{K_{lj}}P = \left[\prod_{i=1, i\neq j}^{m} x_{K_{li}}R'_j\right]P,$$

where x_K is the previous group key.

3) Also M_l computes the new key as follows.

$$NLGK = (x_K)R'_j x^{-1}_{K_{lj}}P = \left[\prod_{i=1, i\neq j}^{m} x_{K_{li}}R'_j\right]P,$$

where x_K is the previous group key.

4 Security Analysis

We Prove that our protocols meet the desirable attributes under the assumption that the Elliptic Curve Discrete Logarithm Problem is secure.

Theorem 1. *The group key derived using CGKA PROTOCOL is indistinguishable in polynomial time from random numbers.*

Proof. If the m-group members execute CGKA protocol then they clearly share a group key K. During the computation of group key K, in Step 1 we have generated $(m-1)$, two-party ECDH style keys.

An adverser tries to extract the private keys x_i from unknown public keys $X_i = [x_i]P$, but this is an Elliptic Curve Discrete Problem and hence two-party ECDH-style keys generated in Step 1 are indistinguishable in polynomial time.

In Step 2 of CGKA, GC generates $(m-1)$ public keys L_i and sends to M_i, respectively. That is

$$M_l \overset{E_{K_{li}}\left[R'_m x^{-1}_{K_{lj}}\right]P}{\longrightarrow} M_i, for 1 \le i \le m, i \ne l, and\ j \ne l.$$

An adversary tries to extract all the products

$$\prod_{j=1, j\ne i}^{m} x_{K_{li}}, for 1 \le i \le m.$$

From publicly known,

$$L_i = [\prod_{j=1, j\ne i}^{m} x_{K_{li}}]P.$$

But this is again an elliptic curve discrete logarithm problem. Therefore all the products

$$\prod_{j=1, j\ne i}^{m} x_{K_{li}}, for 1 \le i \le m,$$

are indistinguishable from random numbers in polynomial time and hence it is difficult to find $x_{K_{li}}$. □

Theorem 2. *DCGKA with join protocol satisfies the properties of backward security.*

Proof. The GC generates a random number R'_{m+1} as soon as a new member joins the network group and broadcasts GC generates a random number R'_{m+1} and broadcasts $\left[x_{K_{lm+1}}.R_{m+1'}\right]P$ to all the previous members of the group, M_i on receiving they compute the new key

$$NLGK = (x_K)R'_m x^{-1}_{K_{lj}} P = \left[\prod_{i=1, i\ne j}^{m} x_{K_{li}} R'_m\right] P,$$

where x_K is the previous group key.

On basis of ECDLP, it is hard for out-sider and new group members to compute previous group key. □

Theorem 3. *DCGKA with leave protocol satisfies the properties of the forward security.*

Proof.

1) When M_j wants to leave the group, intimates the GC and then GC, M_l generates a random number R'_j.

2) M_l sends $\left[R'_j x^{-1}_{K_{lj}}\right]P$ by encrypting with $x_{K_{li}}$ to the corresponding group member M_i, $i \ne j$, i.e., except leaving member.

$$M_l \overset{E_{K_{li}}\left[R'_j x^{-1}_{K_{lj}}\right]P}{\Longrightarrow} M_i, \quad \text{for} \quad 1 \le i \le m,\ i \ne j.$$

$$NLGK = (x_K)R'_j x^{-1}_{K_{lj}} P = \left[\prod_{i=1, i\ne j}^{m} x_{K_{li}} R'_j\right] P,$$

where x_K is the previous group key.

3) Also M_l computes

$$NLGK = (x_K)R'_j x^{-1}_{K_{lj}} P = \left[\prod_{i=1, i\ne j}^{m} x_{K_{li}} R'_j\right] P,$$

where x_K is the previous group key.

As $\left[R'_j x^{-1}_{K_{lj}}\right]P$ is in encrypted form it is secured from outsiders and also GC keeps it secure from leaving member, we have the main security requirement of member leaving are satisfied with respective both outsiders and former group members.

□

5 Comparative Analysis

In this section, the proposed ECDLP-based DCGKA protocol has been firstly compared with DLP based group key distribution protocols, and then with ECDLP based protocols in terms of number of rounds, messages, operations and so on.

Table 4 shows the comparable key sizes (Table 5) of the same security level for an ECDLP-based group scheme and DLP-based scheme. It shows that ECDLP-based schemes can use a much smaller key size than DLP-based group schemes.

The key length for secure DLP-based Diffie-Hellman has increased over recent years, which has also placed a heavier processing load on applications using DLP-based Diffie-Hellman. However, the processing load is especially critical for ad-hoc networks, which have a relatively limited bandwidth, slower CPU speed, limited battery power and high bit-error rate wireless links and ECDLP-based group schemes are having lower communication overheads and less computation load than DLP-based group scheme.

As per the advantages and adaptability for ad-hoc networks of ECDLP over DLP In this paper, we proposed ECDLP-based group key distribution protocol DCGK at

Table 4: Comparative analysis of popular group key agreement protocols

DLP-Protocols		Rounds	Messages	Unicast	Broadcast	Seq exponentions	Seq scalar multiplications
CEGK [3]	Initialize	h	$2m-2$	m	$m-2$	$2h-2$	0
	Join	1	2	1	1	1	0
	Leave	1	1	0	1	$h-1$	0
EGK [1]	Initialize	h	$2m-2$	0	$2m-2$	$2h-2$	0
	Join	1	2	0	2	1	0
	Leave	h	$2(m-1)$	0	$2(m-1)$	$2h$	0
TGDH [9]	Initialize	h	$2m-2$	0	$2m-2$	$2h-2$	0
	Join	2	3	0	3	$3h-3$	0
	Leave	1	1	0	1	$3h-3$	0
STR [8]	Initialize	$m-1$	$2m-2$	0	$2m-2$	$2(m-1)$	0
	Join	2	3	0	3	4	0
	Leave	1	1	0	1	$m-1$	0
GDH.3 [14]	Initialize	$m+1$	$2m-1$	$2m-3$	2	$5m-6$	0
	Join	4	$m+3$	0	$m+3$	$m+3$	0
	Leave	1	1	0	1	m-1	0
ECDLP-based Protocol		Rounds	Messages	Unicast	Broadcast	Seq exponentions	Seq scalar multiplications
GECDH [15]	Initialize	m	m	$m-2$	2	0	$5m-6$
	Join	m	n	0	m	0	$m+3$
	Leave	$m-1$	$m-1$	0	$m-1$	0	$m-1$
TGECDH [15]	Initialize	h	$2m-2$	0	$2m-2$	0	$2h-2$
	Join	2	3	0	3	0	$3h-3$
	Leave	1	1	0	1	0	$3h-3$
DCGKA [our protocol]	Initialize	$m+1$	$2m-1$	$2m-2$	1	0	$2m$
	Join	1	2	1	1	0	6
	Leave	1	1	0	1	0	3

Table 5: Key sizes

ECDLP-based scheme (size of n in bits)	DLP-based scheme (modular size in bits)
112	512
160	1024
224	2048
256	3072
384	7680
512	15360

the same security level as the DLP-based Diffie-Hellman schemes.

Our protocol uses only two steps which involve very simple operations. Being ECDLP-based protocol additions and scalar multiplications are used instead of multiplications and exponentiations (as in DLP-based protocols) respectively and also it uses smaller key sizes. Hence our protocol works with lesser computational expense. However, the group controller needs to execute comparatively more key exchange operations than the other group members, but these operations are very simple with lesser computational expense. The overall delay of key generation depends on the performance of group controller. Since most of today's machines have high computation power, the proposed technique may not be a problem for practical applications.

In view of above comparative analysis in Table 5, our protocol [DCGK] is optimal in terms of comparatively less communication and computation cost and also it provides same security level with smaller key sizes. Our protocol is relatively best protocol for secure group key distribution over ad-hoc networks among the DLP and ECDLP based schemes discussed in this paper.

Computational Complexities.

Initialization of group key. The number of sequential scalar multiplications for initialization of group key in our protocol [CGKA] is lesser than GECDH. Although our protocol uses much number of sequential scalar multiplications than TGECDH. Our protocol is much simpler comprising only two steps with very simple operations (See Figure 1).

Join protocol. The number of sequential scalar multiplications for new member join group key in our protocol is fewer than GECDH and TGECDH protocol, In fact only six scalar multiplications independent of group size (See Figure 2).

Leave protocol. The number of sequential scalar multiplications for new member leave group key in our protocol is fewer than GECDH and TGECDH protocol, In fact only three scalar multiplications independent of group size (See Figure 3).

Communication Complexities.

Number of messages. DCGKA protocol is the best in terms of communication for updating the group key whenever a new member joins or existing member leaves. For initialization of group key our protocol uses $2m - 1$ messages which is nearly same as TGECDH and higher than GECDH.

Storage cost. As per the memory to store the keys at member nodes, the ECC makes the process as easy as possible, since the key sizes are small with ECC. In tree based approaches each node has to maintain the keys of its leaf nodes and so on. So DCGKA consumes very low memory storage cost than tree based approaches.

In view of the above observations, DCGKA is optimal in terms of low communication and computation costs and also it provides same security level with smaller key sizes. Thus it is relatively a better protocol for secure group key distribution over ad-hoc networks among the DLP and ECDLP based schemes discussed in this paper.

6 Conclusion and Future Work

In this paper, we proposed ECDLP based Dynamic contributory Group key agreement (DCGKA) protocol for secure group communication over ad-hoc networks. The theoretical analysis shows that DCGKA is certainly a better protocol in overall performance among the DLP and ECDLP based schemes discussed in this paper. Also it provides secure key attributes such as key secrecy, forward secrecy and backward secrecy.

The performance of DCGKA over ad-hoc networks can be Summarized as follows:

- It has relatively low communication overheads and lesser computational expense.
- It consumes very low memory storage cost than the tree based approaches.
- Most importantly, it is quite simple to implement in the sense that it uses only two steps which involve very simple operations.
- It uses dynamic updating of key without a re-run of the protocol anew as soon as a member joins or leaves the existing group.
- It uses smaller keys.

Therefore it may be apt for secure group key agreements over mobile ad-hoc networks.

In continuation of this paper, there remain some items for future work. Our protocol do not provide authentication of the participants. It should be possible to argument them to provide authentication using public Key Infrastructure (PKI), with out increasing computational and communication load. Also to address most of the active attacks, such as key impersonation and forgery attack etc..

References

[1] J. Alves-Foss, "An efficient secure group key exchange algorithm for large and dynamic groups," in *Proceedings of the 23rd National Information Systems Security Conference*, pp. 254–266, Oct. 2000.

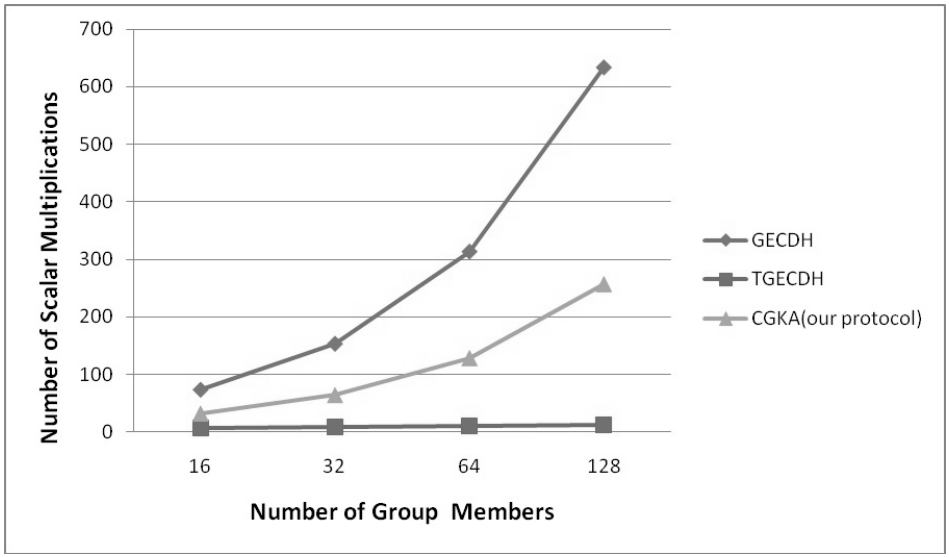


Figure 1: Comparative analysis of ECDLP based protocols for initialization of group key

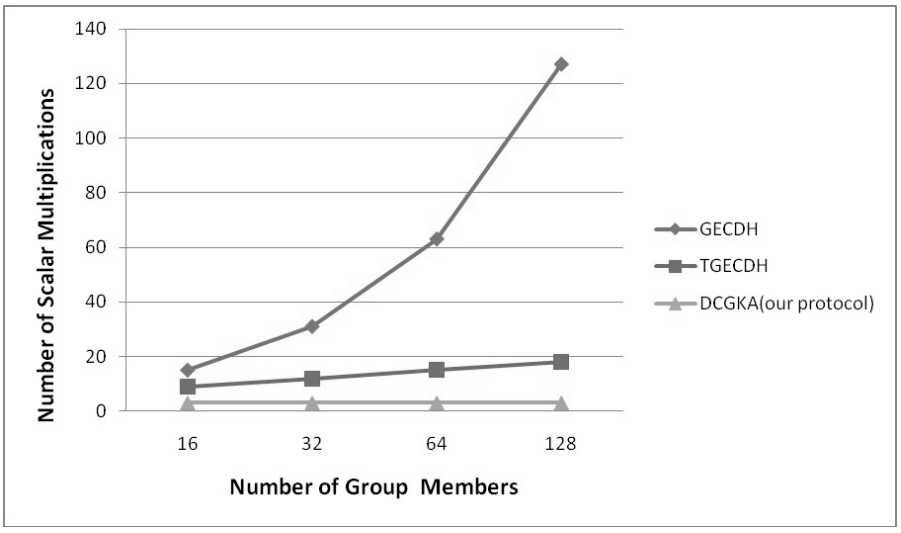


Figure 2: Comparative analysis of ECDLP based protocols for member join group key

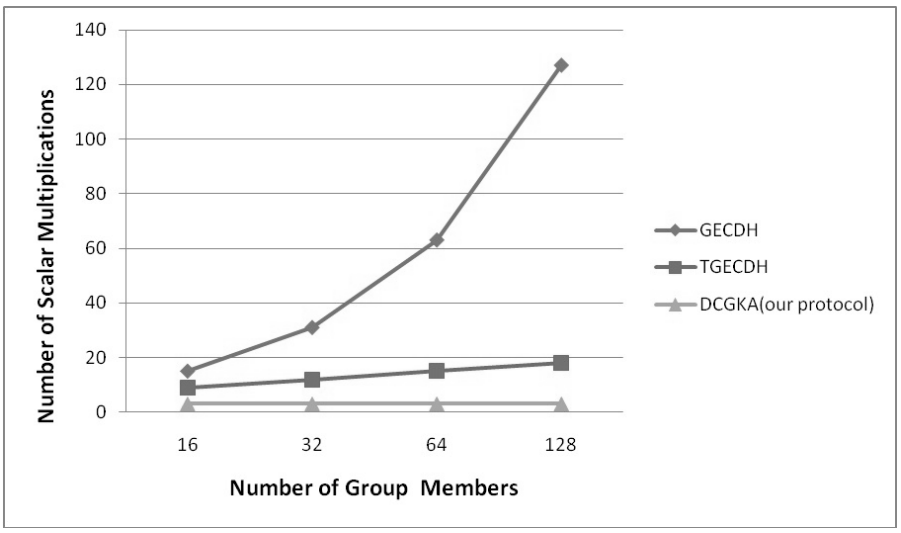


Figure 3: Comparative analysis of ECDLP based protocols for member leave group key

[2] Y. Amir, Y. Kim, C. Nita-Rotaru, and G. Tsudik, "On the performance of group key-agreement protocols," *ACM Transactions on Information and System Security*, vol. 7, no. 3, pp. 457–488, 2004.

[3] K. Becker and U. Wille, "Communication complexity of group key distribution," in *5th Conference on Computer and Communication Security*, pp. 1–6, 1998.

[4] G. P. Biswas, "Diffie hellman technique extended to multiple two party keys and one multi party key," *IET Information Security*, vol. 2, no. 1, pp. 12–18, 2008.

[5] M. Burmester and Y. Desmedt, "A secure and efficient conference key distribution system," in *Eurocrypt'94*, pp. 275–286, May 1994.

[6] K. S. Hagzan and H. P. Bischof, "The performance of group diffie-hellman paradigms," in *2004 International Conference on Wireless Networks*, Las Vegas, Nevada, USA, 2004.

[7] I. Ingemarsson, D. Tang, and C. Wong, "A conference key distribution system," *IEEE Transactions on Information Theory*, vol. 28, no. 5, pp. 714–720, 1982.

[8] Y. Kim, A. Perrig, and G. Tsudik, "Group key agreement efficient in communication," *IEEE Transactions on Computer*, vol. 53, no. 7, pp. 905–921, 2004.

[9] Y. Kim, A. Perrig, and G. Tsudik, "Tree-based group key agreement," *ACM Transactions on Information System Security*, vol. 7, no. 1, pp. 60–96, 2004.

[10] N. Koblitz, "Elliptic curve cryptosystems," *Mathematics of Computation*, vol. 48, pp. 203–209, 1987.

[11] D. Malan, M. Welsh, and M. D. Smith, "A public-key infrastructure for key distribution in tiny os based on elliptic curve cryptography," in *IEEE International Conference on Sensor and Ad Hoc Communications and Networks*, pp. 71–80, Oct. 2004.

[12] V. S. Miller, "Use of elliptic curves in cryptography," in *Crypto'85*, LNCS 218, pp. 417–426, Springer-Verlag, 1986.

[13] V. S. Naresh, N. V.E.S. Murthy, "Diffie-Hellman technique extended to efficiently and simpler group key distribution protocol," *International Journal of Computer Applications*, vol. 4, no. 11, pp. 1–5, Aug. 2010.

[14] M. Steiner, G. Tsudik, and M. Waidner, "Diffie-Hellman key distribution extended to group communication," in *Proceedings of the 3rd ACM Conference on Computer and Communications Security*, pp. 31–37, New York, NY, USA, 1996.

[15] Y. Wang, B. Ramamurthy, and X. Zou, "The performance of elliptic curve based group Diffie-Hellman protocols for secure group communication over ad-hoc networks," in *IEEE International Conference on Communications*, pp. 2243–2248, 2006.

[16] W. Diffie and M. E. Hellman, "New directions in cryptography," *IEEE Transactions on Information Theory*, vol. 22, pp. 644–654, Nov. 1976.

[17] C. Wong, M. Gouda, and S. Lam, "Secure group communication using key graphs," in *Proceeedings of the ACM SIGCOMM'98*, pp. 68–99, 1998.

[18] S. Zheng, D. Manz, and J. Alves-foss, "A communication-computation efficient group key algorithm for large and dynamic groups," *Computer Networks*, vol. 51, no. 1, pp. 69–93, 2007.

Vankamamidi Srinivasa Naresh is currently working as a Director, for the Post Graduate Department of Computer Science Courses in S.V.K.P. and Dr. K.S.R. Arts and Science College. He obtained an M.Sc. in Mathematics from Andhra University, an M.Phil. in Mathematics from Madurai Kamaraj University and an M.Tech in Computer Science and Engineering from J.N.T.University-Kakinada. He is also a recipient of U.GC.-C.S.I.R.JUNIOR RESEARCH FELLOSHIP and cleared NET for Lectureship in Mathematical sciences and also cleared UGC NET in Computer Science and Applications. He published papers in reputed journals in the area of cryptography. Presently pursuing Doctorate from JNTUK.

Nistala V.E.S. Murthy is currently working as a Professor in the department of Computer Science and Systems Engineering of Andhra University, Visakhapatnam. He developed f-Set Theory -wherein f-maps exists between Fuzzy Sets with truth values in *different* complete lattices, generalizing L-fuzzy set theory of Goguen which generalized the [0,1]-fuzzy set theory of Zadeh, the Father of Fuzzy Set Theories. He also published papers on, Representation of Various Fuzzy Mathematical (Sub) Structures in terms of their appropriate crisp cousins, Various Fuzzy Set Theories and Their Applications in Mathematics (Algebra and Topology) and Computer Science (Data Warehousing, Data Mining), Cryptography and Natural Language Modeling. He can be visited at URL: http://andhrauniversity.academia.edu/NistalaVESMurthy.

An Improved Certificateless Signcryption in the Standard Model

Lin Cheng and Qiaoyan Wen
(Corresponding author: Lin Cheng)
State Key Laboratory of Networking and Switch Technology, Beijing University of Posts and Telecommunications
Beijing 100876, China
(Email: stonewoods302@163.com)
(Received Mar. 21, 2013; revised and accepted Jan. 7 & Feb. 6, 2014)

Abstract

Signcryption is a cryptographic primitive which can offer simultaneously security requirements of confidentiality and authentication, and is more efficient than the traditional sign-then-encrypt way. Recently, Liu et al. proposed the first certificateless signcryption scheme in the standard model. However, their scheme is proved to have some security weaknesses. In this paper, we propose a corrected version of Liu et al.'s scheme and prove our scheme is indistinguishable against adaptive chosen ciphertext attacks and is existentially unforgeable against chosen message attacks in the standard model. Performance analysis shows the new scheme has smaller public parameter size than the previous certificateless signcryption schemes without using the random oracles.

Keywords: Cryptography, provable security, signcryption

1 Introduction

In traditional public key cryptography, it needs a certificate issued by the certification authority (CA) to achieve authentication of the user's public key. However, the cost of certificate management is very high. To conquer this problem in traditional public key cryptography, Shamir [18] introduced the Identity-based cryptography. In identity-based cryptography, the user's public key is derived directly from its name, email-address or other identity information, but it requires a trusted third party called Key Generation Center (KGC) generate the user's private key. Unfortunately, we are confronted with the key escrow problem in identity-based cryptography, that is, KGC knows user's private key so that it can decrypt any ciphertext and sign any message on behalf of the user. At 2003, Al-Riyami and Paterson [1] introduced certificateless public key cryptography, which resolves the inherent key escrow problem in identity-based cryptography, without requiring certificates as used in traditional public key cryptography. In certificateless public key cryptography, the user's public key is independently generated by the user, and the user's private key is a combination partial private key computed by KGC and some user-chosen secret value, in such a way that the key escrow problem can be eliminated without requiring certificates.

Confidentiality and authenticity are two fundamentally different security requirements and realized through encryption and signature schemes respectively. A natural solution to offering simultaneously both requirements is using sign-then-encrypt approach. Signcryption, first introduced by Zheng [26], is a cryptographic primitive that combines the functionality of public encryption with digital signature and is more efficient than the traditional signature-then-encrypt approach. Since then, many signcryption schemes [7, 11, 19, 23] were proposed. The first certificateless signcryption (CLSC) scheme was introduced by Barbosa and Farshim [3]. Later, some efficient CLSC schemes were proposed [12, 22, 25]. However, all of these CLSC schemes are provably secure in the random oracle model [5], which can only be considered as a heuristic argument [6]. It has been shown in [4] that the security of the scheme may not preserve when the random oracle is instantiated with a particular hash function such as SHA-1. Based on Waters' identity-based encryption scheme [20] and its variants [8, 9, 13, 24], Liu et al. [14] introduced the first CLSC scheme in the standard model. Unfortunately, in [15, 17], Liu et al.'s CLSC scheme [14] is proved to be not secure against a type I adversary who can compromise users secret value or replace user public key, but neither compromise master secret key nor get access to partial private key. Weng et al. [21] showed that Liu et al.'s CLSC scheme [14] is also not secure against the malicious-but-passive KGC attack [2], where a malicious KGC can control the generation of master public/secret key pair, but cannot compromise user's secret value nor replace any public key. Though a rescued scheme has been proposed by Jin et al. [10], it still can not resist the attacks in [21]. This is because Jin et al.'s scheme has the same signcryption algorithm as Liu et al.'s CLSC scheme [14]. To the best of our knowledge,

a secure CLSC scheme without random oracles is still an open problem. In this paper, we propose an improved Liu et al.'s CLSC scheme which can resist all the attacks in [15, 17, 21]. In addition, by using Naccache's methods [16], our new scheme has a smaller system parameters size than schemes [10, 14].

The rest paper is organized as follows. We provide some preliminaries in Section 2. Then, we recall the definition of certificateless signcryption scheme and its security model in Section 3. We propose a corrected version of Liu et al.'s scheme in Section 4. Its formal security proof is presented in Section 5. Finally a concluding remark is given in Section 6.

2 Preliminaries

In this Section, we recall the bilinear pairing and complexity assumptions [14].

2.1 Bilinear Pairing

Let G and G_T be two (multiplicative) cyclic groups with prime order p. A bilinear pairing is a map $e : G \times G \rightarrow G_T$ with the following properties:

1) Bilinear: $\forall g_1, g_2 \in G, \forall a, b \in Z_p^*$, we have $e\left(g_1^a, g_2^b\right) = e\left(g_1, g_2\right)^{ab}$;

2) Non-degeneracy: There exist $g_1, g_2 \in G$ such that $e\left(g_1, g_2\right) \neq 1_{G_T}$, where 1_{G_T} denotes the identity element of group G_T;

3) Computability: There exists an efficient algorithm to compute $e\left(g_1, g_2\right)$ for $\forall g_1, g_2 \in G$.

2.2 Decisional Bilinear Diffie-Hellman Assumption

Given a group G of prime order p with generator g, a bilinear pairing e: $G \times G \longrightarrow G_T$ and elements $g^a, g^b, g^c \in G$, $e(g,g)^z \in G_T$ where a, b, c, z are selected randomly from Z_p^*. Let $\beta \in 0, 1$ be a random binary coin. If $\beta = 1$, it outputs the tuple $(g, A = g^a, B = g^b, C = g^c, Z = e(g,g)^{abc})$. If $\beta = 0$, it outputs the tuple $(g, A = g^a, B = g^b, C = g^c, Z = e(g,g)^z)$. The decisional Bilinear Diffie-Hellman (**DBDH**) assumption is that no t-time algorithm $\mathcal{B}$ has at least ϵ advantage in determining the value of β, where the advantage is defined as $| Pr[1 \longleftarrow \mathcal{B}(g, g^a, g^b, g^c, e(g,g)^{abc})] - Pr[1 \longleftarrow \mathcal{B}(g, g^a, g^b, g^c, e(g,g)^z)]$.

2.3 Computational Diffie-Hellman Assumption

Given (g, g^a, g^b) where a, b are selected randomly from Z_p^*. The computational Diffie-Hellman (CDH) assumption is that no t-time algorithm $\mathcal{B}$ has at least ϵ advantage in computing g^{ab}, where the advantage is defined as $Pr[\mathcal{B}(g, g^a, g^b) = g^{ab}]$.

3 Formal Model of Certificateless Signcryption

3.1 Definition of Certificateless Signcryption

A CLSC scheme consists of the following six algorithms:

Setup(k). On input a security parameter k, this setup algorithm generates a master public/secret key pair (mpk, msk).

PartialPrivateKeyGen. On input msk, mpk, and a user identity ID, it generates the user's partial Private key psk_{ID}.

UserKeyGen. On input mpk and psk_{ID}, it generates the public/private key pair (pk_{ID}, sk_{ID}).

User-Key-Generate. On input $params$ and a user identity ID, it returns a randomly chosen secret value x_{ID} and a corresponding public key pk_{ID} for the user.

Private-Key-Extract. On input $params$, a user's partial private key psk_{ID} and secret value x_{ID}, it returns the user's full private key sk_{ID}.

Signcrypt. On input $params$, a message M, a sender's private key sk_S, identity ID_S and public key pk_S, and a receivers identity ID_R and public key pk_R, it returns a ciphertext δ or an error symbol $\perp$.

Unsigncrypt. On input a ciphertext δ, the receiver's private key sk_R, and the sender's public key pk_S, it outputs a plaintext M or an error symbol $\perp$.

For consistency, these algorithms must satisfy that if $\delta = Signcrypt(params, M, sk_S, ID_R, pk_R)$, then M should be a part of $Unsigncrypt(params, \delta, sk_R, pk_S)$.

3.2 Security Models

An adversary $\mathcal{A}$ is allowed to access to the following oracles.

Public-Key-Broadcast-Oracle. On input of any identity ID, challenger returns corresponding public key. If such a key does not yet exist, challenger computes the corresponding public key pk_{ID} and returns pk_{ID} to $\mathcal{A}$.

Partial-Private-Key-Oracle. On input of any identity ID, challenger computes the corresponding partial private key psk_{ID} for this identity and returns psk_{ID} to $\mathcal{A}$.

Public-Key-Replacement-Oracle. On input of an identity ID and a new valid public key value pk'_{ID}, challenger replaces the current public key with pk'_{ID}.

Private-Key-Extract-Oracle. On input of an identity ID whose public key was not replaced, challenger computes the private key sk_{ID} for this identity and returns sk_{ID} to $\mathcal{A}$.

Signcrypt. On input of a sender's identity ID_S, a receiver's identity ID_R and a message M, challenger responds by running the **Signcrypt** algorithm on the message M, the sender's private key sk_S and the receiver's public key pk_R. It is possible for the challenger not to be aware of the sender's secret value when the associated public key has been replaced. In this case, we require $\mathcal{A}$ to provide the sender's secret key.

Unsigncrypt. On input of a ciphertext δ, a sender's identity ID_S and a receiver's identity ID_R, challenger returns the result of running the **Unsigncrypt** algorithm on the ciphertext δ, the receiver's private key sk_R and the sender's public key pk_S. It is possible for the challenger not aware of the receiver's secret value when the associated public key has been replaced. In this case, $\mathcal{A}$ is required to provide the receiver's secret key.

Definition 1. *A CLSC scheme is said to have the indistinguishability against adaptive chosen ciphertext attacks property (IND-CLSC-CCA), if no polynomially bounded adversaries $\mathcal{A}$($\mathcal{A}_I$ and $\mathcal{A}_{II}$) have non-negligible advantage of winning the following game.*

Initialization. *If the adversary is $\mathcal{A}_I$, challenger runs algorithm **Setup** to generate the master key msk and the master public key mpk, and then gives mpk to $\mathcal{A}_I$ and keeps msk secret. If the adversary is $\mathcal{A}_{II}$, adversary $\mathcal{A}_{II}$ runs algorithm **Setup** to generate the master secret key msk and the master public key mpk. $\mathcal{A}_{II}$ then gives mpk and msk to challenger.*

Phase 1. *In this phase, $\mathcal{A}$ adaptively performs a polynomially bounded number of oracle queries. Actually, $\mathcal{A}_{II}$ does not need to issue partial private key queries, since it can compute them from the master key by itself.*

Challenge. *At the end of Phase 1, the adversary outputs two distinct identities ID_{S^*}, ID_{R^*} and two equal length messages $\{M_0, M_1\}$. The challenger chooses a bit γ randomly and signcrypts M_γ under the ID_{S^*}'s private key and the ID_{R^*}'s public key to produce δ^*. The challenger returns δ^* to the adversary.*

Phase 2. *The adversary continues to probe the challenger with the same type of queries that it made in Phase 1. To capture insider security, the adversaries are assumed to have access to the private key of the sender ID_{S^*} of a signcrypted message.*

Response. *The adversary returns a bit γ'. We say that the adversary wins the game if $\gamma' = \gamma$and the adversary fulfills the following conditions:*

1) *$\mathcal{A}_I$, $\mathcal{A}_{II}$ cannot extract the private key for ID_{R^*} at any point.*
2) *$\mathcal{A}_I$ cannot extract the private key for any identity if the corresponding public key has already been replaced.*
3) *$\mathcal{A}_I$ cannot extract the partial private key of ID_{R^*} if $\mathcal{A}_I$ replaced the public key pk_{R^*} before the challenge phase.*
4) *In Phase 2, $\mathcal{A}_I$ cannot make an unsigncryption query on the challenge ciphertext δ^* under ID_{S^*} and ID_{R^*} unless the public key pk_{S^*} of the sender or that of the receiver pk_{R^*} used to signcrypt M_γ has been replaced after the challenge was issued.*
5) *In Phase 2, $\mathcal{A}_{II}$ cannot make an unsigncryption query for the challenge ciphertext δ^* under ID_{S^*} and ID_{R^*} and public key pk_{R^*} that were used to signcrypt M_γ.*

$\mathcal{A}$'s advantage is defined as $Adv_{\mathcal{A}}^{IND-CL-CCA} = |2Pr[\gamma' = \gamma] - 1|$.

Definition 2. *A CLSC scheme is said to be secure against an existential forgery for adaptive chosen message attacks (EUF-CLSC-CMA), if no polynomially bounded adversaries ($\mathcal{A}_I$ and $\mathcal{A}_{II}$) have non-negligible advantage of winning the following game.*

Initialization. *It is the same as above.*

Queries. *$\mathcal{A}$ may adaptively issue a polynomially bounded number of queries to $\mathcal{B}$ as above Phase 1. To deal with the insider security, assuming the adversary can gain access to the private key of the receiver of a signcrypted message.*

Output. *Eventually, $\mathcal{A}$ outputs a new triple $(\delta^*, ID_{S^*}, ID_{R^*})$, which is not produced by the signcryption query. The adversary wins if the result of unsigncrypt $(\delta^*, ID_{S^*}, pk_{S^*}, sk_{R^*})$ is not the symbol $\perp$ and the queries are subject to the following constraints:*

1) *$\mathcal{A}_I$, $\mathcal{A}_{II}$ cannot extract the private key for ID_{S^*} at any point.*
2) *$\mathcal{A}_I$ cannot extract the private key for any identity if the corresponding public key has already been replaced.*
3) *$\mathcal{A}_I$ cannot extract the partial private key of ID_{S^*}.*

We define $\mathcal{A}$'s success probability in the game above to be $Succ_{\mathcal{A}}^{EUF-CLSC-CMA} = Pr[\mathcal{A}wins]$.

4 Improved Scheme

Let G and G_T be groups of prime order p and g be a generator of G, and let e: $G \times G \rightarrow G_T$ be a bilinear pairing. ϕ: $\Re \rightarrow G_T$ is a bijection while ϕ^{-1} is its

inverse mapping, $\Re$ is a subset of $\{0,1\}^{m+n}$ with p elements. Identity ID will be represented as n dimensional vectors $d_{ID} = (d_{ID,1}, \cdots, d_{ID,n})$ where each $d_{ID,i}$ is an ℓ-bit integer, and $n' = n \cdot \ell$ is the length of an identity in binary string representation. H': $\{0,1\}^* \rightarrow \{0,1\}^{n'}$, H: $\{0,1\}^* \rightarrow \{0,1\}^m$ are two collision-resistant hash functions.

Setup. The KGC selects randomly values α, u', v' in Z_p^* and two random vectors $\mathbf{U} = (u_i)_n$, $\mathbf{V} = (v_j)_m$ and then computes $g_1 = g^\alpha$, and selects randomly $g_2 \in G$. The master public key mpk and the master secret key msk are respective $(g_1, g_2, u', v', \mathbf{U}, \mathbf{V})$ and g_2^α.

PartialPrivateKeyGen. The KGC picks a random value $r \in Z_p^*$ and computes partial private key $psk_{ID} = (psk_{ID,1}, psk_{ID,2}) = \left(g_2^\alpha (u' \prod_{i=1}^{i=n} u_i^{d_{ID,i}})^r, g^r\right)$. The sender and the receiver's partial private keys are

$$psk_S = (psk_{S,1}, psk_{S,2}) = \left(g_2^\alpha \cdot (u' \prod_{i=1}^{i=n} u_i^{d_{S,i}})^r, g^r\right)$$

$$psk_R = (psk_{R,1}, psk_{R,2}) = \left(g_2^\alpha \cdot (u' \prod_{i=1}^{i=n} u_i^{d_{R,i}})^r, g^r\right)$$

UserkeyGen. Pick a secret value $x_{ID} \in Z_p^*$, and generate public key $pk_{ID} = \{pk_{ID,1}, pk_{ID,2}, pk_{ID,3}\} = \{g^{x_{ID}}, g_1^{x_{ID}}, g_2^{x_{ID}}\}$. Then it randomly picks r' from Z_p^* and computes private key sk_{ID} as

$$\begin{aligned} & (sk_{ID,1}, sk_{ID,2}) \\ = & \left(psk_1^{x_{ID}^2} \cdot (u' \prod_{i=1}^{i=n} u_i^{d_{ID,i}})^{r'}, psk_2^{x_{ID}^2} \cdot g^{r'}\right) \\ = & \left(g_2^{\alpha x_{ID}^2} \cdot (u' \prod_{i=1}^{i=n} u_i^{d_{ID,i}})^t, g^t\right) \end{aligned}$$

where $t = r x_{ID}^2 + r'$.

Signcrypt. To send a message $M \in \{0,1\}^m$ to an identity ID_R with public key pk_R, first check whether the public key pk_R is correctly formed, by checking $e(pk_{R,1}, g_1) = e(g, pk_{R,2})$ and $e(pk_{R,1}, g_2) = e(g, pk_{R,3})$. If not, output $\perp$ and abort the algorithm. Otherwise,the sender first selects a random value $s \in Z_p^*$ and $R \in \{0,1\}^n$ such that $M \| R \in \Re$, and compute:(Let w be a n-bit string and w_i be the i-th bit w).

$$\begin{aligned} \delta_1 &= \phi(M\|R) \cdot e(pk_{R,2}, pk_{R,3})^s, \\ \delta_2 &= g^s, \\ \delta_3 &= (u' \prod_{i=1}^{i=n} u_i^{d_{R,i}})^s, \\ \delta_4 &= sk_{S,2}, \\ \delta_5 &= sk_{S,1} \cdot F(w)^s \end{aligned}$$

where $w = H(\delta_1, \delta_2, \delta_3, \delta_4, R, pk_S, pk_R) \in \{0,1\}^m$ and $F(w) = v' \prod_{j=1}^{m} v_j^{w_j}$.

Unsigncrypt. Upon receiving a ciphertext $C = (\delta_1, \delta_2, \delta_3, \delta_4, \delta_5)$, first compute

$$\phi^{-1}(\delta_1 \cdot e(\delta_3, sk_{R,2}) / e(\delta_2, sk_{R,1})) \rightarrow M \| R$$

and then check whether the public key pk_S is correctly formed, by checking $e(pk_{S,1}, g_1) = e(g, pk_{S,2})$ and $e(pk_{S,1}, g_2) = e(g, pk_{S,3})$. If not, output $\perp$ and abort the algorithm. Otherwise, Accept the message M if

$$e(\delta_5, g) = e(pk_{S,2}, pk_{S,3})\, e(u' \prod_{i=1}^{i=n} u_i^{d_{S,i}}, \delta_4) e(F(w), \delta_2),$$

where $w = H(\delta_1, \delta_2, \delta_3, \delta_4, R, pk_S, pk_R) \in \{0,1\}^m$.

It is easy to see the proposed scheme is consistent. In the next Section, we will give a formal security proof.

5 Analysis of the Improved Scheme

5.1 Security Analysis

We now prove that the above proposed scheme is secure in the standard model. Our proof very much falls along the lines of the security proof in Liu et al.'s scheme [14].

Theorem 1. *The new CLSC scheme is indistinguishable against adaptive chosen ciphertext attacks (IND - CLSC - CCA) in the standard model under the decisional BDH intractability assumption.*

This theorem follows Lemmas 1 and 2.

Lemma 1. *The new CLSC scheme is indistinguishable against the Type I attacker in the standard model if the decisional BDH assumption holds.*

Proof. Assume there exists a type I adversary $\mathcal{A}_I$ against our scheme. We construct a PPT simulator $\mathcal{B}$ that makes use of $\mathcal{A}_I$ to solve the **DBDH** problem with probability at least ϵ' and in time at most t'. $\mathcal{B}$ is given a **DBDH** problem instance $(g, A = g^a, B = g^b, C = g^c, Z)$ and replies the queries of $\mathcal{A}_I$ as follows.

Setup. Let $l_v = 2(q_{pp} + q_p + q_s + q_u)$ and $l_w = 2q_u$. $\mathcal{B}$ randomly chooses the following elements:

1) Two integers $k_v (0 \le k_v \le 2^\ell \cdot n)$, $k_w (0 \le k_w \le m)$. We assume that $2^\ell (n+1) l_v < p$, $(m+1) l_w < p$ for the given values of n and m.
2) An integer $x' (x' \in Z_{l_v})$ and a vector $\overrightarrow{X} = (x_i)_n (x_i \in Z_{l_v})$.
3) An integer $z' (z' \in Z_{l_w})$ and a vector $\overrightarrow{Z} = (z_j)_m (z_j \in Z_{l_w})$.

4) Two integers $y', t' \in Z_p$ and three vectors $\overrightarrow{Y} = (y_i)_n (y_i \in \mathbf{Z_p})$, $\overrightarrow{T} = (t_j)_m (t_j \in \mathbf{Z_p})$, $\overrightarrow{w} = (w_j)_m (w_j \in \mathbf{Z_2})$.

Identity ID will be represented as n dimensional vectors $d_{ID} = (d_{ID,1}, \ldots, d_{ID,n})$ where each $d_{ID,i}$ is an ℓ-bit integer, and $n' = n \cdot \ell$ is the length of an identity ID in binary string representation. For convenience, we define as follows:

$$\begin{aligned} F(d_{ID}) &= x' - l_v k_v + \sum_{i=1}^{i=n} x_i d_{ID,i}, \\ J(d_{ID}) &= y' + \sum_{i=1}^{i=n} y_i d_{ID,i}, \\ K(w) &= z' - l_w k_w + \sum_{j=1}^{j=m} z_j w_j, \\ L(w) &= t' + \sum_{j=1}^{j=m} w_j t_j. \end{aligned}$$

Then the challenger constructs a set of public parameters as follows:

$$\begin{aligned} g_1 &= g^a, \\ g_2 &= g^b, \\ u' &= g_2^{x'-l_v k_v} g^{y'}, \\ v' &= g_2^{z'-l_w k_w} g^{t'}, \\ u_i &= g_2^{x_i} g^{y_i}, \\ v_j &= g_2^{z_j} g^{t_j}. \end{aligned}$$

Note that the master secret key will be $g_2^a = g^{ab}$ and the following equation holds:

$$\begin{aligned} u' \prod_{i=1}^{i=n} u_i^{d_{ID,i}} &= g_2^{F(d_{ID})} g^{J(d_{ID})}, \\ v' \prod_{j=1}^{j=m} v_j^{w_j} &= g_2^{K(w)} g^{L(w)}. \end{aligned}$$

Phase 1. In the query phase, $\mathcal{B}$ answers the queries of $\mathcal{A}_I$ as follows:

Public-Key-Broadcast-Oracle. Upon receiving a query for a public key of an identity ID, if (ID, pk_{ID}) exists in PublicKeyList, $\mathcal{B}$ returns pk_{ID} as the answer. Otherwise, $\mathcal{B}$ runs the algorithm **UserKeyGen** to generate public key $pk_{ID} = \{pk_{ID,1}, pk_{ID,2}, pk_{ID,3}\} = \{g^{x_{ID}}, g_1^{x_{ID}}, g_2^{x_{ID}}\}$. $\mathcal{B}$ adds (ID, x_{ID}) to SecretValueList and adds (ID, pk_{ID}) to PublicKeyList, then returns the public key pk_{ID} as the answer.

Partial-Private-Key-Extract-Oracle. Upon receiving a query for a partial private key of an identity ID, $\mathcal{B}$ first searches PartialPrivateKeyList for a tuple (ID, psk_{ID}). If it exists, $\mathcal{B}$ returns (ID, psk_{ID}) as the answer. Otherwise, $\mathcal{B}$ can construct a partial private key by assuming $F(d_{ID}) \neq 0 \bmod p$. $\mathcal{B}$ randomly chooses $r \in \mathbf{Z_p}$ and computes a partial private key:

$$\begin{aligned} & psk_{ID} = (psk_{ID,1}, psk_{ID,2}) \\ = \; & (g_1^{-J(d_{ID})/F(d_{ID})} (u' \prod_{i=1}^{i=n} u_i^{d_{ID,i}})^r, g_1^{-1/F(d_{ID})} g^r). \end{aligned}$$

psk_{ID} is a valid partial private key for the identity ID shown as follows.

$$\begin{aligned} psk_{ID,1} &= g_1^{-J(d_{ID})/F(d_{ID})} (u' \prod_{i=1}^{i=n} u_i^{d_{ID,i}})^r \\ &= g_2^a (g_2^{F(d_{ID})} g^{J(d_{ID})})^{r-a/F(d_{ID})} \\ &= g_2^a (u' \prod_{i=1}^{i=n} u_i^{d_{ID,i}})^{r'}, \\ psk_{ID,2} &= g_1^{-1/F(d_{ID})} g^r = g^{r-a/F(d_{ID})} = g^{r'} \end{aligned}$$

where $r' = r - a/F(d_{ID})$. From $-p < F(d_{ID}) < p$, we conclude that $F(d_{ID}) = 0 \bmod p$ implies $F(d_{ID}) = 0 \bmod l_v$, so $F(d_{ID}) \neq 0 \bmod l_v$ suffices to have $F(d_{ID}) \neq 0 \bmod p$. $\mathcal{B}$ adds (ID, psk_{ID}) to its PartialPrivateKeyList and returns the partial private key psk_{ID} as the query output. If, on the other hand, $F(d_{ID}) = 0 \bmod p$, $\mathcal{B}$ aborts and randomly chooses its guess β' of β.

Private-Key-Extract-Oracle. Upon receiving a query for a private key of an identity ID, if the PrivateKeyList contains (ID, sk_{ID}), $\mathcal{B}$ returns sk_{ID}. Otherwise, $\mathcal{B}$ can construct a private key by assuming $F(d_{ID}) \neq 0 \bmod p$. $\mathcal{B}$ searches SecretValueList to find out x_{ID}. If it does not exist, $\mathcal{B}$ runs the algorithm **UserKeyGen** to generate secret-public key pair (x_{ID}, pk_{ID}), and adds (ID, x_{ID}) to SecretValueList and adds (ID, pk_{ID}) to PublicKeyList, then $\mathcal{B}$ chooses $r \in Z_p$ randomly and computes

$$\begin{aligned} sk_{ID,1} &= (g_1^{x_{ID}^2})^{-J(d_{ID})/F(d_{ID})} (u' \prod_{i=1}^{i=n} u_i^{d_{ID,i}})^r \\ &= g_2^{ax_{ID}^2} (g_2^{F(d_{ID})} g^{J(d_{ID})})^{r-ax_{ID}^2/F(d_{ID})} \\ &= g_2^{ax_{ID}^2} (u' \prod_{i=1}^{i=n} u_i^{d_{ID,i}})^t, \\ sk_{ID,2} &= (g_1^{x_{ID}^2})^{-1/F(d_{ID})} g^r \\ &= g^{r-ax_{ID}^2/F(d_{ID})} = g^t, \end{aligned}$$

where $t = r - ax_{ID}^2/F(d_{ID})$. $\mathcal{B}$ adds (ID, sk_{ID}) to PrivateKeyList and returns the private key

sk_{ID}. If, on the other hand, $F(d_{ID}) = 0 \bmod p$, $\mathcal{B}$ aborts and randomly chooses its guess β' of β.

Public-Key-Replacement-Oracle. Upon receiving a query for replacing the current public key pk_{ID} of an identity ID with a new and valid public key pk'_{ID}, $\mathcal{B}$ finds out pk_{ID} in its PublicKeyList, and replaces it with the new public key pk'_{ID}. If pk_{ID} does not exist, $\mathcal{B}$ directly sets $pk_{ID} = pk'_{ID}$, while the adversary delivers x'_{ID} to $\mathcal{B}$. Then $\mathcal{B}$ adds (ID, x_{ID}) to SecretValueList and adds (ID, pk_{ID}) to PublicKeyList. $\mathcal{B}$ sets $sta = 1$ for the identity ID.

Signcrypt-Oracle. Upon receiving a query for a message M and identities ID_S and ID_R, if $F(d_S) \neq 0 \bmod p$, $\mathcal{B}$ obtains the public key pk_R of ID_R and the private key sk_S of ID_S by running **Public-Key-Broadcast-Oracle** and **Private-Key-Extract-Oracle**, then runs the **Signcrypt** algorithm to create a ciphertext δ and sends it to $\mathcal{A}_I$. If $F(d_S) = 0 \bmod p$ $\mathcal{B}$ aborts and randomly chooses its guess β' of β.

Unsigncrypt-Oracle. Upon receiving a unsigncryption query on a ciphertext $\delta = (\delta_1, \delta_2, \delta_3, \delta_4, \delta_5)$, and identities ID_S and ID_R, $\mathcal{B}$ computes the unsigncryption as follows:

1) If $sta = 0$ for ID_R, $\mathcal{B}$ obtains the private key sk_R of ID_R by running **Private-Key-Extract-Oracle** (assume $F(d_R) \neq 0 \bmod l_v$), then runs the **Unsigncrypt** algorithm to recover the message M. $\mathcal{B}$ executes the verification part of the **Unsigncrypt** algorithm. If the verification does not succeed, $\mathcal{B}$ returns a failure symbol $\perp$. Otherwise, returns M to $\mathcal{A}_I$.
2) If $sta = 1$ for ID_R, or $F(d_R) = 0 \bmod l_v$, $\mathcal{B}$ will try to decrypt the ciphertext δ. Assume $K(w) \neq 0 \bmod l_w$, where $w = H(\delta_1, \delta_2, \delta_3, \delta_4, R, pk_S, pk_R)$. $\mathcal{B}$ retrieves the secret value x_R s.t. pk_R, and computes $g_2^s = (\delta_5/(sk_{S,1}\delta_2^{L(w)}))^{1/K(w)}$ and $M \| R = \Phi^{-1}(\delta_1/e(g_1, g_2^s)^{x_R^2})$. $\mathcal{B}$ executes the verification part of the **Unsigncrypt** algorithm. If the verification does not succeed, $\mathcal{B}$ returns a failure symbol $\perp$. Otherwise, returns M to $\mathcal{A}_I$. If $L(w) = 0 \bmod l_w$, $\mathcal{B}$ aborts and randomly chooses its guess β' of β.

Challenge. At the end of the first stage, $\mathcal{A}_I$ outputs two equal length messages M_0, M_1 together with two identities ID_S and ID_R on which it wishes to be challenged. If $F(d_{R^*}) \neq 0 \bmod l_v$, $\mathcal{B}$ aborts. Otherwise, chooses a random bit γ from {0, 1} and constructs a ciphertext of M_γ as follows. Let pk_{S^*}, pk_{R^*} be ID_{S^*}, ID_{R^*}'s public keys, respectively. $\mathcal{B}$ retrieves the secret values x_{S^*}, x_{R^*}, $t_{S^*} \in Z_p$ and $R \in \{0,1\}^n$ such that $M_\gamma \| R \in \Re$, then computes as follows:

$$\begin{aligned}
\delta_1^* &= \phi(M_\gamma \| R) \cdot Z^{x_{R^*}^2}, \\
\delta_2^* &= C, \\
\delta_3^* &= C^{J(d_{R^*})}, \\
\delta_4^* &= (g_1^{x_{S^*}^2})^{-1/F(d_{S^*})} g^{t_{S^*}}, \\
w^* &= H(\delta_1^*, \delta_2^*, \delta_3^*, \delta_4^*, R, pk_{S^*}, pk_{R^*}).
\end{aligned}$$

If $K(w^*) \neq 0 \bmod p$, $\mathcal{B}$ aborts. Otherwise, $\mathcal{C}$ sets $\delta_5^* = (g_1^{x_{S^*}^2})^{-J(d_{S^*})/F(d_{S^*})} (u' \prod_{i=1}^{i=n} u_i^{d_{S^*,i}})^{t_{S^*}} C^{L(w^*)}$. $\mathcal{B}$ returns the ciphertext $\delta^* = (\delta_1^*, \delta_2^*, \delta_3^*, \delta_4^*, \delta_5^*)$ to the adversary.

Phase 2. $\mathcal{A}_I$ continues to perform the same type of queries made in Phase 1. But in this phase, $\mathcal{A}_I$ can not make any **Unsigncrypt** query on the challenge ciphertext δ^* for ID_{S^*}, ID_{R^*}.

Guess. Finally, $\mathcal{A}_I$ outputs a guess γ' of γ. If $\gamma' = \gamma$ then $\mathcal{B}$ outputs 1 indicating $Z = e(g,g)^{abc}$, and else outputs 0 indicating Z is a random element of G_T.

□

Remark 1. Liu et al. showed their scheme is secure if the user's public key is with the correctly form $e(g_1, g_2)^{x_{ID}}$ i.e the $\mathcal{A}_I$ can replace public key only by choosing a different secret value x'_{ID}. However, in [15, 17], Liu et al.'s scheme is showed that a Type I adversary $\mathcal{A}_I$ can cheat the sender and decrypt the ciphertext by replacing receiver's public key with $e(g,g)^{x'_R}$. The weakness in [14] is that receiver's public key pk_R is just a group element $e(g_1, g_2)^{x_R}$, and it can not check whether the public key pk_R is correctly formed during signcryption stage. In order to defend against attacks [15, 17], we revise **UserkeyGen** so that the receiver's public key pk_R can be checked whether it is correctly formed during signcryption stage. We omit the analysis of the success probability and the time complexity, which are similar to that of Liu et al. [14].

Lemma 2. *The new CLSC scheme is indistinguishable against the Type II attacker in the standard model if the decisional BDH assumption holds.*

Proof. Assume there exists a type II adversary $\mathcal{A}_{II}$ against our scheme. We construct a PPT simulator $\mathcal{B}$ that makes use of $\mathcal{A}_{II}$ to solve the **DBDH** problem with probability at least ϵ' and in time at most t'. $\mathcal{B}$ is given a**DBDH** problem instance $(g, A = g^a, B = g^b, C = g^c, Z)$ and replies the queries of $\mathcal{A}_{II}$ as follows.

Setup. Let $l_v = 2(q_p + q_s + q_u)$ and $l_w = 2q_u$. The Type II adversary $\mathcal{A}_{II}$ chooses a random integer $\alpha \in \mathbf{Z_p}$ as the master secret key and computes $g_1 = A^\alpha$. The other public parameters are identical to those of Theorem 1. Then $\mathcal{A}_{II}$ sends all public parameters and the master secret key α to $\mathcal{B}$.

Phase 1. $\mathcal{A}_{II}$ can compute partial private key of any identity by itself and carry out the following queries.

Public-Key-Broadcast-Oracle. Upon receiving a query for a public key of an identity ID, if (ID, pk_{ID}) exists in PublicKeyList, $\mathcal{B}$ returns pk_{ID} as the answer. Otherwise, $\mathcal{B}$ runs the algorithm **User-Key-Gen** to generate public key $pk_{ID} = (g^{x_{ID}}, A^{\alpha x_{ID}}, B^{x_{ID}})$, $\mathcal{B}$ adds (ID, x_{ID}) to SecretValueList and adds (ID, pk_{ID}) to PublicKeyList, and returns the public key to $\mathcal{A}_{II}$.

Private-Key-Extract-Oracle. Upon receiving a query for a private key of an identity ID, if the PrivateKeyList contains (ID, sk_{ID}), $\mathcal{B}$ returns sk_{ID}. Otherwise, $\mathcal{B}$ can construct a private key by assuming $F(d_{ID}) \neq 0 \bmod p$. $\mathcal{B}$ first searches SecretValueList to find out x_{ID}. If it does not exist, $\mathcal{B}$ runs the algorithm **UserKeyGen** to generate secret-public key pair (x_{ID}, pk_{ID}), and adds (ID, x_{ID}) to SecretValueList and adds (ID, pk_{ID}) to PublicKeyList, then $\mathcal{B}$ chooses $r \in Z_p$ randomly and computes

$$\begin{aligned} sk_{ID,1} &= (A^{\alpha x_{ID}^2})^{-J(d_{ID})/F(d_{ID})}(u' \prod_{i=1}^{i=n} u_i^{d_{ID,i}})^r \\ &= g_2^{a\alpha x_{ID}^2}(g_2^{F(d_{ID})} g^{J(d_{ID})})^{r - a\alpha x_{ID}^2/F(d_{ID})} \\ &= g_2^{a\alpha x_{ID}^2}(u' \prod_{i=1}^{i=n} u_i^{d_{ID,i}})^t, \\ sk_{ID,2} &= (A^{\alpha x_{ID}^2})^{-1/F(d_{ID})} g^r \\ &= g^{r - a\alpha x_{ID}^2/F(d_{ID})} = g^t, \end{aligned}$$

where $t = r - a\alpha x_{ID}^2/F(d_{ID})$. $\mathcal{B}$ adds (ID, sk_{ID}) to the PrivateKeyList and returns the private key sk_{ID}. If $F(d_{ID}) = 0 \bmod p$, $\mathcal{B}$ simply aborts and randomly outputs a guess β' of β.

Signcrypt-Oracle. For a signcryption query for a message M and identities ID_S and ID_R, $\mathcal{B}$ answers the signcryption query in the same way as Lemma 1.

Unsigncrypt-Oracle. Upon receiving a unsigncryption query on a ciphertext $\delta = (\delta_1, \delta_2, \delta_3, \delta_4, \delta_5)$, and a sender's identity ID_S and a receiver's identity ID_R, $\mathcal{B}$ computes the unsigncryption as follows:

1) If $F(d_R) \neq 0 \bmod l_v$, $\mathcal{B}$ searches PrivateKeyList to find out (ID_R, sk_R),then performs the Unsigncrypt algorithm to recover the message M, and sends it to A_{II}. If the corresponding entry does not exist, $\mathcal{B}$ obtains the private key sk_R of ID_R by running **Private-Key-Extract-Oracle**, then runs the **Unsigncrypt** algorithm to recover the message M. $\mathcal{B}$ executes the verification part of the **Unsigncrypt** algorithm. If the verification does not succeed, $\mathcal{B}$ returns a failure symbol $\perp$. Otherwise, returns M to $\mathcal{A}_{II}$.
2) If $F(d_R) = 0 \bmod l_v$, $\mathcal{B}$ will try to decrypt the ciphertext δ. Assume $K(w) \neq 0 \bmod l_w$, where $w = H(\delta_1, \delta_2, \delta_3, \delta_4, R, pk_S, pk_R)$. $\mathcal{B}$ searches PrivateKeyList to obtain ID_S's private key sk_S (to deal with the insider security, we assume that the adversary has access to the private key of the sender) and retrieve the secret value x_R s.t. pk_R. $\mathcal{B}$ can compute $g_2^s = (\delta_5/(sk_{S,1}\delta_2^{L(w)}))^{1/K(w)}$, $M \| R = \Phi^{-1}(\delta_1 / e(A^\alpha, g_2^s)^{x_R^2})$. Then $\mathcal{B}$ executes the verification part of the **Unsigncrypt** algorithm. If the verification does not succeed, $\mathcal{B}$ returns a failure symbol $\perp$. Otherwise, returns M to $\mathcal{A}_{II}$. If $L(w) = 0 \bmod l_w$, $\mathcal{B}$ aborts and randomly chooses its guess β' of β.

Challenge. At the end of the first stage, $\mathcal{A}_{II}$ outputs two equal length messages M_0, M_1 together with two identities ID_S and ID_R on which it wishes to be challenged. If $F(d_{R^*}) \neq 0 \bmod l_v$, $\mathcal{B}$ aborts. Otherwise, chooses a random bit γ from $\{0, 1\}$ and constructs a ciphertext of M_γ as follows. Let pk_{S^*}, pk_{R^*} be ID_{S^*}, ID_{R^*}'s public keys, respectively. $\mathcal{B}$ retrieves the secret values x_{S^*}, x_{R^*} and randomly chooses a bit $\gamma \in \{0,1\}$, $t_{S^*} \in Z_p$ and $R \in \{0,1\}^n$ such that $M_\gamma \| R \in \Re$, then computes as follows: $\delta_1^* = \phi(M_b \| R) \cdot Z^{\alpha x_{R^*}^2}, \delta_2^* = C, \delta_3^* = C^{J(d_{R^*})}, \delta_4^* = (g_1^{x_{S^*}^2})^{-1/F(d_{S^*})} g^{t_{S^*}}, w^* = H(\delta_1^*, \delta_2^*, \delta_3^*, \delta_4^*, R, pk_{S^*}, pk_{R^*})$. If $K(w^*) \neq 0 \bmod p$, $\mathcal{B}$ aborts. Otherwise, $\mathcal{C}$ sets $\delta_5^* = (g_1^{x_{S^*}^2})^{-J(d_{S^*})/F(d_{S^*})}(u' \prod_{i=1}^{i=n} u_i^{d_{S^*,i}})^{t_{S^*}} C^{L(w^*)}$. $\mathcal{B}$ returns the ciphertext $\delta^* = (\delta_1^*, \delta_2^*, \delta_3^*, \delta_4^*, \delta_5^*)$ to the adversary.

Phase 2. $\mathcal{A}_{II}$ continues to perform the same type of queries made in Phase 1. But in this phase, $\mathcal{A}_{II}$ can not make any **Unsigncrypt** query on the challenge ciphertext δ^* for ID_{S^*}, ID_{R^*}.

Guess. Finally, $\mathcal{A}_{II}$ outputs a guess γ' of γ. If $\gamma' = \gamma$ then $\mathcal{B}$ outputs 1 indicating $Z = e(g,g)^{abc}$, and else outputs 0 indicating Z is a random element of G_T.

□

Remark 2. Weng et al. [21] proved that Liu et al's scheme is not indistinguishable against a Type II adversary. That is, given a challenged ciphertext δ^*, $\mathcal{A}_{II}$ could convert the challenged ciphertext δ^* into a new valid ciphertext δ' in phase 2. When adversary issues an unsigncryption query on the ciphertext δ', the challenger has to return the underlying message M_γ to $\mathcal{A}_{II}$. With M_γ, adversary $\mathcal{A}_{II}$ can certainly know the value γ, and thus wins

the game. One of the main difference between Liu et al. and our proof is the signcryption ciphertext that returns at the stage of **Challenge**. In our proof, the challenged ciphertext δ^* includes a random binary string R which $\mathcal{A}_{II}$ does not know, so the above defect can be avoided.

Theorem 2. *The new CLSC scheme is existentially unforgeable against chosen message attacks (EUF-CLSC-CMA) in the standard model under the CDH intractability assumption.*

This theorem follows Lemma 3 and 4.

Lemma 3. *The new CLSC scheme is existentially unforgeable against the Type I attacker in the standard model if the CDH assumption holds.*

Proof. Assume that there exists a Type I forger $\mathcal{A}_I$ against our scheme. In the following, we construct an algorithm $\mathcal{B}$ to solve the CDH problem.

Suppose $\mathcal{B}$ is given a random instance of the CDH problem (g^a, g^b). Its goal is to output g^{ab}. The simulation process is the same as that described in Lemma 1. Finally, the adversary $\mathcal{A}_I$ produces a new ciphertext $\delta^* = (\delta_1^*, \delta_2^*, \delta_3^*, \delta_4^*, \delta_5^*)$ on message $w^* = H(\delta_1^*, \delta_2^*, \delta_3^*, \delta_4^*, R, pk_S, pk_R)$. If $F(d_{S^*}) \neq 0 \bmod p$ or $K(w^*) \neq 0 \bmod p$, then $\mathcal{B}$ aborts. Otherwise, $F(d_{S^*}) = 0 \bmod p$ and $K(w^*) = 0 \bmod p$, $\mathcal{B}$ computes

$$\begin{aligned} & \frac{\delta_5^*}{(\delta_4^*)^{J(d_{S^*})}(\delta_2^*)^{L(w^*)}} \\ = & \frac{g_2^{ax_{S^*}^2}(u'\prod_{i=1}^{i=n} u_i^{d_{S^*,i}})^{t_{S^*}}(\delta_2^*)^{L(w^*)}(v'\prod_{j=1}^{j=m} v_j^{w_j})^s}{g^{J(d_{S^*})t_{S^*}} g^{L(w^*)r''}} \\ = & g_2^{ax^2} \\ = & g^{abx_{S^*}^2}. \end{aligned}$$

$\mathcal{B}$ retrieves the secret value x_{S^*} s.t. pk_{S^*} and thus can output g^{ab} as the solution to the CDH problem instance. □

Remark 3. Liu et al. showed their scheme is secure against an existential forgery for adaptive chosen message attacks (EUF-CLSC-CMA) if the user's public key is with the correctly form $e(g_1, g_2)^{x_{ID}}$ i.e the $\mathcal{A}_I$ can replace public key only by choosing a different secret value x'_{ID}. However, in [15], Liu et al. scheme is showed that a Type I adversary $\mathcal{A}_I$ can cheat the receiver and forge a valid signcrypted text by replacing the sender's public key $e(g, g)^{x'_S}$. The weakness in [14] is that the sender's public key pk_S is just a group element $e(g_1, g_2)^{x_S}$, and it can not check whether the public key pk_S is correctly formed during unsigncrypt stage. In order to defend against attacks [15], we revise **UserkeyGen** so that the receiver can check whether the sender's public key pk_S is correctly formed during the unsigncrypt stage. We omit the analysis of the success probability and the time complexity, which are similar to that of Liu et al. [14].

Lemma 4. *The new CLSC scheme is existentially unforgeable against the Type II attacker in the standard model if the CDH assumption holds.*

Proof. Assume that there exists a Type II forger $\mathcal{A}_{II}$ against our scheme forger. In the following,we construct an algorithm $\mathcal{B}$ to solve the CDH problem.

Suppose $\mathcal{B}$ is given a random instance of the CDH problem (g^a, g^b). Its goal is to output g^{ab}. The simulation process is the same as that described in Lemma 2.

Finally, if $\mathcal{B}$ does not abort, the adversary $\mathcal{A}_{II}$ returns a new ciphertex $\delta^* = (\delta_1^*, \delta_2^*, \delta_3^*, \delta_4^*, \delta_5^*)$ on message $w^* = H(\delta_1^*, \delta_2^*, \delta_3^*, \delta_4^*, R, pk_{S^*}, pk_{R^*})$ where w^* has never been queried. If $F(d_{S^*}) \neq 0 \bmod p$ or $K(w^*) \neq 0 \bmod p$, then $\mathcal{B}$ aborts. Otherwise, $F(d_{S^*}) = 0 \bmod p$ and $K(w^*) = 0 \bmod p$, $\mathcal{B}$ computes

$$\begin{aligned} & \frac{\delta_5^*}{(\delta_4^*)^{J(d_{S^*})}(\delta_2^*)^{L(w^*)}} \\ = & \frac{g_2^{a\alpha x_{S^*}^2}(u'\prod_{i=1}^{i=n} u_i^{d_{S^*,i}})^{t_{S^*}}(\delta_2^*)^{L(w^*)}(v'\prod_{j=1}^{j=m} v_j^{w_j})^s}{g^{J(d_{S^*})t_{S^*}} g^{L(w^*)r''}} \\ = & g_2^{a\alpha x^2} \\ = & g^{ab\alpha x_{S^*}^2}. \end{aligned}$$

Since $\mathcal{B}$ has the value x_{S^*} and the master secret key α, it can output g^{ab} as the solution to the CDH problem instance. □

Remark 4. Weng et al. [21] proved that a Type II adversary can use a ciphertext generated by a sender to arbitrarily forge signcryption on behalf of this sender. In our improved scheme, we embed a random binary string R into the signcryption ciphertex. Since $\mathcal{A}_{II}$ does not know R, he can not successfully launch the same attacks as in [21].

5.2 Performance Analysis

The existing CLSC schemes without using random oracles are given in [10, 14]. However, there exists security weakness in these two schemes [10, 14], that is, we can not check whether the user's public key is correctly formed during signcryption and unsigncryption stages. To avoid the security weakness, we has to add verification equations which results in our improved scheme has more computational cost in the signcryption and unsigncryption stages. Due to adopting Naccache's methods [16] in our improved scheme, identity ID with $n' = n \cdot \ell$ bit length can be reduced to n dimensional vectors $d_{ID} = (d_{ID,1}, \ldots, d_{ID,n})$ where each $d_{ID,i}$ is an ℓ-bit integer. So the new scheme is with a smaller master public size than the other existing CLSC schemes [10, 14] The detailed comparisons of our scheme with these schemes [10, 14] are summarized in Table 1 where H denotes the Hash function computation, E_{G_T} denotes an exponentiation computation in G_T, P denotes a pairing computation, and $| aG |$ denotes the binary length of a elements in G.

Table 1: Comparisons among different CLSC schemes

Schemes	**Public parameter size**	**Operations**	Ciphertext size	Security
[10]	$\|(m+n\cdot l+4)G\|$	$2H+1E_{G_T}+5P$	$\|4G\|+\|1G_T\|$	No
[14]	$\|(m+n\cdot l+4)G\|$	$2H+1E_{G_T}+5P$	$\|4G\|+\|1G_T\|$	No
Our Scheme	$\|(m+n+4)G\|$	$2H+1E_{G_T}+15P$	$\|4G\|+\|1G_T\|$	Yes

From Table 1, we know our scheme has more computational cost in the signcryption and unsigncryption stages, but it can provide the provable security and has the same ciphertext size as schemes [10, 14] and has smaller public parameter size than schemes [10, 14].

6 Conclusions

Liu et al. [14] proposed the first CLSC scheme in the standard model. However, their scheme has some security weaknesses [15, 17, 21]. In this paper, we propose a corrected version of Liu et al's scheme and prove the new scheme is secure against Type I and Type II (a malicious-but-passive KGC) adversaries in the standard model. Our new scheme not only provides the provable security but also has smaller public parameter size than the previous schemes [10, 14].

Acknowledgments

We would like to thank the anonymous reviewers for giving valuable comments. This work is supported by NSFC (Grant Nos. 61272057, 61202434, 61170270, 61100203, 61003286, 61121061), the Fundamental Research Funds for the Central Universities (Grant No. 2012RC0612, 2011YB01).

References

[1] S. S. Al-Riyami and K. Paterson, "Certificateless public key cryptography," in *Advances in Cryptology (Asiacrypt'03)*, pp. 452–473, 2003.

[2] M. H. Au, Y. Mu, J. Chen, D. S. Wong, J. K. Liu, and G. Yang, "Malicious KGC attacks in certificateless cryptography," in *Proceedings of the 2007 ACM Symposium on Information, Computer and Communications Security*, pp. 302–311, 2007.

[3] M. Barbosa and P. Farshim, "Certificateless signcryption," in *Proceedings of the 2008 ACM Symposium on Information, Computer and Communications Security*, pp. 369–372, 2008.

[4] M. Bellare, A. Boldyreva, and A. Palacio, "An uninstantiable random-oracle-model scheme for a hybrid-encryption problem," in *Advances in Cryptology (Eurocrypt'04)*, pp. 171–188, 2004.

[5] M. Bellare and P. Rogaway, "Random oracles are practical: A paradigm for designing efficient protocols," in *Proceedings of the first ACM Conference on Computer and Communications Security*, pp. 62–73, 1993.

[6] R. Canetti, O. Goldreich, and S. Halevi, "The random oracle methodology, revisited," *Journal of the ACM*, vol. 51, no. 4, pp. 557–594, 2004.

[7] H. Chen, Y. Li, and J. Ren, "A practical identity-based signcryption scheme," *International Journal of Network Security*, vol. 15, no. 6, pp. 484–489, 2013.

[8] A. W. Dent, B. Libert, and K. G. Paterson, "Certificateless encryption schemes strongly secure in the standard model," in *Proceedings of the 11th International Workshop on Practice and Theory in Public Key Cryptography*, pp. 344–359, 2008.

[9] Y. H. Hwang, J. K. Liu, and S. S. M. Chow, "Certificateless public key encryption secure against malicious KGC attacks in the standard model," *Journal of Universal Computer Science*, vol. 14, no. 3, pp. 463–480, 2008.

[10] Z. Jin, Q. Wen, and H. Zhang, "A supplement to Liu et al.'s certificateless signcryption scheme in the standard model," in *IACR Eprint Archive*, 2010.

[11] F. Li, X. Xin, and Y. Hu, "Id-based signcryption scheme with (t,n) shared unsigncryption," *International Journal of Network Security*, vol. 3, no. 2, pp. 155–159, 2006.

[12] P. Li, M. He, and W. Liu, "Efficient and provably secure certificateless signcryption from bilinear pairings," *Journal of Computational Information Systems*, vol. 6, no. 11, pp. 3643–3650, 2011.

[13] J. K. Liu, M. H. Au, and W. Susilo, "Self-generated-certificate public key cryptography and certificateless signature/encryption scheme in the standard model," in *Proceedings of the Second ACM Symposium on Information, Computer and Communications Security*, pp. 273–283, 2007.

[14] Z. Liu, Y. Hu, X. Zhang, and H. Ma, "Certificateless signcryption scheme in the standard model," *Information Sciences*, vol. 180, no. 3, pp. 452–464, 2010.

[15] S. Miao, F. Zhang, S. S. Li, and Y. Mu, "On security of a certificateless signcryption scheme," *Information Sciences*, vol. 232, pp. 475–481, 2013.

[16] D. Naccache, "Secure and practical identity-based encryption," *Iet Information Security*, vol. 1, no. 2, pp. 59–64, 2007.

[17] S. S. D. Selvi, S. S. Vivek, and C. P. Rangan, "Certificateless signcryption," in *Cryptology ePrint Archive, Report 2010/92*, 2010.

[18] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Advances in Cryptology (Crypto'84)*, pp. 47–53, 1984.
[19] M. Toorani and A. Beheshti, "Cryptanalysis of an elliptic curve-based signcryption scheme," *International Journal of Network Security*, vol. 10, no. 1, pp. 51–56, 2010.
[20] B. Waters, "Efficient identity-based encryption without random oracles," in *Advances in Cryptology (Eurocrypt'05)*, pp. 114–127, 2005.
[21] J. Weng, G. X. Yao, R. H. Deng, M. R. Chen, and X. X. Li, "Cryptanalysis of a certificateless signcryption scheme in the standard model," *Information Sciences*, vol. 181, no. 3, pp. 661–667, 2011.
[22] C. Wu and Z. Chen, "A new efficient certificateless signcryption scheme," in *The International Symposium on Information Science and Engieering*, pp. 661–664, 2008.
[23] H. Xiong, J. Hu, and Z. Chen, "Security flaw of an ECC-based signcryption scheme with anonymity," *International Journal of Network Security*, vol. 15, no. 4, pp. 317–320, 2013.
[24] H. Xiong, Z. G. Qin, and F. G. Li, "Certificateless public key encryption secure against malicious KGC attacks in the standard model," *Fundamenta Informaticae*, vol. 88, pp. 193–206, 2008.
[25] G. Yu, Z. Yang, S. Fan, Y. Shen, and W. Han, "Efficient certificateless signcryption scheme," in *Proceedings of the Third International Symposium on Electronic Commerce and Security Workshops*, pp. 55–59, 2010.
[26] Y. Zheng, G. Goos, J. Hartmanis, and J. V. Leeuwen, "Digital signcryption or how to achieve cost (signature and encryption) $\leq$ cost (signature) + cost(encryption)," in *Advances in Cryptology (Crypto'97)*, pp. 291–312, 1997.

Lin Cheng biography. He is currently a PhD candidate in State Key Laboratory of Networking and Switch Technology, Beijing University of Posts and Telecommunications. His present research interests are cryptography, information security and cloud computing..

Qiaoyan Wen biography. She received her BS and MS from Shanxi normal University in 1981 and 1984, respectively, and her PhD from Xidian University in 1997. Now, she is a professor of Beijing University of Posts and Telecommunications. Her present research interests include cryptography, information security, and cloud computing.

Adjustment Hiding Method Based on Exploiting Modification Direction

Chin-Feng Lee[1], Chin-Chen Chang[2,3], Pei-Yan Pai[2], and Chia-Ming Liu[3]
(Corresponding author: Chin-Chen Chang)

Department of Information Management, Chaoyang University of Technology[1]
168, Jifeng E. Rd., Wufeng District, Taichung, 41349 Taiwan, R.O.C.
Department of Computer Science, National Tsing Hua University[2]
101, Section 2, Kuang-Fu Rd., Hsinchu, 30013, Taiwan, R.O.C.
Department of Information Engineering and Computer Science, Feng Chia University[3]
100, Wenhwa Rd., Seatwen, Taichung 40724, Taiwan, R.O.C
(Email: alan3c@gmail.com)

(Received Mar. 21, 2013; revised and accepted Nov. 6, 2013)

Abstract

In the exploiting modification direction (EMD) method, a secret digit in $(2\boldsymbol{n}+1)$-ary notational system is embedded into a group that is consisted of $\boldsymbol{n}$ pixels. Only one pixel in a group at most is being modified by either increasing or decreasing one. Therefore, the maximum embedding rate of EMD method is $(\log_2(2\boldsymbol{n}+1))/n$ when $\boldsymbol{n} \geqq 2$. In order to increase the embedding capacity of EMD method, a new adjustment data hiding method based on $\boldsymbol{c}^{\boldsymbol{n}}$-ary notational system is proposed in this paper. In the proposed method, at most $\boldsymbol{n}$ pixels in a group can be modified and each pixel has $\boldsymbol{c}$ different ways of modification. As the result, the maximal embedding rate of the proposed method is $\log_2 \boldsymbol{c}$. The experimental results demonstrated that the proposed method provides a higher embedding capacity than the compared methods and a satisfied image quality of stego-image. The proposed method provides an average 1~4.75 bits per pixel (bpp) and an average peak signal-to-noise rate (***PSNR***) of 51.14~30.30 dB with different $\boldsymbol{c}$. In addition, the proposed method also inherits the advantageous properties of EMD method: the computation efficiency and the ability to resist steganographic attack.

Keywords: Embedding capacity, embedding rate, image hiding, steganographic attack

1 Introduction

As the Internet has developed rapidly and became more and more popular, it is quite common for people to transmit the data to others via Internet. However, the illegal attackers can easily steal the data through the public Internet. Image hiding is one of the data security technologies to protect the secret data from illegal interception [1-2, 4-8, 10-12]. In the image hiding, the secret data is embedded into an image called cover image. Once the secret data are embedded, the cover image becomes a so-called "stego-image", and then the data becomes invisible for the illegal attacker. That is, illegal attackers should not be able to notice the existence of the embedded data, even though they have carefully analyzed the stego-image. Two important issues of image hiding are preserving good image quality and increasing the embedding capacity at the same time [4-5]. However, this is a trade-off problem. If we want to improve the image quality, we would have to sacrifice the embedding capacity, and vice versa.

In recent years, many image hiding methods have been proposed. Turner [9] presented a simple hiding method called least significant bit (LSB) replacement method. In the LSB replacement method, the binary secret bits are embedded into the LSBs of pixels in the cover image by replacement operation. The maximum embedding capacity is quite limited. However, the LSB replacement method cannot resist against the statistical analysis [3, 11]. Milikainen [7] proposed the LSB matching revisited method based on pixel pair to improve the security of LSB replacement method. The LSB matching revisited method can resist the statistical analysis, since it does not possess the asymmetric property of LSB replacement method [7]. However, maximum embedding capacity of the LSB matching revisited method is not raised same with the LSB replacement method. Chang *et al.* [2] introduced a new image hiding method based on discrete cosine transform (DCT). In Chang *et al.*'s method, the cover image is first converted into frequency domain. After that, the secret data are embedded into the coefficients of the medium frequency. The embedding capacity of Chang *et al.*'s method is low since only a few coefficients can carry the secret data. In addition, Chang *et al.*'s method needs complicated computations to transform the cover image into the frequency domain and the stego-image into its spatial domain.

Zhang *et al.* [12] proposed a new data hiding method based on exploiting modification direction (EMD) method, called EMD method. The binary secret bits are converted

into a sequence of secret digits in a (2n+1)-ary notational system and a group with n pixels used to carry one secret digit. In the EMD method, only one pixel at most in a group is increased or decreased by one. Accordingly, the EMD method provides a high image quality but the theoretical maximum embedding rate of EMD method is only about 1.16 bits per pixel (bpp) when n=2. In practical experiment, the embedding rate for the EMD method reaches only 1 bpp at the base 5 of numerical system. Moreover, the embedding rate of the EMD method decreases along with an increase of n. Lee *et al*. [5] improved the EMD method based on pixel segmentation strategy. Lee *et al.*'s method can produce an embedding capacity 1.7 times more than that of EMD method, but the average peak signal-to-noise rate (***PSNR***) of Lee *et al.*'s method decreased 8 dB more than EMD method. The EMD-2 method is proposed to improve the EMD method by Kim *et al*. [4]. The EMD-2 method allows only two pixels at most in a group of n pixels are modified by increasing one or decreasing one. The EMD-2 method can provide 1.58 bpp and similar image quality of EMD method when n=2.

In this paper, an adjustable image hiding method for improving the EMD-based methods is proposed. In this method, the binary secret bits are firstly converted into a sequence of secret digits in c^n-ary notational system. After that, a group with n pixels is used to embed a secret digit, and n pixels are modified at most with c different ways of modification. With various c values, the proposed method can provide an average of 1~4.75 bpp and an average ***PSNR*** of 51.14~30.3 dB, which means the embedding capacity can be adjusted depending on the requirements of application. In addition, the proposed method also inherits the advantages of EMD method: the computation efficiency and the ability to resist steganographic attack [4-5]. The experimental results have demonstrated that the proposed method provides a higher embedding capacity while satisfying image quality of stego-image than the compared methods.

The rest of this paper is organized as follows. In Section 2, we review the EMD method. Then, in Section 3, we introduce the proposed method including embedding and extracting procedures for gray-level images. In Section 4, we make comparisons of embedding rate R and stego-image quality between the proposed method and other related methods. Finally, we will provide a conclusion of our work in Section 5.

2 The EMD Embedding Method

Zhang *et al.* [12] presented a new image hiding method, which is called exploiting modification direction (EMD) method. In this method, the secret message is converted into a sequence of binary bit streams. The secret bits are divided into N pieces with L bits, and each secret piece is presented as a decimal value by D digital numbers in a (2n+1)-ary notational system, where

$$\begin{aligned} L &= \lfloor D \times \log_2(2n+1) \rfloor, \\ N &= \left\lceil \frac{\text{Secret bits length}}{L} \right\rceil, \end{aligned} \tag{1}$$

n is a parameter to determine how many pixels of cover image are used to hide one secret digit.

In the embedding phase, EMD method firstly uses pseudo-random generator to permute all pixels of cover image according to a secret key. After that, EMD method partitions the permuted pixels into a series of groups. The group is denoted as a vector G_n=(g_1, g_2, …, g_n), which is consisted of n cover pixels. A weight vector W_n=(w_1, w_2, …, w_n)=(1, 2, …, n) is defined. Therefore, the EMD method defines an embedding function f as weighted sum function modulo (2n+1) for each group, a secret digit d can be carried by the n cover pixels, and at most one pixel is increased or decreased by one. f can be expressed as Equation (2):

$$f(g_1, g_2, \ldots, g_n) = \left[\sum_{i=1}^{n} (g_i \times w_i) \right] \bmod (2n+1) \tag{2}$$

After embedding a secret digit d in the group G_n, G_n is modified into $G_n' = (g_1', g_2', \ldots, g_n')$, which is defined according to following conditions:

1. $G_n' = (g_1', g_2', \ldots, g_n') = (g_1, g_2, \ldots, g_n)$, if d=f.
2. When $d \neq f$, computes s=d-f mod (2n+1) and $g_i' = \begin{cases} g_i, & \text{if } i \neq s \\ g_i + 1, & \text{if } i = s \end{cases}$, and $s \leq n$, for $i = 1, 2, ..., n$.
3. Otherwise, $g_i' = \begin{cases} g_i, & \text{if } i \neq 2n+1-s \\ g_i - 1, & \text{if } i = 2n+1-s \end{cases}$, and $s > n$, for $i = 1, 2, ..., n$

From the above properties, the EMD method allows only one pixel value to be modified in a group, or none of the pixel values in the group to be modified. That means, one pixel is either increased or decreased by one; otherwise, all the pixels in the group get no modification at all. Thus, we can generate a corresponding table by the above embedding strategy to modify the pixel value while the secret digit is embedded in the G_n. Table 1 demonstrates the corresponding table with n=3 in EMD.

For example, let a secret digit d be 2, n be 3, G_n=(3, 1, 5), so that the f is computed as (3×1+1×2+5×3) mod 7=6 and s=(2-6) mod 7=3. Since s=3 is less than n=3, g_3 is increased by one, and it is corresponded to Case 2 (see Table 1). Therefore, G_3' is calculated as (3+0, 1+0, 5+1)=(3, 1, 6) with the secret digit d=2 embedded.

In the extracting phase, the secret digit can be extracted from stego-group $G_n' = (g_1', g_2', \ldots, g_n')$ by the following extraction function shown as Equation (3).

$$f(g_1', g_2', \ldots, g_n') = \left[\sum_{i=1}^{n} (g_i' \times w_i) \right] \bmod (2n+1). \tag{3}$$

Table 1: The corresponding table with n=3 in the EMD method

i / W_3 / Case	1	2	3
	1	2	3
Case 1	0	0	0
Case 2	0	0	+1
Case 3	0	0	-1
Case 4	0	+1	0
Case 5	0	-1	0
Case 6	+1	0	0
Case 7	-1	0	0

Table 2: A mapping corresponding to the minimum modification vector with n=3, c=2 and k in the proposed method ($W_{3,2}$={2^0, 2^1, 2^2})

$W_{3,2}$ / MV / Case	$w_1=2^0$	$w_2=2^1$	$w_3=2^2$	k
	mv_1	mv_2	mv_3	
Case 1	0	0	0	0
Case 2	0	0	+1	4
Case 3	0	+1	0	2
Case 4	0	+1	+1	6
Case 5	+1	0	0	1
Case 6	+1	0	+1	5
Case 7	+1	+1	0	3
Case 8	+1	+1	+1	7

Table 3: The corresponding mapping with n=2, c=4 in the proposed method ($W_{2,4}$={4^0, 4^1})

$W_{2,4}$ / MV / Case	$w_1=4^0$	$w_2=4^1$	k
	mv_1	mv_2	
Case 1	0	0	0
Case 2	0	+1	4
Case 3	0	-1	12
Case 4	0	+2	8
Case 5	+1	0	1
Case 6	+1	+1	5
Case 7	+1	-1	13
Case 8	+1	+2	9
Case 9	-1	0	15
Case 10	-1	+1	3
Case 11	-1	-1	11
Case 12	-1	+2	7
Case 13	+2	0	2
Case 14	+2	+1	6
Case 15	+2	-1	14
Case 16	+2	+2	10

The EMD method provides a good quality of stego-image with an average ***PSNR*** value of more than 51 dB, and the theoretical maximal embedded rate $R=\frac{\log_2(2n+1)}{n}$ of EMD method is 1.16 bpp for the best case n=2.

3 Proposed Method

In this section, the proposed data hiding method is introduced in detail. The proposed method is consisted of embedding and extracting procedures. In the embedding procedure, the secret digits are embedded into a c^n-ary notational system in a group of pixels with c as a constant parameter, which implies there will be c status for a pixel to be changed for embedding a secret digit. In the extracting procedure, the secret data can be completely extracted from the stego-image. In a group of the proposed method, at most n pixels need to be modified, and each pixel has c modifications.

3.1 Embedding Procedure

Let S be a series of binary secret data and I be a gray-level image of $W\times H$ pixels. S is divided into N pieces with L bits and each secret piece is converted into a secret digit d based on c^n-ary notational system. The proposed method

firstly selects a key α to pseudo-randomly permute all pixels in I. After that, the proposed method partitions the permuted pixels into a series of groups. Each group contains n permuted pixels and is denoted as a vector, $G_n=(g_1, g_2, \ldots, g_n)$. In EMD method, only one pixel is changed at most by either increasing or decreasing one; otherwise, no modification is needed. The modification interval for a pixel to carry secret digit in the EMD method is [-1, 1], which narrows the interval's limit on the embedding capacity. In order to improve the capacity, the proposed method allows at most n pixels can be modified and the number of modifications for each cover pixel is c. Hence, the modification interval of the pixel in the proposed method is $[-\lceil 0.5\times(c-1)\rceil+1, \lceil 0.5\times(c-1)\rceil]$ where c is even; otherwise, the minimum modification interval of the pixel is $[-\lceil 0.5\times(c-1)\rceil, \lceil 0.5\times(c-1)\rceil]$ if c is odd.

In the proposed method, the weight vector is defined as $W_{n,c}=(w_1, w_2,\ldots, w_n)=(c^0, c^1,\ldots, c^{n-1})$. For example, let n and c be 3 and 2, respectively, so that $W_{3,2}=(2^0, 2^1, 2^2)=(1, 2, 4)$ and the minimum modification interval is [0, 1]. Similarly, when n and c are given as 2 and 4, the weight vector and modification interval are $W_{2,4}=(4^0, 4^1)=(1, 4)$ and [-1, 2], respectively. Therefore, when n and c are given, the proposed method generates a minimum modification interval which then creates a minimum modification vector MV. Accordingly, a modulus function k as shown in Equation (4) is formulated. The function is a kind of a mapping which transforms an n-dimensional vector to a value k within the range from zero to (c^n-1), and it can also be observed by a corresponding mapping with possible c^n combinations of pixel changes. Tables 2 and 3 display the corresponding mappings with n=3, c=2 and n=2, c=4, respectively, while k is calculated as follows: Let each row in the Table 2 be a vector, which is denoted as a minimum modification vector $MV=(mv_1, mv_2,\ldots, mv_n)$. Then, the k is formulated as:

$$k=\left(\sum_{i=1}^{n} mv_i \times w_i\right) \bmod c^n \tag{4}$$

Let us consider the case with n and c set as 3 and 2, respectively. $W_{3,2}$ and MV are given as (1, 2, 4) and (0, 1, 1) (see Table 2). Note that k is computed as 6 = (0×1+1×2+1×4) mod 2^3. For the second case, k is 4 as generated by (0×1+0×2+1×4) mod 2^3.

Thus, the new embedding function f_c is used as weighted sum function for modulo c^n for each group, and f_c can be calculated as follows:

$$f_c(g_1, g_2, \ldots, g_n)=\left[\sum_{i=1}^{n}(g_i \cdot w_i)\right] \bmod c^n \tag{5}$$

The modified pixel value vector G_n' can be computed by Equation (6):

$$G_n' = G_n + MV_t, \tag{6}$$

where

$$t=(d-f_c) \bmod c^n, \tag{7}$$

and MV_t is the minimum modification vector for $k = t$ associated with the parameters c and n.

After embedding phase, if the modified pixel value g_i' is less than 0 or more than 255, then the pixel will be in underflow and overflow situations. To solve the underflow fand overflow problems, the new modified pixel value g_i'' is calculated with Equation (8):

$$G_n''(g_1'', g_2'', \ldots, g_n'')=\begin{cases} g_i''=255-Max, & \text{if } g_i''>255 \\ g_i''=0-Min, & \text{if } g_i''<0 \\ g_i''=g_i, & \text{otherwise} \end{cases}, \tag{8}$$

where Max and Min are the maximal and minimal values in the modification interval, respectively. After that, G_n'' is used to embed the secret digit using Equations (5~8). After all secret digits are embedded, the modified pixels are pseudo-randomly re-permuted using the same key α to generate the stego-image I'.The embedding steps of our proposed method are summarized as follows:

Step 1. Calculate f_c value with G_n according to Equation (5).

Step 2. If $d \neq f_c$, compute the value t using Equation (7). Otherwise, go to Step 6.

Step 3. Find the minimum modification vector MV_t, and compute the modified pixel value vector G_n' by Equation (6).

Step 4. Check G_n' whether the underflow or overflow occurs. If G_n' is showing underflow or overflow, go to Step 5. Otherwise, go to Step 6.

Step 5. Calculate G_n'' according to Equation (8), and reset G_n as $G_n = G_n''$.

Step 6. Read next G_n and d, and go to Step 1 until all secret digits are embedded.

Figure 1 displays the flowchart of embedding procedure in our proposed method. We use the following two examples to demonstrate how to embed a secret digital d into a group G_n with and without underflow/overflow in the embedding procedure of proposed method.

Example 1. Assume that n and c are equal to 3 and 2, respectively. Therefore, $W_{3,2}$ is (1, 2, 4). A group G_3 with three cover pixels is given as (200, 203, 208). The to-be-embedded secret bit stream $S=101_2$ is converted into a secret digit d =5 such as d=5 mod 2^3 in a 2^3-ary notational system.

Step 1. Compute f_c value with the G_3 according to Equation (5). In this case, f_c value is 6 because f_c = (200×1+203×2+208×4) mod 2^3=6.

Step 2. Since (d=5)≠(f_c=6), the value of t is set as 7 using Equation (7) (i.e. t =(5-6) mod 2^3=7).

Step 3. For the value t=7, n =3 and c=2, we can derive the corresponding modification vector MV_7 =(1,

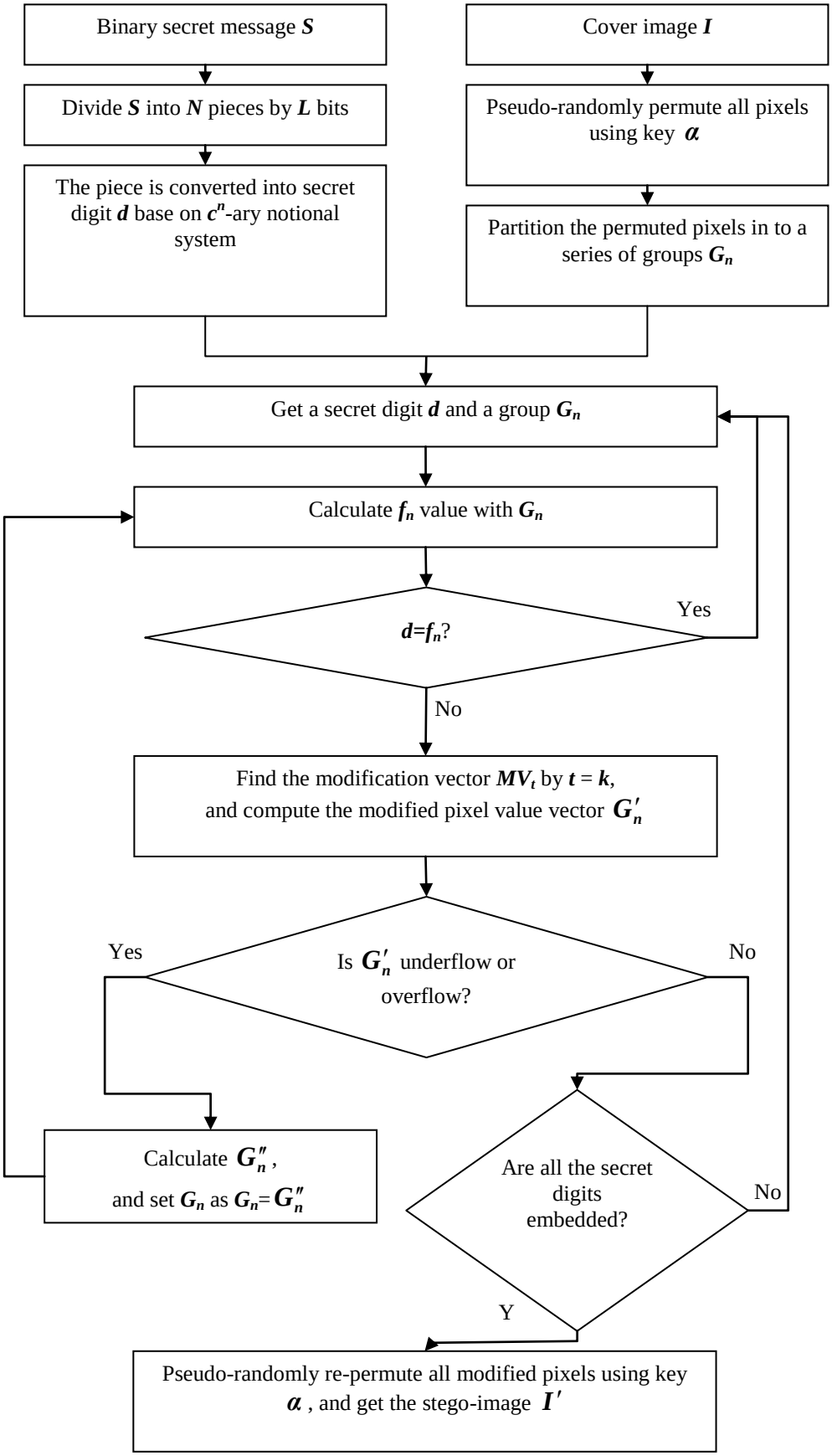


Figure 1: Flowchart of embedding procedure in our proposed method.

1, 1) for k=t=7, could be derived from $t = \left\lfloor \sum_{i=1}^{n} (mv_i \times w_i) \right\rfloor \bmod c^n$ for i=1, 2 and 3. Namely, 7=(1×2^0+1×2^1+1×2^2) mod 2^3. From Table 2, the minimum modification vector MV_7 for the 8th case with (1, 1, 1) is also obtained. Then, according to Equation (6), the modified pixel value vector G_3' turns to (201, 204, 209) (i.e. $G_3' = (200+1, 203+1, 208+1) = (201, 204, 209)$).

Example 2. Let n and c be 2 and 4, respectively. The G_2 is given as (255, 255), and $W_{2,4}$ is given as (1, 4). S=1000_2 is converted into a secret digit d=8 (i.e. d=8 mod 4^2=8) in a 4^2-ary notational system.

Step 1. Compute f_c value with G_2 according to Equation (5). In this case, the value of f_c is 11 because f_c = (255×1+255×4) mod 4^2=11.

Step 2. Since d is smaller than f_c (i.e. 8 < 11), the value of t is 13 from Equation (7) (i.e., 13= (8-11) mod 4^2=(1×4^0+(-1)×4^1) mod 4^2). We also can look up Table 3 and find the corresponding minimum modification vector MV_{13} for k=t=13, which is the two-element vector (1, -1). Thus, according to Equation (6), the modified pixel value vector G_2' is set as (256, 254) by G_2'=(255+1, 255-1)=(256, 254).

Step 3. Since g_1' is overflow now, it should be adjusted such that the secret digit can be embedded. In this case, the maximal value in the modification interval=[-1, 2] is 2; so G_2 is modified into G_2''=(255-2, 255)=(253, 355) using Equation (8). After that, G_2 is set as G_2''.

Step 4. Re-computer f_c value with the G_2. According to Equation (5), f_c equals 9 (i.e. f_c = (253×1+255×4) mod 4^2=9).

Step 5. Since d=8 is smaller than f_c=9, the value of t is 15 from Equation (7) (i.e. t= (8-9) mod 4^2=15).

Step 6. The corresponding modification vector MV_{15} is given as (-1, 0) such that k=t=15=((-1)×4^0+0×4^1) mod 4^2. Therefore, the modified pixel value vector G_2' is modified to (253, 255) (i.e. G_2'=(253-1, 255+0)=(253, 255)).

3.2 Extracting Procedure

Same as the embedding procedure, all the pixels of stego-image I' are pseudo-randomly permuted using key α. After that, the permuted pixels are partitioned into a series of groups, where each group G_n' contains n permuted pixels such as $G_n' = (g_1', g_2', \ldots, g_n')$. The secret digit d can be extracted from G_n' by extracted function f_c'. The extracted function f_c' is defined as follows:

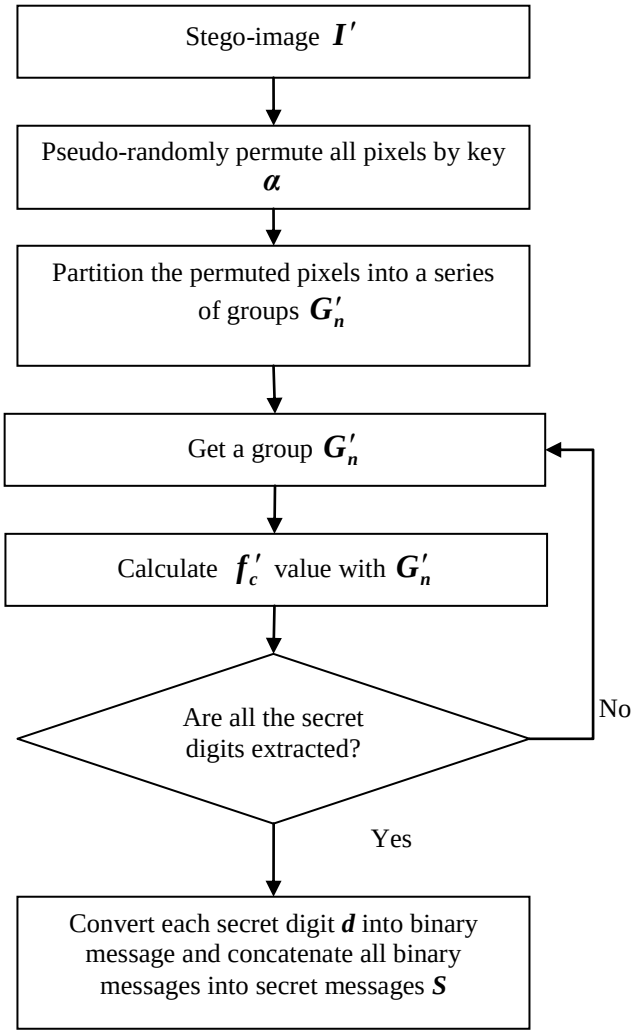


Figure 2: Flowchart of extracting procedure in our proposed method

The value of f_c' is equal to the secret digit d. While all secret digits are being extracted, all the binary messages are concatenated to recover the secret message S. Figure 2 shows the flowchart of extracting procedure in our proposed method.

Example 3 illustrates how to extract a secret digit d from a group G_n' below.

Example 3. Assume that n and c equal to 3 and 2, respectively. G_3' is given as (201, 204, 209), and $W_{3,2}$ is (1, 2, 4). According to Equation (9), the secret digit d is calculated as d= f_c' =(201×1+204×2+209×4) mod 2^3=5, and the secret message S=101_2 is in its binary representation.

4 Experimental Result

The aim of this section is to investigate the performance of the proposed method and to compare it with Wang *et al.*'s method [10], EMD-2 method [4], Lee *et al.*'s method [5],

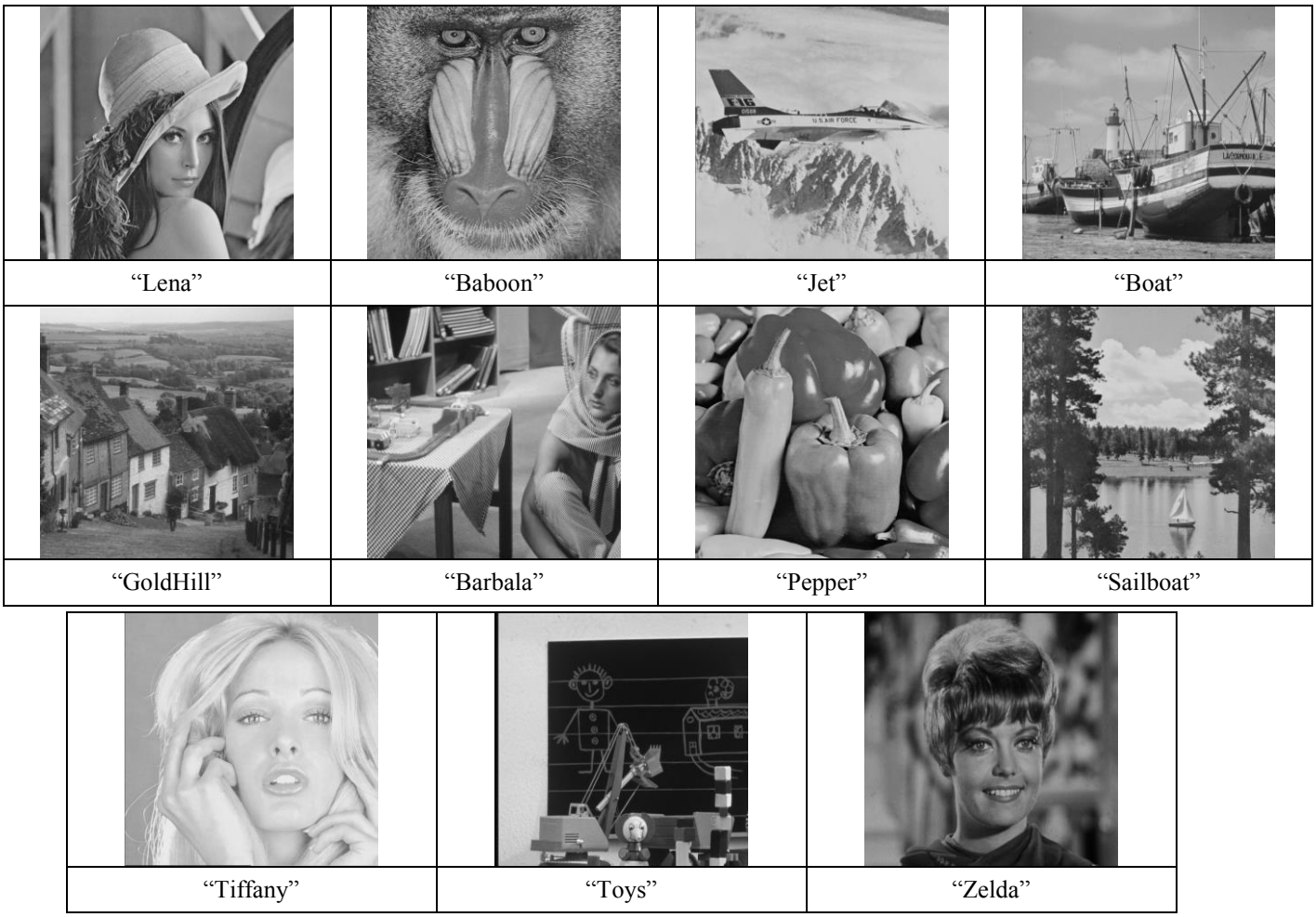


Figure 3: The test images

Lee *et al.*'s method [6], and EMD [12]. In the experiments, the gray-level images with size of 512×512 "Lena", "Baboon", "Jet", "Boat", "GoldHill", "Barbala", "Pepper", "Sailboat", "Tiffany", "Toys", and "Zelda" are used as test images shown in Figure 3.

The peak signal to noise ratio (***PSNR***) [4] is used to evaluate the image quality. The ***PSNR*** is defined as follows:

$$\boldsymbol{PSNR} = 10 \times \log_{10} \frac{255}{\boldsymbol{MSE}} \text{dB}, \tag{10}$$

MSE is the mean squared error and is computed as follows:

$$\boldsymbol{MSE} = \frac{1}{\boldsymbol{H} \times \boldsymbol{W}} \times \sum_{i=1}^{\boldsymbol{H}} \sum_{j=1}^{\boldsymbol{W}} (\boldsymbol{x}_{ij} - \bar{\boldsymbol{x}}_{ij})^2, \tag{11}$$

where $\boldsymbol{H}$ and $\boldsymbol{W}$ are the height and width of image, respectively; $\boldsymbol{x}_{ij}$ is the original pixel value at coordinate $(\boldsymbol{i}, \boldsymbol{j})$, and $\bar{\boldsymbol{x}}_{ij}$ is the modified pixel value at coordinate $(\boldsymbol{i}, \boldsymbol{j})$. The embedding rate (***R***) is employed to calculate the number of secret bits carried by one pixel. The embedding rate is formulated as

$$\boldsymbol{R} = \frac{\text{The number of embedding bits}}{\boldsymbol{H} \times \boldsymbol{W}} (\text{bpp}). \tag{12}$$

4.1 Parameter Selection

The first experiment investigates two parameters $\boldsymbol{n}$ and $\boldsymbol{c}$ to observe how the parameters affect embedding performance. Table 4 presents the embedding capacity and the image quality of proposed method with different $\boldsymbol{n}$ and $\boldsymbol{c}$ for image "Lena." By Table 4, results showed that only the parameter $\boldsymbol{c}$ controls the embedding capacity and the image quality. When $\boldsymbol{c}$ is increasing, the embedded rate will increase and the ***PSNR*** of stego-image will decrease. The theoretical embedding rate $\boldsymbol{R}$ of the proposed method is presented as the following equation, and it shows that the embedding capacity depends on the parameter $\boldsymbol{c}$.

$$\boldsymbol{R} = (\log_2 \boldsymbol{c}^{\boldsymbol{n}}) / \boldsymbol{n} = [\boldsymbol{n} \cdot (\log_2 \boldsymbol{c})] / \boldsymbol{n} = \log_2 \boldsymbol{c}. \tag{13}$$

Accordingly, the embedding capacity of our proposed method determined by the parameter $\boldsymbol{c}$ is scalable. The adjusted parameter $\boldsymbol{c}$ provides the proposed method the flexibility in both embedding capacity and image visual quality.

Table 4: The ***PSNR*** and ***R*** of the proposed method with different ***c*** and ***n*** for "Lena"

n \ ***c***	***c***=2		***c***=3		***c***=4		***c***=5		***c***=6	
	PSNR (dB)	***R*** (bpp)	***PSNR*** (dB)	***R*** (bpp)	***PSNR*** (dB)	***R*** (bpp)	***PSNR*** (dB)	***R*** (bpp)	***PSNR*** (dB)	***R*** (bpp)
n=2	51.14	1.00	49.90	1.58	46.38	2.00	45.13	2.32	43.14	2.58
n=3	51.13	1.00	49.90	1.58	46.36	2.00	45.12	2.32	43.10	2.58
n=4	51.14	1.00	49.88	1.58	46.38	2.00	45.12	2.32	43.12	2.58
n=5	51.15	1.00	49.89	1.58	46.38	2.00	45.11	2.32	43.13	2.58
n=6	51.14	1.00	49.89	1.58	46.37	2.00	45.12	2.32	43.12	2.58
n=7	51.14	1.00	49.89	1.58	46.36	2.00	45.11	2.32	43.13	2.58

Table 5: The ***PSNR*** and ***R*** of the proposed method with different ***c*** and ***n***=2

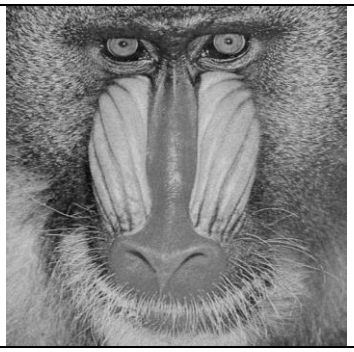
Image \ ***c***	***c***=2		***c***=3		***c***=4		***c***=27		***c***=28	
	PSNR (dB)	***R*** (bpp)	***PSNR*** (dB)	***R*** (bpp)	***PSNR*** (dB)	***R*** (bpp)	***PSNR*** (dB)	***R*** (bpp)	***PSNR*** (dB)	***R*** (bpp)
Lena	51.14	1.00	49.88	1.58	46.38	2.00	30.29	4.75	29.97	4.81
Baboon	51.14	1.00	49.89	1.58	46.35	2.00	30.30	4.75	29.98	4.81
Jet	51.14	1.00	49.89	1.58	46.36	2.00	30.32	4.75	29.99	4.81
GoldHill	51.14	1.00	49.89	1.58	46.36	2.00	30.28	4.75	29.96	4.81
Boat	51.14	1.00	49.90	1.58	46.36	2.00	30.30	4.75	29.96	4.81
Barbala	51.15	1.00	49.89	1.58	46.37	2.00	30.30	4.75	29.99	4.81
Pepper	51.13	1.00	49.89	1.58	46.36	2.00	30.31	4.75	29.98	4.81
Sailboat	51.14	1.00	49.89	1.58	46.35	2.00	30.30	4.75	29.98	4.81
Tiffany	51.14	1.00	49.89	1.58	46.35	2.00	30.20	4.75	29.98	4.81
Toys	51.13	1.00	49.88	1.58	46.36	2.00	30.31	4.75	29.98	4.81
Zelda	51.14	1.00	49.88	1.58	46.36	2.00	30.30	4.75	29.98	4.81

"Lena"		"Baboon"		"Jet"	
PSNR	51.14 dB	***PSNR***	51.14 dB	***PSNR***	51.14 dB
R	1 bpp	***R***	1 bpp	***R***	1 bpp
Capacity	262144 bits	Capacity	262144 bits	Capacity	262144 bits

"Boat"		"GoldHill"	
PSNR	51.14 dB	***PSNR***	51.14 dB
R	1 bpp	***R***	1 bpp
Capacity	262144 bits	Capacity	262144 bits

Figure 4: The ***PSNR***, ***R***, and embedding capacity of the proposed method(***n***=2, ***c***=2).

"Lena"	
PSNR	30.29 dB
R	4.75 bpp
Capacity	1245184 bits



"Baboon"	
PSNR	30.30 dB
R	4.75 bpp
Capacity	1245184 bits

"Jet"	
PSNR	30.32 dB
R	4.75 bpp
Capacity	1245184 bits

"Boat"	
PSNR	30.30 dB
R	4.75 bpp
Capacity	1245184 bits

"GoldHill"	
PSNR	30.28 dB
R	4.75 bpp
Capacity	1245184 bits

Figure 5: The ***PSNR***, ***R***, and embedding capacity of the proposed method (***n***=2, ***c***=27).

4.2 Embedding Rate versus Image Distortion in the Proposed Method

Since the parameter ***c*** dominates the embedding rate, the second experiment will focus on analyzing the embedded rate of the proposed method. Table 5 shows the ***PSNR*** and ***R*** of the proposed method with different ***c*** when ***n***=2 for the test images. Table 5 indicates that the proposed method can provide average values of ***PSNR*** at 51.14 dB~30.30 dB and average embedding rates at 1.00~4.75 bpp with different ***c***=2~27. However, it is difficult for human eyes to discriminate the difference between the original image and the stego-image when ***PSNR*** exceeds 30 dB. Therefore, the image quality of the proposed method with ***c***=27 is satisfied. In addition, the proposed method provides various image qualities and embedding capacities by adjusting ***c***, according to user application requirements. Figures 4 and 5 display the ***PSNR***, ***R***, and embedding capacity of the proposed method with ***n***=2, ***c***=2, and ***n***=2, ***c***=27 for a subset of test images, respectively with smooth images "Lena" and "Jet" and some are complex like "Baboon," "Boat", and "GoldHill."

4.3 Embedding Capacity versus Image Distortion for Related Methods

The third experiment was designed to compare the performance of the proposed method with those of the other image hiding methods based on the EMD method with similar conditions, including Wang *et al.*'s method [10], Lee *et al.*'s method [5], Lee *et al.*'s method [6], EMD-2 method [4], and EMD [12]. Table 6 shows the performances of the proposed method and the compared methods for the test images. In Wang *et al.*'s method, ***K*** is the parameter to control the capacity and image quality of stego-image. In this experiment, the average values of ***PSNR*** and ***R*** in Wang *et al.*'s method are 46.89 dB and 1.13 bpp while ***K*** is 1, respectively. In addition, when ***K*** is set to 70, Wang *et al.*'s method can provide an average ***PSNR*** value of 45.16 dB and embedding rate ***R*** of 1.99 bpp. In Lee *et al.*'s method, it keeps $(16 - p_m)$ most significant bits of a pixel-pair unchanged while altering p_m least significant bits to indicate the virtual modifications on a ***m***-dimensional pseudo-random vectors for carrying the secret data, where $m = 2^{p_m - 1} - 1$. The average values of ***PSNR*** and ***R*** in Lee *et al.*'s method [5] are 46.89 dB and 1.13 bpp when p_m is 4 and ***m*** is 7, respectively. However, Table 4 indicates that the ***PSNR*** and ***R*** of the proposed method are superior to that of Wang *et al.*'s [10] and Lee *et al.*'s methods [5]. Under similar conditions, the proposed method supplies the same ***PSNR*** and ***R*** with the EMD-2 method. However, the maximal embedding rate of the

Table 6: The performances of the proposed method and the compared methods

Method	Parameter	Measure	Lena	Baboon	Jet	GoldHill	Boat
Proposed method	n=2, c=3	***PSNR***(dB)	49.88	49.89	49.89	49.89	49.90
		R(bpp)	1.58	1.58	1.58	1.58	1.58
	n=2, c=4	***PSNR***(dB)	46.38	46.35	46.36	46.36	46.36
		R(bpp)	2.00	2.00	2.00	2.00	2.00
Wang *et al.*'s method[10]	K=1	***PSNR***(dB)	46.90	46.89	46.89	46.89	46.88
		R(bpp)	1.13	1.13	1.13	1.13	1.13
	K=70	***PSNR***(dB)	45.16	45.15	45.16	45.15	45.16
		R(bpp)	1.99	1.99	1.99	1.99	1.99
Lee *et al.*'s method[5]	m=7, p_m=4	***PSNR***(dB)	44.31	44.28	44.45	44.28	44.33
		R(bpp)	1.95	1.95	1.95	1.95	1.95
Lee *et al.*'s method[6]	n=4	***PSNR***(dB)	44.17	44.17	44.14	44.17	44.16
		R(bpp)	2.00	2.00	2.00	2.00	2.00
EMD-2 method[4]	n=2	***PSNR***(dB)	49.89	49.89	49.88	49.89	49.90
		R(bpp)	1.58	1.58	1.58	1.58	1.58
EMD method[12]	n=2	***PSNR***(dB)	52.11	52.13	52.12	52.09	52.08
		R(bpp)	1.16	1.16	1.16	1.16	1.16
LSB method[9]		***PSNR***(dB)	44.15	44.17	44.15	44.15	44.13
		R(bpp)	2	2	2	2	2

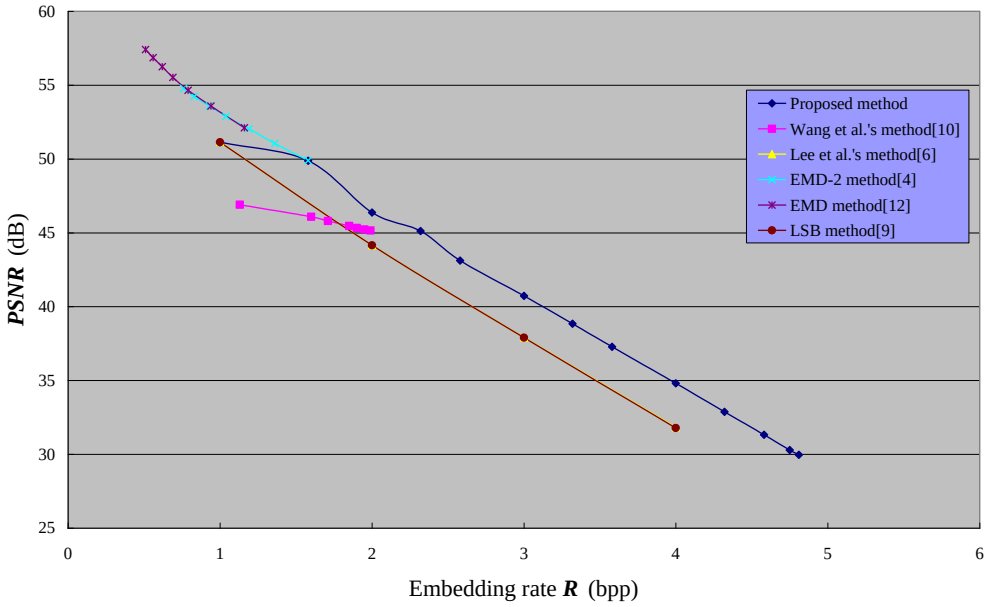


Figure 6: The performances of the proposed method and the compared methods for images "Lena"

proposed method (i.e. 4.75 bpp) is more than that of the EMD-2 method, which is 3.32 bpp and can be referred by [4]. Lee *et al.*'s method [6] develops a hiding method based on modulus function which provides the same performance as the LSB replacement method when compared using two criteria, namely visual quality and embedding capacity. Their average PSNR was 31.847 dB when the embedding rate was 4 bits for each gray-level pixel. Our proposed method is superior compared to the Lee *et al.*'s [6] and LSB replacement methods [9] with respect to visual quality and embedding capacity. From Table 6, when the embedding rate reaches 2 bpp, the PSNR value can be improved about 2.19 dB, which is 46.35-44.16, for the same capacity.

Figure 6 demonstrates the relationship between the ***PSNR*** and ***R*** for image "Lena" obtained by Wang *et al.*'s [10], Lee *et al.*'s [6], EMD-2 [4], EMD [12], LSB [9], and the proposed methods. The maximal embedding rate of Wang *et al.*'s method is close to 2 bpp and provides a ***PSNR*** value over 45 dB; Lee *et al.*'s and LSB methods provide a ***PSNR*** value of 31.78 dB when the maximal embedding rate is at 4 bpp. Although the EMD and EMD-2 methods generate a better image quality (i.e. over 50 dB), the maximal embedding rates of both methods can only achieve 1.16 and 1.58 bpp, respectively, which are not that high. Overall, the proposed method has shown the ability to provide the maximal embedding rate of 4.75 bpp and a good image quality than compared methods.

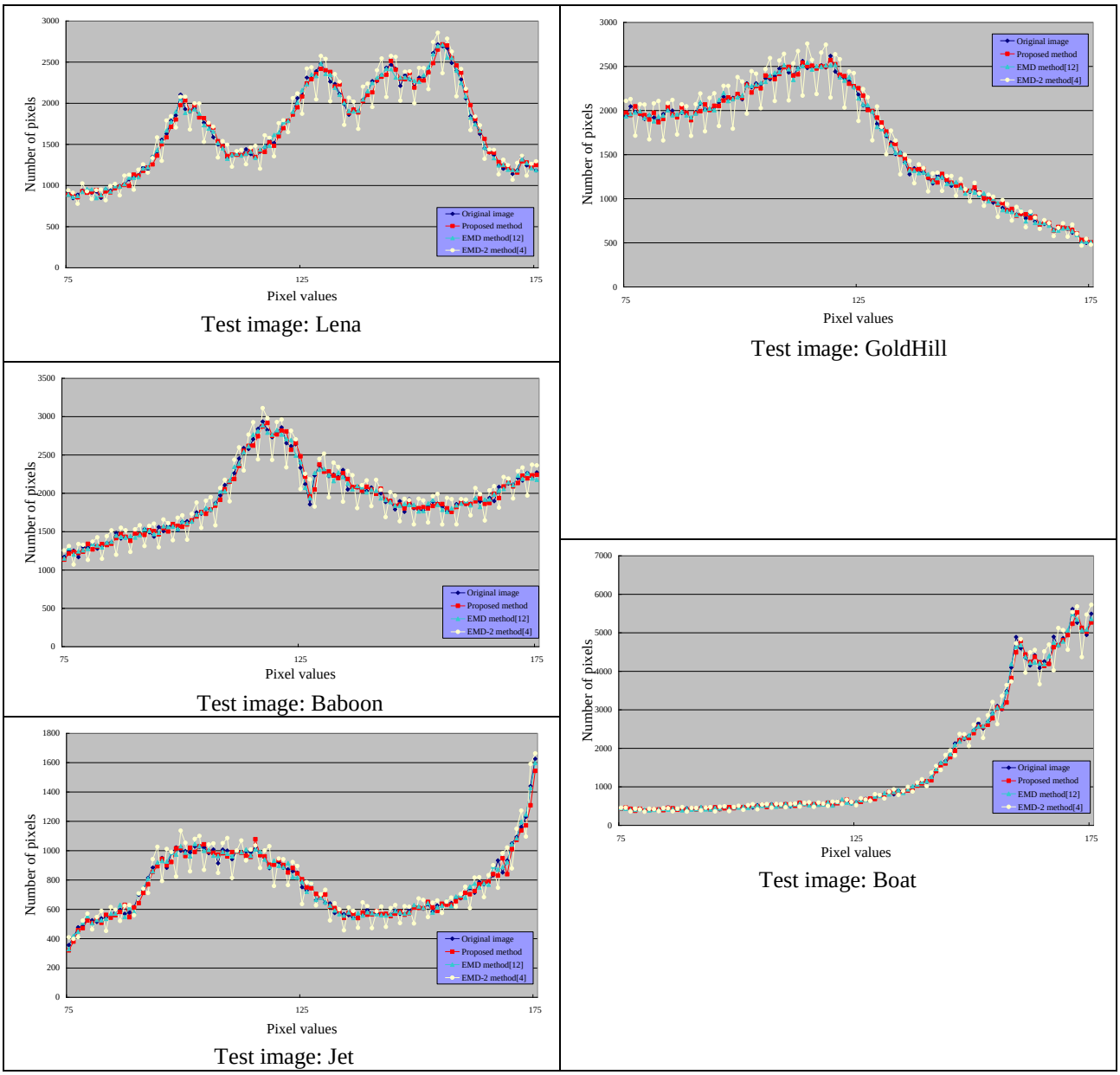


Figure 7: The histograms of the original test images and stego-images using the proposed, EDM, and EDM-2 methods

4.4 Imperceptibility Analysis for EMD-based Method

In the last experiment, in order to establish a fair comparison among the proposed method, EMD method, and EMD-2 method, we have embedded the same secret capacity into the test images "Lena," "Baboon," "Jet," "GoldHill", and "Boat." Figure 7 displays the pixel histograms of both original images and stego-images obtained by the proposed method, EMD method, and EMD-2 method in the test image. These histograms represent the frequency of pixel values between pixel values from 75 to 175. Also, comparing with our proposed method, EMD method and EMD-2 method, the curves of histogram in the proposed method and EMD method are both closer to that of the original image; this means that the less the modification made to the original image, the less the difference between the stego-image and original image. Compared with the original image histogram, minute differences are inevitable; therefore, the steganographic attack (i.e. statistic analysis) may be hardly detected.

5 Conclusion

The EMD method allows at most one pixel is modified in a group. Therefore, the maximal embedding rate of EMD method is 1.16 bpp when n is 2. To improve the embedding capacity of EMD method, a new high payload adjustment image hiding method is proposed in this paper. In the proposed method, n pixels in a group can be modified at most and each pixel has c different ways of modification. According to experimental results, the minimal and maximal embedding rates of our proposed method are 1 bpp and 4.75 bpp respectively with *PSNR* over 30 dB. As

the experiment results indicated, our proposed method provides a higher embedding rate and a better image quality of stego-image than the compared methods. Moreover, the proposed method can achieve certain degree of security with low computation. The users are granted the ability to adjust the embedding rate of the proposed method by using different *c* for various applications requirements.

References

[1] C. K. Chan and L. M. Cheng, "Hiding data in image by simple lsb substitution," *Pattern Recognition*, vol. 37, no. 3, pp. 469-474, 2004.

[2] C. C. Chang, C. C. Lin, C. S. Tseng, and W. L. Tai, "Reversible hiding in dct-based compressed images," *Information Sciences*, vol. 177, no. 13, pp. 2768-2786, 2007.

[3] S. Dumitrescu, X. Wu, and Z. Wang, "Detection of LSB steganography via sample pair analysis," *IEEE Transactions on Signal Processing*, vol. 51, no. 7, pp. 1995-2007, 2003.

[4] H. J. Kim, C. Kim, Y. Choi, S. Wang, and X. Zhang, "Improved modification direction method," *Computers & Mathematics with Applications*, vol. 60, no. 2, pp. 319-325, 2010.

[5] C. F. Lee, C. C. Chang, and K. H. Wang, "An improvement of EMD embedding method for large payloads by pixel segmentation strategy," *Image and Vision Computing*, vol. 26, no. 12, pp. 1670-1676, 2008.

[6] C. F. Lee and H. L. Chen, "A novel data hiding scheme based on modulus function," *Journal of Systems and Software*, vol. 83, no. 5, pp. 832-843, 2010.

[7] J. Mielikainen, "LSB matching revisited," *IEEE Signal Processing Letters*, vol. 13, no. 5, pp. 285-287, 2006.

[8] C. C. Thien and J. C. Lin, "A simple and high-hiding capacity method for hiding digit-by-digit data in images based on modulus function," *Pattern Recognition*, vol. 36, no. 12, pp. 2875-2881, 2003.

[9] L. F. Turner, "Digital data security system," *Patent IPN*, WO 89/08915, 1989.

[10] Z. H. Wang, T. D. Kieu, C. C. Chang, and M. C. Li, "A novel information concealing method based on exploiting modification direction," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 1, no. 1, pp. 1-9, 2010.

[11] A. Westfeld and A. Pfitamann, "Attacks on steganographic systems," *Lecture Notes in Computer Science*, vol. 1768, pp. 61-76, 1999.

[12] X. Zhang and S. Wang, "Efficient steganographic embedding by exploiting modification direction," *IEEE Communications Letters*, vol. 10, no. 11, pp. 781-783, 2006.

Chin-Feng Lee received Ph.D. degree in Computer Science and Information Engineering in 1998 from National Chung Cheng University in Taiwan. She is currently a professor in Department of Information Management at Chaoyang University of Technology, at Taichung. Her research interests include database design, data mining, image processing, and information hiding.

Chin-Chen Chang received his Ph.D. degree in computer engineering from National Chiao Tung University. He's first degree is Bachelor of Science in Applied Mathematics and master degree is Master of Science in computer and decision sciences. Both were awarded in National Tsing Hua University. Dr. Chang served in National Chung Cheng University from 1989 to 2005. His current title is Chair Professor in Department of Information Engineering and Computer Science, Feng Chia University, from Feb. 2005. Professor Chang was an associate professor in Chiao Tung University, chair professor in National Chung Cheng University. He had also been Visiting Researcher and Visiting Scientist to Tokyo University and Kyoto University, Japan. During his service in Chung Cheng, Professor Chang served as Chairman of the Institute of Computer Science and Information Engineering, Dean of College of Engineering, Provost and then Acting President of Chung Cheng University and Director of Advisory Office in Ministry of Education, Taiwan. He is currently a Fellow of IEEE and a Fellow of IEE, UK. And since his early years of career development, he consecutively won Outstanding Youth Award of the R. O. C., Outstanding Talent in Information Sciences of the R. O. C., AceR Dragon Award of the Ten Most Outstanding Talents, Outstanding Scholar Award of the R. O. C., Outstanding Engineering Professor Award of the R. O. C., Chung-Shan Academic Publication Awards, Distinguished Research Awards of National Science Council of the R. O. C., Outstanding Scholarly Contribution Award of the International Institute for Advanced Studies in Systems Research and Cybernetics, Top Fifteen Scholars in Systems and Software Engineering of the Journal of Systems and Software, and so on. His current research interests include database design, computer cryptography, image compression and data structures.

Pei-Yan Pai received the B.S. degree in Department of Information Management from National Taichung Institute of Technology, Taichung, Taiwan, in 2002. He received the M.S. degree in Department of Information Management from Chao Yang University of Technology, Taichung, Taiwan, in 2007. In 2011, he obtained Ph.D. degree from the Department of Computer Science of National Tsing Hua University. His research interests include medical image analysis, pattern recognition and multimedia applications.

Chia-Ming Liu received the B.S. degree in 2009 and M.S degree in 2011 from Department of Information Engineering Computer Science of Feng Chia University, Taichung, Taiwan. His research interests include image processing and digital watermarking.

A Practical Forward-Secure Public-Key Encryption Scheme with Untrusted Update

Xiujie Zhang[1], Chunxiang Xu[2]
(Corresponding author: Xiujie Zhang)
School of Computer Engineering, WeiFang University[1]
No.5147, Dongfeng Street, Weifang 261061, China
(Email: 2008xiujie@163.com)
School of Computer Science and Engineering, University of Electronic Science and Technology of China[2]
No.2006, Xiyuan Avenue, West Hi-tech Zone, Chengdu 611731, China
(Received Feb. 9, 2014; revised and accepted Nov. 15 & Dec. 26, 2014)

Abstract

Forward-secure public-key schemes with untrusted update are a variant of forward-secure public-key schemes such that the private key can be updated in the encrypted version. In this paper, we firstly describe forward-secure public-key scheme with untrusted update and security definition. We put forward a generic construction and prove its security in standard model. In our construction, the forward security is realized by applying a binary tree encryption and the update security is achieved by using a symmetric encryption. Meanwhile, we proved our scheme in the standard model. Our generic construction is very practical compare with previous works.

Keywords: Composite order bilinear groups, forward security, key exposure, provable security, public key encryption, untrusted update

1 Introduction

Traditionally, a forward secure cryptography assumes that one to mitigate the damage caused by exposure of secret keys. In a forward-secure public-key system, private keys are updated at regular time periods; an exposure of the private key SK_i corresponding to a given time period i does not enable an adversary to compromise the security of the scheme for any time period prior to i. For the case of signature, forward security guarantees that past signatures are protected even if the secret key of the current time period exposed. There are a number of known forward-secure signature schemes [1, 2, 10, 11, 12, 17], and forward-secure symmetric-key setting has also been studied [3]. For the case of encryption, forward security ensures that even after an adversary having the private key SK_i(for some i), the adversary can obviously compromise future usage of the scheme but messages encrypted for past periods remains secret. In Eurocrypt 2003, Canetti et al. [8] introduced forward security to Public-Key Encryption(fsPKE) scheme. Subsequently, the work of [4, 13, 20, 21] construct forward-secure (Hierarchical) IBE scheme based on the study of a number of Identity-Based Encryption(IBE) schemes [5, 19]. Recently, Nieto et al. [18] put forth a forward-secure Hierarchical Predicate Encryption(HPE) which ensures forward security for plaintexts and for attributes that are hidden in HPE ciphertexts.

However, Boyen et al. [7] showed that forward security is not applied to many software environments such as GNU-PG or S/MIME. A Forward-Secure Signature scheme with Untrusted Update(FSS-UU) was first proposed by Boyen et al. [7] in which the signing key is additionally shielded by a second factor and key update can be performed on an encrypted version of the signing key. And they left the open problem of adding untrusted update to other existing fsPKE scheme. Sequently, the work of [15] solved the open problem of [7] and presented a very efficient generic construction from any FSS scheme by expanding MMM construction [17]. Specializing for the case of encryption, which is the focus of this work, forward security with untrusted update guarantees that even if an adversary learns the encrypted private key $EncSK_i$(for some i), he can not decrypt correctly any ciphertext. The work of [16] extended the untrusted update model to fsPKE scheme. Libert el al. gave the resulting fsPKE with untrusted update(uufsPKE) scheme with update security and forward security in the chosen-plaintext setting. In their scheme, their method needs both a fsPKE and a traditional PKE which is not practical and leads to double-cost of size of ciphertext,public key and secret key. Therefore, how to construct more efficient generic uufsPKE scheme is worth researching.

This paper shows that forward-secure public-key encryption scheme with untrusted update is easier to derive from Symmetric Encryption and semantically secure Binary Tree Encryption. We firstly give the formal def-

inition of forward-secure public-key encryption scheme with untrusted update and security definition. We propose a generic transformation of forward-secure public-key encryption scheme with untrusted update which supports both forward security and update security. In our generic transformation, the forward security can be achieved by applying binary tree encryption scheme and the update security can be obtained using symmetric encryption scheme. Our scheme does not rely random oracle. Meanwhile, we present a concrete construction based on Binary Tree Encryption which we propose and prove its security in standard model. To the author's knowledge, this is the first provable efficient uufsPKE scheme which realizes constant ciphertext size and decreases the size of private key compared with Libert [16].

The rest of this paper is organized as follows. In Section 2, we provide some necessary preliminaries about binary tree encryption and symmetric encryption. In Section 3, we define uufsPKE scheme and its security notion formally. Our generic construction and its security proof will be proposed in Section 4. Finally, we conclude in Section 5.

2 Preliminaries

For a positive integer T, $[T]$ denotes $\{1, \cdots, T\}$. Let N be a positive integer and $Z_N = \{1, \ldots, N-1\}$. We let x be chosen uniformly from Z_N denote by $x \xleftarrow{\$} Z_N$. By $negl(\lambda)$, we denote a negligible function of parameter λ. PPT stands for probabilistic polynomial time. Algorithm A with input x and random tape r is denoted by $A(x;r)$.In this section, we give the definitions of primitives which our scheme is based on.

2.1 Composite Order Bilinear Groups

We will use composite order bilinear groups introduced in [6]. In the group generator algorithm $\mathcal{G}$ takes as input a security parameter 1^λ and outputs the description of a bilinear group G of composite order $N = p_1p_2p_3$ where p_1, p_2, p_3 are three prime numbers of magnitude $\Theta(2^\lambda)$. We let G_1, G_2, G_3 denote these subgroups. We assume that the generator algorithm outputs the values of p_1, p_2, p_3 and g_i is a generator of subgroup G_i.

For $i, j = \{1, 2, 3\}$, we also let G_{ij} denote the subgroup of order p_ip_j. If $h_i \in G_i$ and $h_j \in G_j$ for $i \neq j$, we have that $e(h_i, h_j) = 1_{G_T}$.

2.2 Hardness Assumptions

We define the following three assumptions in composite order bilinear groups.

Assumption 1. *The challenger runs $\mathcal{G}(1^\lambda)$ and gives to the adversary $\mathcal{A}$ the tuple $D^1 = (N, G, G_T, e, g_1, g_3)$. Then the challenger flips a random coin $\nu \in_R \{0,1\}$ and picks random $(z, \xi) \xleftarrow{\$} Z_{p_1} \times Z_{p_2}$. He computes $T_0^1 = g_1^z$ and $T_1^1 = g_1^z g_2^\xi$, sends T_ν^1 to $\mathcal{A}$. In the end, $\mathcal{A}$ outputs a bit ν', and succeeds if $\nu' = \nu$.*

Assumption 2. *The challenger runs $\mathcal{G}(1^\lambda)$ and picks random exponents $(z, \upsilon, \mu, \tau) \xleftarrow{\$} Z_{p_1} \times Z_{p_2} \times Z_{p_2} \times Z_{p_3}$. He gives to $\mathcal{A}$ the tuple $D^2 = (N, G, G_T, e, g_1, g_3, g_1^z g_2^\upsilon, g_2^\mu g_3^\tau)$. Then he flips a random coin $\nu \leftarrow \{0,1\}$ and picks random $(\gamma, \xi, \kappa) \xleftarrow{\$} Z_{p_1} \times Z_{p_2} \times Z_{p_3}$. He computes $T_0^2 = g_1^\gamma g_3^\kappa$ and $T_1^2 = g_1^\gamma g_2^\xi g_3^\kappa$, sends T_ν^2 to $\mathcal{A}$. In the end, $\mathcal{A}$ outputs a bit ν', and succeeds if $\nu' = \nu$.*

Assumption 3. *The challenger runs $\mathcal{G}(1^\lambda)$ and picks random exponents $(\alpha, s, \xi, \mu, \varpi) \xleftarrow{\$} Z_{p_1} \times Z_{p_1} \times Z_{p_2} \times Z_{p_2} \times Z_{p_2}$. He gives to $\mathcal{A}$ the tuple $D^3 = (N, G, G_T, e, g_1, g_3, g_1^\alpha g_2^\xi, g_1^s g_2^\mu, g_2^\varpi)$. Then he flips a random coin $\nu \leftarrow \{0,1\}$ and picks a random $w \xleftarrow{\$} Z_{p_1}$. He computes $T_0^3 = e(g_1, g_1)^{\alpha s}$ and $T_1^3 = e(g_1, g_1)^{\alpha w}$, sends T_ν^3 to $\mathcal{A}$. In the end, $\mathcal{A}$ outputs a bit ν', and succeeds if $\nu' = \nu$.*

Assumption 4. *The advantage of any PPT adversary $\mathcal{A}$ in Assumption i where $i \in \{1,2,3\}$ is $Adv^i_{\mathcal{G},\mathcal{A}} = \frac{1}{2}(Pr[\mathcal{A}(D^i, T_0^i) = 0] - Pr[\mathcal{A}(D^i, T_1^i) = 0])$. We say that $\mathcal{G}$ satisfies Assumption i for all PPT algorithms $\mathcal{A}$, $Adv^i_{\mathcal{G},\mathcal{A}} \leq negl(n)$.*

2.3 Binary Tree Encryption Scheme

Binary tree encryption (BTE) was introduced by Canetti et al. [8]. In a BTE scheme, each node w has two children (labeled $w0$ and $w1$) while in a HIBE scheme, each node has arbitrarily-many children labeled with arbitrary strings. We review the relevant definitions of BTE scheme due to [8].

Definition 1. *A binary tree encryption scheme BTE is a 4-tuple of PPT algorithms (Gen,Der,Enc,Dec) such that:*

$Gen(\lambda, \ell) \rightarrow (sk_\varepsilon, pk)$. The randomized algorithm inputs a security parameter k and the maximum tree depth ℓ. It outputs some system parameters pk along with a master (root) secret key sk_ε. (We assume that λ and ℓ are implicit in pk and all secret keys.)

$Der(w, sk_w) \rightarrow (sk_{w0}, sk_{w1})$. The key derivation algorithm takes as input the name of a node $w \in \{0,1\}^{\leq \ell}$ and its associated secret key sk_w. It outputs secret keys sk_{w0}, sk_{w1} for the two children of w.

$Enc(w, pk, M) \rightarrow CT$. It takes as input pk, the name of a node $w \in \{0,1\}^{\leq \ell}$, and a message M, and returns a ciphertext CT.

$Dec(w, sk_w, CT) \rightarrow M$. The deterministic algorithm takes as input $w \in \{0,1\}^{\leq \ell}$, its associated secret key sk_w, and a ciphertext CT. It returns a message M or the distinguished symbol $\perp$.

We require that for all (pk, sk_w) output by Gen, any $w \in \{0,1\}^{\leq \ell}$ and any correctly-generated secret key sk_w for this node, any message M, and all CT output by $Enc_{pk}(w, M)$ we have $Dec_{sk_w}(w, Enc_{pk}(w, M)) = M$.

Definition 2. *A binary tree encryption scheme BTE is secure against selective-node, chosen-plaintext attacks (SN-CPA) if for all polynomially-bounded functions $\ell(\cdot)$ the advantage of any PPT adversary A in the following game cpa-bte is negligible in the security parameter k.*

Setup Phase. *$\mathcal{A}(k, \ell(k))$ outputs a node label $w^* \in \{0,1\}^{\leq \ell(k)}$. Algorithm $Gen(1^\lambda, 1^\ell)$ outputs (pk, sk_ε).*

Phase 1. *In addition, algorithm $Der()$ is run to generate the secret keys of all the nodes on the path from the root to w^* (we denote this path by P). The adversary is given pk and the secret keys sk_w for all nodes w of the following form:*

- *$w = w'b$,where $w'b$ is a prefix of w^* and $b \in \{0,1\}$(w is a sibling of some node in P);*
- *$w = w^*0$ and $w = w^*1$, if $|w^*| < \ell$(w is a child of w^*). Note that it allows the adversary to compute $sk_{w'}$ for any node $w' \in \{0,1\}^{\leq \ell}$ that is not a prefix of w^*.*

Challenge Phase. *The adversary generates a request challenge (M_0, M_1). A random bit b is selected and the adversary is given $C^* = Enc(pk, w^*, M_b)$.*

Guess Phase. *Eventually, $\mathcal{A}$ outputs a guess $b' \xleftarrow{\$} \{0,1\}$.*

2.4 Symmetric Encryption

A symmetric encryption (SE) scheme is a tuple of PPT algorithms $\Pi = (Gen, Enc, Dec)$ such that:

1) The key-generation algorithm **Gen** is a randomized algorithm. Takes as input the security parameter 1^n and outputs a key k; we write this as $k \leftarrow Gen(1^n)$. Without loss of generality, we assume that any key k output by $Gen(1^n)$ satisfies $n \leq |k|$.

2) The encryption algorithm **Enc** may be randomized. Takes as input a key k and a plaintext message $m \in \{0,1\}^*$, and outputs a ciphertext c, write by $c \leftarrow Enc_k(m)$.

3) The decryption algorithm **Dec** is deterministic. Takes as input a key k, a ciphertext c and outputs a message m. That is, $m := Dec_k(c)$.

It is required that for every n, every key k output by $Gen(1^n)$, and every $m \in \{0,1\}^*$, it holds that $Deck(Enck(m)) = m$.

We present the formal security definition against chosen-ciphertext attacks where the adversary has access to a decryption oracle and the encryption oracle. Consider the following game c/a for any private-key encryption scheme $\Pi = (Gen, Enc, Dec)$, adversary $\mathcal{A}$ and the security parameter n.

1) A random key k is generated by running $Gen(1^n)$.

2) The adversary $\mathcal{A}$ is given input 1^n and oracle to $Enc_k(\cdot)$ and $Dec_k(\cdot)$. It outputs a pair of messages m_0, m_1 of the same length.

3) A random bit $b \leftarrow \{0,1\}$ is chosen, and then a ciphertext $c \leftarrow Enc_k(m_b)$ is computed and given to $\mathcal{A}$. We call c the challenge ciphertext.

4) The adversary $\mathcal{A}$ continues to have oracle access to $Enc_k(\cdot)$ and $Dec_k(\cdot)$, but is not allowed to query the latter on the challenge ciphertext itself. Eventually, $\mathcal{A}$ outputs a bit $b' \leftarrow \{0,1\}$.

5) The adversary wins the game if output of the game is 1, that is, $b' = b$.

Definition 3. *The symmetric encryption scheme SE has indistinguishable encryptions under a chosen-ciphertext attack(IND-CCA) if for every PPT adversaries A has the advantage $Adv^{c/a}_{SE,A}(\cdot)$ is negligible.*

3 Our Model

Our model extends the forward security model of the work [8] to untrusted update setting. We focus here on public-key encryption schemes secure against breakin attack in the untrusted update environments.

3.1 Forward-Secure Public-Key Encryption Scheme with Untrusted Update

We give the formal definition of Forward-Secure Public-Key encryption scheme with untrusted update (uufsPKE).

Definition 4. *A uufsPKE scheme consists of four algorithms, each of which is described in the following.*

$KeyGen_{uu}(\lambda, N)$. The key setup algorithm is a probabilistic algorithm that takes as input a security parameter λ and time period N, outputs decryption key $DecK$, initial encrypted secret key $EncSK_0$ and public parameters PK.

$Update_{uu}(EncSK_{i-1}, i)$. The untrusted update algorithm that takes as input the encrypted secret key $EncSK_{i-1}$ for time period $i-1$, generates a new encrypted secret key $EncSK_i$. Then deletes the old key $EncSK_{i-1}$. Note that this algorithm does not require the decryption key.

$Encrypt_{uu}(PK, i, M)$. The encryption algorithm is a probabilistic algorithm that takes as input public parameters, the current time period i and a message M, outputs the ciphertext CT for time period i.

$Decrypt_{uu}(PK, i, EncSK_i, DecK, CT)$. The decryption algorithm is a deterministic algorithm that takes as input public parameters PK, the current time period

i, the current encrypted secret key $EncSK_i$, the decryption key DecK and a ciphertext CT, outputs the message M.

Decyption Consistency Requirements. *For any message M, the public key PK, the decryption key DecK, the secret key SK_i for time period i and the output of $Encrypt_{uu}(PK, i, M)$, $Decrypt(PK, SK_w, Encrypt_{uu}(PK, i, M)) = M$ always holds.*

3.2 Security Definitions for uufsPKE

Now we give the formal security definition for Forward-secure Public-Key encryption scheme with untrusted update in terms of two games.

3.2.1 Forward Security

Formally, for a uufsPKE scheme, its semantic security against an adaptive chosen ciphertext attack under an adaptive break-in attack can be defined via the following game fsc between an attacker $\mathcal{A}$ and a challenger $\mathcal{B}$.

Setup Phase. The challenger $\mathcal{B}$ runs algorithm $\mathbf{KeyGen}_{uu}(\lambda, N)$ and gives $\mathcal{A}$ the resulting public parameters PK, keeping the secret key $(EncSK_0, DecK)$ to itself. Here, a handle counter i is set to 0.

Phase 1. $\mathcal{A}$ adaptively issues the following three queries:

- $update(i)$ queries. $\mathcal{B}$ runs algorithm $\mathbf{Update}_{uu}$ and updates the handle counter to $i \leftarrow i + 1$.
- $breakin(i')$ queries. At any time i', $\mathcal{B}$ firstly checks if $i' \leq N - 1$. If this is true, it responds with the corresponding private-key share $EncSK_{i'}$ for current time period i'.
- $decryption(j, CT)$ queries. At any time j, $\mathcal{A}$ picks a ciphertext CT and sends to $\mathcal{B}$. The challenger makes a call to $\mathbf{Decrypt}_{uu}(PK, j, EncSK_j, DecK, CT)$ using the corresponding private-key and forwards the result to $\mathcal{A}$.

Challenge Phase. $\mathcal{A}$ submits two message M_0, M_1 of equal size. $\mathcal{B}$ flips a uniform coin $v \xleftarrow{\$} \{0, 1\}$ and encrypts M_v under i^* with a call to $\mathbf{Encrypt}_{uu}(i^*, M_v, PK)$, where i^* is the index of the current time period. Then $\mathcal{B}$ sends the resulting ciphertext CT^* to $\mathcal{A}$.

Phase 2. The adversary $\mathcal{A}$ continues to issue additional queries as in Phase 1 other than $decryption(i^*, CT)$ with $CT \neq CT^*$ or $decryption(i, CT^*)$ with $i \neq i^*$.

Guess Phase. Finally, $\mathcal{A}$ outputs a guess $v' \xleftarrow{\$} \{0, 1\}$.

We refer to the above game as an IND-fs-CCA2 game. We let $Adv^{fsc}_{\mathbf{\Pi},\mathcal{A}}$ denote the advantage of an attacker $\mathcal{A}$ in this game fsc.

Definition 5. *An uufsPKE scheme Π is IND-fs-CCA2 secure if for every PPT adversary $\mathcal{A}$, we have $Adv^{fsc}_{\mathbf{\Pi},\mathcal{A}} \leq negl(\lambda)$ in the above IND-fs-CCA2 game.*

3.2.2 Update Security

Formally, we define update security for a uufsPKE scheme via the following game uuc between A and B.

Setup Phase. The challenger $\mathcal{B}$ runs algorithm $\mathbf{KeyGen}_{uu}(\lambda, N)$ and gives $(PK, EncSK_0)$ to $\mathcal{A}$, keeping the secret key $DecK$ for itself. Also, a handle counter i is set to 0.

Phase 1. $\mathcal{A}$ adaptively issues the following two queries:

- $update(i)$ queries. $\mathcal{B}$ updates the handle counter to $i \leftarrow i + 1$.
- $decryption(j, CT)$ queries. At any time j, $\mathcal{A}$ picks a ciphertext CT and sends to $\mathcal{B}$. The challenger runs algorithm $\mathbf{Decrypt}_{uu}$ using $(EncSK_0, DecK)$ and sends the result to $\mathcal{A}$.

Challenge Phase. Once $\mathcal{A}$ decides that Phase1 is over, it submits two message M_0, M_1 of equal size. $\mathcal{B}$ flips a uniform coin $c \xleftarrow{\$} \{0, 1\}$ and encrypts M_c under i^* with a call to $\mathbf{Encrypt}_{uu}(i^*, M_c, PK)$, where i^* is the index of the current time period. Then $\mathcal{B}$ sends the resulting ciphertext CT^* to $\mathcal{A}$.

Phase 2. $\mathcal{A}$ issues additional queries as in Phase 1 other than $decryption(i^*, CT)$ with $CT \neq CT^*$) or $decryption(i, CT^*)$ with $i \neq i^*$.

Guess Phase. Finally, $\mathcal{A}$ outputs a guess $c' \xleftarrow{\$} \{0, 1\}$.

We refer to the above adversary $\mathcal{A}$ as a IND-uu-CCA2 adversary. We let $Adv^{uuc}_{\mathbf{\Pi},\mathcal{A}}$ denote the advantage of an attacker $\mathcal{A}$ in this game uuc.

Definition 6. *An uufsPKE scheme Π is IND-uu-CCA2 secure if for any PPT adversary $\mathcal{A}$, we have $Adv^{uuc}_{\mathbf{\Pi},\mathcal{A}} \leq negl(\lambda)$ in the above IND-uu-CCA2 game.*

4 Generic Construction from BTE and SE

In our construction, it allows for automated updates of encrypted keys and the user holding the second factor does not have to intervene in operations where the update algorithm is programmed to update the blinded version of the secret key at the beginning of each period. The second factor is only needed for decrypting messages as in many typical implementations of public key encryption. Beyond the forward security requirement, such a scheme prevents an adversary just in possession of the encrypted secret key from forging ciphertext for past, current, and future periods.

4.1 The General Transformation

In this section, we present the generic construction of uufsPKE from any binary tree encryption. We apply a symmetric encryption scheme to implement update security. And we can use any chosen ciphertext secure symmetric encryption. Formally, a BTE consists of PPT algorithms $\mathcal{E}_1 = (Gen, Der, Enc, Dec)$ and a SE scheme $\mathcal{E}_2 = (Gen', Enc', Dec')$. Our scheme can be described by four algorithms, denoted by $\mathbf{\Pi} = (\mathbf{KeyGen_{uu}}, \mathbf{Update_{uu}}, \mathbf{Encrypt_{uu}}, \mathbf{Decrypt_{uu}})$.

$KeyGen_{uu}(k, N)$**.** It runs algorithm $Gen(k, \ell) \rightarrow (sk_\varepsilon, pk)$ that takes as input a security parameter $k \in \mathbb{N}$ and ℓ, the smallest integer satisfying $N \leq 2^\ell$. Firstly, sets $DecK \leftarrow Gen'(1^k)$ Then, symmetric encryption algorithm generates $esk_\varepsilon = Enc'_{DecK}(sk_\varepsilon)$ and uses esk_ε generating the initial encrypted secret key of our generic construction by calling algorithm $Der(.,.)$. Denoted by $EncSK_0 = (esk_{0^\ell}, \{esk_1, esk_{01}, esk_{001}, ..., esk_{0^{\ell-1}1}\})$. Finally, it returns public key (pk, N) and the secret key $(EncSK_0, DecK)$.

$Update_{uu}(EncSK_i, i+1)$**.** The encrypted secret key $EncSK_i$ be organized as a stack of node keys where the secret key $esk_{\langle i \rangle}$ on top. Here, $\langle i \rangle = i_1 i_2 ... i_\ell$ is the binary expression for the i^{th} time period. Firstly, pops the top off the stack. If $i_\ell = 0$, search the only path from node $\langle i \rangle$ to root, denoted by P_i. And generate another set $R_i = (\{esk_{i_1 ... i_l}\}_{l \in \{1,...,\ell\} s.t. i_l = 0})$. Push the node secret key onto the stack from R_i according the relation between the leaf $\langle i \rangle$ and the sibling of the node in R_i. The closer two nodes, then first-in stack. Otherwise, let $h \in \{1, ..., \ell\}$ denote the largest index such that $i_h = 0$. It have $w = i_1 i_2 ... i_{h-1} 1 \in \{0,1\}^h$ and recursively use $Der(w, esk_w)$ to generate node keys $esk_{w1}, esk_{w01}, ..., esk_{w0^{\ell-h-1}}, esk_{w0^{\ell-l}}$. Push all these node secret keys onto the stack by the reverse order. The new top of the stack is $esk_{w0^{\ell-l}}$ that is $\langle i+1 \rangle = w0^{\ell-l}$. In both cases, it erase the leaf node key $esk_{\langle i \rangle}$ and return the new stack.

$Encrypt_{uu}(i, pk, M)$**.** In period i, to encrypt a message $M \in G_T$, the sender parses $\langle i \rangle$ as $i_1 i_2 ... i_\ell$. Then, it computes $CT \leftarrow Enc(pk, \langle i \rangle, M)$.

$Decrypt_{uu}(i, DecK, EncSK_i, pk, CT)$**.** To decrypt ciphertext CT, it regenerates $sk_{\langle i \rangle} \leftarrow Dec'_{DecK}(esk_{\langle i \rangle})$. Then, it computes $M \leftarrow Dec(sk_{\langle i \rangle}, \langle\ i\ \rangle, CT)$.

4.2 Simulation Theorems

In this section, we give the proof on how to reduce the security of SN-CPA for BTE scheme and SE scheme against chosen ciphertext attack to forward security and update security for a uufsPKE scheme. We only consider the strongest security, that is, chosen ciphertext security. We will prove our general transformation's forward security and update security against an adaptive chosen ciphertext attack in the following two theorems.

Theorem 1. *Suppose $\mathcal{E}_1$ is a SN-CPA(resp.SN-CCA) secure BTE and $\mathcal{E}_2$ is a symmetric encryption scheme with chosen ciphertext security. Then the uufsPKE scheme $\mathbf{\Pi}$ described above is IND-fs-CPA(resp.IND-fs-CCA2) securtiy.*

Proof. Suppose there is an adversary $\mathcal{A}$ has non-negligible advantage in attacking forward security of the above scheme. That is, $Adv^{fsc}_{\mathcal{A},\mathbf{\Pi}} > \varepsilon$, ε is a negligible parameter. We build an algorithm $\mathcal{B}$ that breaks BTE scheme $\mathcal{E}_1$ with advantage $Adv^{fsc}_{\mathcal{A},\mathbf{\Pi}}/N(k)$ where $N(k)$ is polynomial in the security parameter k. Algorithm $\mathcal{B}$ uses $\mathcal{A}$ to interact with a BTE challenger as follows:

Initialization. Firstly, $\mathcal{A}$ outputs the challenge time period i^* and the corresponding to the node label $\langle i^* \rangle$ of the binary tree. Secondly, $\mathcal{B}$ runs $Gen'(k) \rightarrow DecK$. It outputs $(i^*, DecK)$.

Setup. The BTE challenger runs $\mathbf{Gen}(k, \ell) \rightarrow (sk_\varepsilon, PK)$ and gives to $\mathcal{B}$. And then $\mathcal{B}$ forwards $(PK, DecK)$ to $\mathcal{A}$.

Phase 1. $\mathcal{A}$ adaptively issues the following queries.

- $update(i)$ queries: $\mathcal{B}$ runs algorithm **Der(.,.)** and updates the handle counter to $i \leftarrow i+1$ at the same time.
- $breakin(i')$ queries: At any time i', $\mathcal{B}$ firstly checks if $i' \leq N-1$. If this is true and $i' \leq i^*$, then $\mathcal{B}$ outputs a random bit and halts. Otherwise, if $i' > i^*$, $\mathcal{B}$ runs $Enc'_{DecK}(sk_\varepsilon)$ to obtain esk_ε and forwards to the BTE challenger. Using esk_ε. The BTE challenger recursively apply algorithm **Der(.,.)** to obtain node keys and finally $EncSK_{\langle i' \rangle}$. Then it responds with the corresponding private-key $EncSK_{i'}$ for the current time period i'.
- $decryption(j, CT)$ queries: At any time j, $\mathcal{A}$ picks a ciphertext CT and sends to $\mathcal{B}$. If $i^* \leq j$, $\mathcal{B}$ decrypt the ciphertext by himself. Otherwise,$\mathcal{B}$ makes a call to $\mathbf{Dec'}_{DecK}(esk_{\langle j \rangle})$ and forwards to the BTE challenger for $\mathbf{Dec}(PK, j, sk_{\langle j \rangle}, CT)$ using the corresponding private-key. Then returns the result to $\mathcal{A}$.

Challenge Phase. $\mathcal{A}$ outputs two equal length messages M_0, M_1. If $i \leq i^*$, $\mathcal{B}$ forwards M_0, M_1 to the BTE challenger. It runs $\mathbf{Enc}(PK, \langle i^* \rangle, M_b)$ to obtain CT^* for a random $b \in \{0,1\}$ and gives $\mathcal{A}$ the challenge ciphertext CT^*. Otherwise, $\mathcal{B}$ outputs a random bit and halts.

Phase 2. $\mathcal{A}$ continues to issue queries as **Phase 1** where decryption queries for (j, CT) with $j \neq i^*$.

Guess Phase. Finally, $\mathcal{A}$ outputs its guess $b' \in \{0,1\}$ for b. $\mathcal{B}$ forwards b' to the BTE challenger and wins the game if $b = b'$.

This completes the description of algorithm $\mathcal{B}$. Let $Adv^{cpa-bte}_{\mathcal{B},\mathbf{\Pi}}$ be $\mathcal{B}$'s advantage in winning the BTE game cpa-bte. Let $Adv^{fsc}_{\mathcal{A},\mathbf{\Pi}}$ be $\mathcal{A}$'s advantage in winning the fsc game of uufsPKE scheme. It is straightforward to see that when $i^* = i$ the copy of $\mathcal{A}$ running within $\mathcal{B}$ has exactly the same view as in a real fsc game. Since $\mathcal{B}$ guesses $i^* = i$ with probability $1/N$, we have that $\mathcal{A}$ correctly predicts the bit b with advantage $Adv^{fsc}_{\mathcal{A},\mathbf{\Pi}}/N(k)$. □

Theorem 2. *Suppose $\mathcal{E}_1$ is a SN-CPA(resp. SN-CCA) secure BTE and $\mathcal{E}_2$ is a symmetric encryption scheme against chosen ciphertext attack. Then the uufsPKE scheme $\mathbf{\Pi}$ above is IND-uu-CPA(resp.IND-uu-CCA2) security.*

Proof. Suppose $\mathcal{A}$ wins uuc game with non-negligible advantage in the above scheme. That is, $Adv^{uuc}_{\mathcal{A},\mathbf{\Pi}} > \varepsilon$ where ε is a negligible parameter. Then we show how to construct an algorithm $\mathcal{B}$ that breaks SE scheme $\mathcal{E}_2$. Algorithm $\mathcal{B}$ starts by breaking the SE scheme $\mathcal{E}_2$. Using $\mathcal{A}$, $\mathcal{B}$ interacts with a SE challenger as follows:

Setup. To launch the game, $\mathcal{B}$ runs $\mathbf{Gen}(k,\ell) \rightarrow (sk_\varepsilon, PK)$. It sends sk_ε to the SE challenger and obtains the initial encrypted node key esk_ε. Using esk_ε, $\mathcal{B}$ recursively runs algorithm **Der(.,.)** and generates all the encrypted keys $\{EncSK_0, EncSK_1, ..., EncSK_N\}$. And then, $\mathcal{B}$ sends the set of all the encrypted keys to $\mathcal{A}$.

Phase 1. $\mathcal{A}$ adaptively issues the following two queries:

- *update*(i) queries. $\mathcal{B}$ updates the handle counter to $i \leftarrow i+1$.
- *decryption*(j, CT) queries. At any time j, $\mathcal{A}$ picks a ciphertext CT and sends to $\mathcal{B}$. $\mathcal{B}$ request the secret key of time period j for the SE challenger. It runs algorithm **Dec'**$(EncSK_j)$ and outputs the secret key $sk_{\langle j\rangle}$. $\mathcal{B}$ computes $\mathbf{Dec}(sk_{\langle j\rangle}, CT) = M$ and sends the result to $\mathcal{A}$.

Challenge Phase. $\mathcal{A}$ chooses two equal length messages M_0, M_1 and sends to $\mathcal{B}$. It runs $\mathbf{Enc}(PK, \langle i^*\rangle, M_b)$ to obtain CT^* for a random $b \in \{0,1\}$ and gives $\mathcal{A}$ the challenge ciphertext CT^*.
Phase 2. $\mathcal{A}$ continues to issue queries as **Phase 1** where decryption queries for (j, CT) with $j \neq i^*$.

Guess Phase. Finally, $\mathcal{A}$ outputs its guess $b' \in \{0,1\}$ for b. $\mathcal{B}$ forwards b' to the SE challenger and wins the game if $b = b'$.

□

5 A Concrete Forward-Secure Public-Key Encryption Scheme with Untrusted Update in Standard Model

In this section, we give a concrete construction of BTE scheme and SE scheme, respectively. Combined with both schemes, we present a uufsPKE scheme. We prove its security in standard model. Finally, we compare our scheme with the previous.

5.1 Boilding Blocks: BTE and SE

Firstly, we propose a new binary tree encryption scheme $\mathcal{E}_1$ with SN-CPA based on dual system encryption and prove its security. For simplicity, we do not consider chosen-ciphertext security of binary tree encryption which can be added by simply using CHK transformation [9]. In the description below, we imagine binary tree of height ℓ where the root (at depth 0) has label ε.When a node at depth $\leq \ell$ has label w, its children are labeled with $w0$ and $w1$.Besides,$\langle i\rangle$ stands for the ℓ-bit representation of integer i. The leaves of the reed correspond to successive time periods, stage i being associated with the leaf labeled by $\langle i\rangle$.The BTE scheme denote by $\mathcal{E}_\mathbf{1} = (\mathbf{Gen}, \mathbf{Der}, \mathbf{Enc}, \mathbf{Dec})$ that works as follows.

Gen(k,ℓ) is an algorithm that, given a security parameter k and the maximum tree depth ℓ.

1) Choose bilinear map groups (G, G_T) of order $N = p_1p_2p_3$ where p_i is the order of the subgroup G_i in G, with $p_i > 2^k$ for each $i \in \{1,2,3\}$.
2) Let $w = w_1...w_l$ be an l-bits string representing a node of binary tree and $w_i \in \{0,1\}$ for all $i \in \{1,...,l\}$. Define a function $H : \{0,1\}^{\leq \ell} \rightarrow G_1$ as $H(w) = h_0 \prod_{j=1}^{l} h_j^{w_j}$.
3) Outputs the public key $pk = (g_1, g_3, V = e(g_1,g_1)^\alpha, h_0, h_1, ..., h_\ell, H)$ and a root secret key $sk_\varepsilon = \alpha$ for independent and uniformly random $\alpha \in Z_N, (h_0, h_1, ..., h_\ell) \in G_1^{\ell+1}, g_1 \in G_1$ and g_3 is a generator of G_3.

Der(pk, w, sk_w) is an algorithm that deduce the secret keys sk_{w0}, sk_{w1} where $w0, w1$ are two children of w. Execute the following steps.

1) Let $w = w_1...w_l$. Parse $sk_w = (k_0, k_1, s_l, ..., s_\ell) = (g_1^\alpha \cdot H(w_1...w_l)^r \cdot g_3^u, g_1^r \cdot g_3^{v_0}, h_l^r \cdot g_3^{v_1}, ..., h_\ell^r \cdot g_3^{v_\ell})$, $u, v_0, v_1, ..., v_\ell \in Z_{p_3}, r \in Z_N$.
2) For $j \in \{0,1\}$, output $sk_{wj} = (k_0 \cdot s_l^j \cdot H(w_1...w_lj)^{r'_j} \cdot g_3^{u'}, k_1 \cdot g_1^{r'_j} \cdot g_3^{v'_0}, s_{l+1} \cdot h_{l+1}^{r'_j} \cdot g_3^{v'_1}, ..., s_\ell \cdot h_\ell^{r'_j} \cdot g_3^{v'_\ell}) = (g_1^\alpha \cdot H(w_1...w_lj)^{r_j} \cdot g_3^{\widetilde{u}}, g_1^{r_j} \cdot g_3^{\widetilde{v_0}}, h_l^{r_j} \cdot g_3^{\widetilde{v_1}}, ..., h_\ell^{r_j} \cdot g_3^{\widetilde{v_\ell}})$ where $u', v'_0, v'_1, ..., v'_\ell \in$

$Z_{p_3}, r'_j \in Z_N$ and $r_j = r'_j + r, \widetilde{u} = u + u', \widetilde{v_0} = v_0 + v'_0, ..., \widetilde{v_\ell} = v_\ell + v'_\ell$.

$Enc(\langle i \rangle, pk, M)$ does the following:

1) Let $\langle i \rangle = i_1...i_\ell \in \{0,1\}^\ell$, select random $s \in Z_N$.
2) Compute and output the ciphertext $CT = (i, M \cdot V^s, g_1^s, H(i_1...i_\ell)^s)$.

$Dec(w, sk_w, CT)$ does the following:

1) et $\langle i \rangle = i_1...i_\ell \in \{0,1\}^\ell$, parse $sk_{\langle i \rangle}$ as (k_0, k_1) and parse CT as (i, C_0, C_1, C_2).
2) Output the message $M = \dfrac{C_0 \cdot e(k_1, C_2)}{e(k_0, C_1)}$.

Correctness. Assuming the ciphertext is well-formed, we have

$$\begin{aligned} \frac{C_0 \cdot e(k_1, C_2)}{e(k_0, C_1)} &= \frac{M \cdot V^s \cdot e(g_1^r \cdot g_3^{v_0}, H(i_1...i_\ell)^s)}{e(g_1^\alpha \cdot H(w_1...w_l)^r \cdot g_3^u, g_1^s)} \\ &= M. \end{aligned}$$

Secondly, we give a construction of SE scheme $\boldsymbol{\mathcal{E}_2} = (\mathbf{Gen'}, \mathbf{Enc'}, \mathbf{Dec'})$ applying to build uufsPKE scheme. The encryption algorithm $Enc'_{sk_{ss}}(m) = (c_1, c_2, ..., c_\kappa, m \cdot \prod_{j \in [\kappa]} c_j^{\mu_j})$ for independent and uniformly random $c_j \in G$. The decryption algorithm $Dec'_{sk_{ss}}(c_1, c_2, ..., c_\kappa, c_0)$ firstly parses the ciphertext c as $(c_1, c_2, ..., c_\kappa, c_0)$. And compute the message

$$m = \frac{c_0}{\prod_{j \in [\kappa]} c_j^{\mu_j}}.$$

5.2 A Concrete uufsPKE Scheme

Our concrete uufsPKE scheme consists of the following algorithms where periods are indexed from 0 to T-1 with $T = 2^\ell$.

$KeyGen_{uu}(k, N)$.

1) Run $Gen(k, \ell) \rightarrow (sk_\varepsilon, pk)$ where $pk = (g_1, g_3, V = e(g_1, g_1)^\alpha, h_0, h_1, ..., h_\ell, H)$ and $sk_\varepsilon = \alpha$.
2) Generate $G(r) \rightarrow DecK$, denoted by $DecK = (d_1, d_2, ..., d_t)$ where $d_i \in Z_N$.
3) Compute the initial encrypted root key $esk_\varepsilon = Enc'_{DecK}(sk_\varepsilon) = (c_1, c_2, ..., c_t, sk_\varepsilon \prod_{i \in [t]} c_i^{d_i})$.
4) The initial encrypted secret key $EncSK_0 = (esk_{0^\ell}, \{esk_1, esk_{01}, esk_{001}, ..., esk_{0^{\ell-1}1}\})$ using esk_ε recursively apply algorithm Der.

$Update_{uu}(EncSK_i, i+1)$.

1) Parse $\langle i \rangle = i_1...i_\ell \in \{0,1\}^\ell$ and $EncSK_i = (esk_{\langle i \rangle}, \{esk_{i_1...i_l}\}_{l \in \{1,...,\ell\} s.t. i_l = 0})$. And delete $esk_{\langle i \rangle}$.
2) If $i_\ell = 0$, $EncSK_{i+1} = (esk_{i_1 \cdot i_{\ell-1}}, \{esk_{i_1 \cdot i_{l-1}}\}_{l \in \{1,...,\ell-1\} s.t. i_l = 0})$, that is, $EncSK_{i+1}$ includes the remaining node keys. Otherwise, let $l' \in \{0,1\}^\ell$ be the largest index such that $i_{l'} = 0$. Let $w' = i_1...i_{l'-1}1 \in \{0,1\}^{l'}$. Recursively run Der for the node key $esk_{w'}$ to generate node keys $esk_{w'1}$, $esk_{w'01}$, $\cdots$, $esk_{w'0^{\ell-l'-1}1}$ and $esk_{w'0^{\ell-l'-1}1} = esk_{\langle i+1 \rangle}$. Delete $esk_{w'}$ and return $EncSK_{i+1} = (esk_{\langle i+1 \rangle}, \{esk_{i_1...i_{l-1}1}\}_{l \in \{1,...,l'-1\} s.t. i_l = 0}, \{esk_{w'1}, esk_{w'01}, \cdots, esk_{w'0^{\ell-l'-1}1}\})$.

$Encrypt_{uu}(i, pk, M)$. For $i \in [1, N]$, to encrypt the message M, does the following.

1) Parse $\langle i \rangle$ as $i_1...i_\ell \in \{0,1\}^\ell$.
2) Run $Enc(\langle i \rangle, pk, M)$ and return the ciphertext $CT = (i, M \cdot e(g_1, g_1)^{\alpha s}, g_1^s, H(i_1...i_\ell)^s)$.

$Decrypt_{uu}(i, EncSK_i, pk, CT)$. Given a ciphertext $CT = (i, C_0, C_1, C_2)$ and a encrypted secret key $EncSK_i = (esk_{\langle i \rangle}, \{esk_{i_1...i_l}\}_{l \in \{1,...,\ell\} s.t. i_l = 0})$ for the current period i.

1) Parse $esk_{\langle i \rangle}$ as $(esk^0_{\langle i \rangle}, esk^1_{\langle i \rangle})$.
2) Compute $sk_{\langle i \rangle} = (\dfrac{esk^0_{\langle i \rangle}}{\prod_{i \in [t]} c_i^{d_i}}, \dfrac{esk^1_{\langle i \rangle}}{\prod_{i \in [t]} c_i^{d_i}})$ by applying decryption algorithm Dec'.
3) Run the algorithm $Dec(sk_{\langle i \rangle}, \langle i \rangle, CT)$ to obtain the message M.

5.3 Security Proof

In the following, we devote to prove SN-CPA for the above BTE scheme $\mathcal{E}_1$ under three static assumptions in the standard model. Our security proof will use semi-functional ciphertext and semi-functional node keys which defined as follows. All the ciphertexts and node keys defined by the following are normal, where by normal we mean that they have no G_2 parts. On the other hand, a semi-functional key or ciphertext has G_2 parts.

- semi-functional ciphertexts is generated from a normal ciphertext $CT = (C'_0, C'_1, C'_2)$ and some $g_2 \in G_{p_2}$, by choosing random $x, z_c \xleftarrow{\$} Z_N$ and setting $CT^{semi} = (C'_0, C'_1 \cdot g_2^{x \cdot z_c}, C'_2 \cdot g_2^x)$.
- semi-functional node key are generated from a normal node key $sk'_w = (k'_0, k'_1, s'_l, ..., s'_\ell)$ by choosing random $y, z_w \xleftarrow{\$} Z_N$, and setting $sk_w^{semi1} = (k'_0 \cdot g_2^{y \cdot z_w}, k'_1 \cdot g_2^y, s'_l \cdot g_2^y, ..., s'_\ell \cdot g_2^y)$.

We now prove the semantic security of this BTE against selective-node chosen plaintext attacks(SN-CPA). In order to prove security we need a hybrid argument using a sequence of games. For each i, we denote by S_i the probability that the challenger returns 1 at the end of $Game_i$. We also define $Adv_i = |Pr[S_i - 1/2]|$ for each i.

- *Game*: It is the real security game as described in Section 2.3.
- *Game′*: It is exactly the same as *Game* except that in the challenge phase, the ciphertext responds with the semi-functional ciphertext instead of normal ciphertext. Besides, at the begging of the game, the challenger chooses an index $i^* \xleftarrow{R} \{0,1,...,T-1\}$. At the challenge phase, the challenger halts and outputs a random bit if the challenge ciphertext is encrypted for a period i such that $i \neq i^*$. Because the choice of i^* is independent of $\mathcal{A}$'s view, we have $Pr[S_1] \leq Pr[S_0]/T$. Note that we denote $Path^*$ the path from the root to the leaf associated with i^*.
- $Game_k$ In this game, for $k \in [1, T-1]$, the ciphertext given to the attacker is semi-functional. For the first k keys, if the corresponding node $w^k \in Path^*$, then returns semi-functional node key. Otherwise, the rest of the keys are normal.
- $Game_T$ is the same as $Game_{T-1}$ but the semi-functional challenge ciphertext is replaced by a a semi-functional encryption of a random message instead of M_b.

Theorem 3. *If Assumption1,2 and 3 hold, then our BTE scheme $\mathcal{E}_1$ is a SN-CPA.*

Proof. The proof proceeds using a sequence of games including steps similar to [14]. In order to prove this theorem we need the following lemmas:

Lemma 1. *Suppose there exists an algorithm $\mathcal{A}$ such that $Adv^{Game}_{\mathcal{A},\mathcal{E}_1} - Adv^{Game'}_{\mathcal{A},\mathcal{E}_1} \leq \epsilon$. Then we can build an algorithm $\mathcal{B}$ with advantage ϵ in breaking Assumption 1.*

Proof. Suppose a PPT challenger $\mathcal{B}$ that breaks Assumption 1 with the help of a PPT adversary $\mathcal{A}$. $\mathcal{B}$ simulates *Game* or *Game′*. Initially, $\mathcal{B}$ receives input from the assumption's challenger, i.e. $D^1 = (N, G, G_T, e, g_1, g_3)$ and a challenge term T which is equal to g_1^z and $g_1^z g_2^\xi$. Algorithm $\mathcal{B}$ works as follows:

Setup Phase. $\mathcal{B}$ picks $\alpha, x_0, x_1, ..., x_\ell \xleftarrow{\$} Z_N$ and computes $V = e(g_1, g_1)^\alpha, h_0 = g_1^{x_0}, h_1 = g_1^{x_1}, ...,$ and $h_\ell = g_1^{x_\ell}$. It gives the public parameters $pk = (g_1, g_3, V = e(g_1, g_1)^\alpha, h_0, h_1, ..., h_\ell, H)$ to $\mathcal{A}$ where (N, e, g_1, g_3) are given by the challenger $\mathcal{B}$. Initially, $\mathcal{A}$ outputs the target node $w^* = w_1^* w_2^* ... w_l^*$ with $l \leq \ell$. Next, $\mathcal{B}$ constructs node keys as follows. For a node label $w = w_1 w_2 ... w_l (l < \ell)$, $\mathcal{B}$ chooses $u, v_0, v_1, ..., v_\ell \in Z_{p_3}, r \xleftarrow{\$} Z_N$ and sets $sk_w = (g_1^\alpha \cdot \prod_{j=1}^{l} (g_1^{x_j \cdot w_j})^r \cdot g_3^u, g_1^r \cdot g_3^{v_0}, h_l^r \cdot g_3^{v_1}, ..., h_\ell^r \cdot g_3^{v_\ell})$.

Phase 1. $\mathcal{B}$ provides $\mathcal{A}$ with all secret keys for sibling of the nodes on the path from the root to w^*, as well as for the children of w^*.

Challenge Phase : Once $\mathcal{A}$ outputs two equal-length plaintexts $M_0, M_1 \in \mathcal{M}$ on which it wishes to be challenged. $\mathcal{B}$ flips a random coin $c \in \{0,1\}$, and generates the challenge ciphertext to be $CT^* = (M_c \cdot e(T, g_1)^\alpha, T, \prod_{j=1}^{l} T^{x_j w_j})$, which is sent to $\mathcal{A}$.

Analysis. If $T = g_1^z$, then this is a normal ciphertext which has no G_2 component. If $T = g_1^z g_2^\xi$, this is a semi-functional ciphertext with $z_c = \Sigma_{j=1}^{l} x_j w_j$. If $\mathcal{A}$ succeeds in distinguishing these two games then our challenger $\mathcal{B}$ can use $\mathcal{A}$ to break Assumption 1. Thus if Assumption 1 is holds, these two games are indistinguishable.

□

It is straightforward that from *Game′* to $Game_1$ is just a conceptual change since the adversary $\mathcal{A}$'s view is the same in both games. Thus we have $Adv_1 = Adv'/T$.

Lemma 2. *Suppose there exists an algorithm $\mathcal{A}$ such that $Adv^{Game_{k-1}}_{\mathcal{A},\mathcal{E}_1} - Adv^{Game_k}_{\mathcal{A},\mathcal{E}_1} \leq \epsilon$. Then we can build an algorithm $\mathcal{B}$ with advantage ϵ in breaking Assumption 2.*

Proof. $\mathcal{B}$ first receives $D^2 = (N, G, G_T, e, g_1, g_3, g_1^z g_2^v, g_2^\mu g_3^\tau)$ and T where $T = g_1^\gamma g_3^\kappa$ or $T = g_1^\gamma g_2^\xi g_3^\kappa$. $\mathcal{B}$ picks a random exponents $a, b, \alpha \in Z_N$ and sets the public parameters as Lemma 1.

When $\mathcal{A}$ requests the i^{th} key for the period i corresponding to node $\langle i \rangle$ in the tree. If $i < k$ and $\langle i \rangle \in Path^*$, $\mathcal{B}$ creates a semi-functional node key of $Type1$. It does this by choosing random exponents $y, z_w, r \in Z_N$ and setting $sk_w^{semi1} = (g_1^\alpha \cdot \prod_{j=1}^{l} (g_1^{x_j \cdot w_j})^r \cdot (g_2^\mu g_3^\tau)^{y \cdot z_w}, g_1^r \cdot (g_2^\mu g_3^\tau)^y, h_l^r \cdot (g_2^\mu g_3^\tau)^y, ..., h_\ell^r \cdot (g_2^\mu g_3^\tau)^y)$. This is a properly distributed semi-functional node key of $Type1$ with G_2 component g_2^y. For $i > k$, $\mathcal{B}$ generates normal keys by using random exponents $r, y \in Z_N$ and setting $sk_w = (g_1^\alpha \cdot \prod_{j=1}^{l} (g_1^{x_j \cdot w_j})^r \cdot g_3^y, g_1^r \cdot g_3^y, h_l^r \cdot g_3^y, ..., h_\ell^r \cdot g_3^y)$. To create the k^{th} requested key, $\mathcal{B}$ sets $z_k = \sum_{j=1}^{l} x_j \cdot w_j$, chooses a random exponent $y_w \in Z_N$, and sets $sk_w^* = (g_1^\alpha \cdot T^{z_k} \cdot g_3^{y_w}, T, h_l^r \cdot T^{y_w}, ..., h_\ell^r \cdot T^{y_w})$.

Challenge Phase. $\mathcal{A}$ sends $\mathcal{B}$ two equal-length plaintexts $M_0, M_1 \in \mathcal{M}$. $\mathcal{B}$ chooses a random coin $v \in \{0,1\}$, and generates the ciphertext $CT = (M_c \cdot e(g_1^z g_2^v, g_1)^\alpha, g_1^z g_2^v, \prod_{j=1}^{l} (g_1^z g_2^v)^{x_j w_j})$, which is sent to $\mathcal{A}$.

Analysis. We note that this sets $z_c = \Sigma_{j=1}^{l} x_j w_j$. If $T = g_1^\gamma g_3^\kappa$, then this is a normal ciphertext which has no G_2 component. If $T = g_1^\gamma g_2^\xi g_3^\kappa$, this is a semi-functional ciphertext with $z_c = \Sigma_{j=1}^{l} x_j w_j$. If $\mathcal{A}$ succeeds in distinguishing these two games then our challenger $\mathcal{B}$ can use $\mathcal{A}$ to break Assumption 2. Thus if Assumption 2 is holds, these two games are indistinguishable.

□

Lemma 3. *Suppose there exists an algorithm $\mathcal{A}$ such that $Adv^{Game_{T-1}}_{\mathcal{A},\mathcal{E}_1} - Adv^{Game_T}_{\mathcal{A},\mathcal{E}_1} \leq \epsilon$. Then we can build an algorithm $\mathcal{B}$ with advantage ϵ in breaking Assumption 3.*

Table 1: Comparison of performance with existing schemes

S(Scheme)	Encryption/Decryption	Hard Problems	Key update time	Ciphertext length	Public-key/Secret-key size
S of [16]	$\mathcal{O}(N \cdot (loglogN)^2)$	Three static assumptions	$\mathcal{O}(logN)$	$\mathcal{O}(logN)$	$\mathcal{O}(logN)$
Our Scheme	$\mathcal{O}(1)$	$BDDH$	$\mathcal{O}(logN)$	$\mathcal{O}(1)$	$\mathcal{O}(logN)$

Proof. $\mathcal{B}$ first receives $(N, G, G_T, e, g_1, g_3, g_1^{\alpha} g_2^{\xi}, g_1^{s} g_2^{\mu}, g_2^{\varpi})$ and T, where $T = e(g_1, g_1)^{\alpha s}$ or $T = e(g_1, g_1)^{\alpha w}$. $\mathcal{B}$ chooses random exponents $x_0, x_1, ..., x_\ell \overset{\$}{\leftarrow} Z_N$ and sets the public parameters as $V = e(g_1, g_1^{\alpha} g_2^{\xi}), h_0 = g_1^{x_0}, h_1 = g_1^{x_1}, ...,$ and $h_\ell = g_1^{x_\ell}$. It sends these to $\mathcal{A}$. When $\mathcal{A}$ requests a key for time period i, $\mathcal{B}$ generates a semi-functional. It does this by setting $sk_w = (g_1^{\alpha} g_2^{\xi} \cdot \prod_{j=1}^{l} (g_1^{x_j \cdot w_j})^r \cdot g_3^u, g_1^r \cdot g_3^{v_0}, h_l^r \cdot g_3^{v_1}, ..., h_\ell^r \cdot g_3^{v_\ell})$. After providing the appropriate secret keys, $\mathcal{B}$ responds to the challenge query from $\mathcal{A}$. Specifically, $\mathcal{B}$ chooses a random bit b and returns $CT = (M_b \cdot T, g_1^s g_2^{\mu}, (g_1^s g_2^{\mu})^{\Sigma_{j=1}^{l} x_j w_j})$. If $T = e(g_1, g_1)^{\alpha s}$, then this is a properly distributed semi-functional ciphertext with message M_b. On the other hand, if $T \overset{\$}{\leftarrow} G_T$, then this is a semi-functional ciphertext with a random message. Therefore, the value of b is information-theoretically hidden and the probability of success of any algorithm $\mathcal{A}$ in this game is exactly $1/2$, since $b \overset{\$}{\leftarrow} \{0, 1\}$. Thus, $\mathcal{B}$ can use the output of $\mathcal{A}$ to break Assumption 3 with non-negligible advantage. □

This conclude the proof of Theorem 3. □

According to Theorem 1 and Theorem 2, we conclude our concrete scheme has forward security and update security.

5.4 Comparison with the Existing Schemes

Now we compare the efficiency of our method with the prevail classic uufsPKE [16] in Table 1. Similarly, we associate time periods N with the leaves only, we have the same efficiency of key generation phase. From Table 1, our scheme maintains the efficiency of key update and the size of public-key, secret-key. Using the techniques of Lewko et al. [14], the efficiency of our encryption/decrytion scheme and size of the ciphertext from $\mathcal{O}(logN)$ to $\mathcal{O}(1)$. Therefore, considering the security and the performance efficiency, our scheme is much better than previous schemes.

6 Conclusions

In this paper, motivated by the work of Libert [16], we give formal definition for uufsPKE scheme, as well as a general framework for constructing uufsPKE from BTE and SE scheme. We have proved our scheme is IND-uufs-CCA secure in standard model. Furthermore, we give a concrete construction of BTE scheme and prove the SN-CPA security under three static assumptions. Finally, we presented the first completely uufsPKE scheme that is forward security and update security against chosen ciphertext attack without random oracle. Compared with existing scheme, our scheme performs more faster.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (NO. 61370203), the Key Technology Research and Development Program of Sichuan Province and Chengdu Municipality (NO. szjj2015-054), the doctoral research fund of Weifang university (No.2015BS11).

References

[1] M. Abdalla, L. Reyzin, "A new forward-secure digital signature scheme", in *Advances in Cryptology (ASIACRYPT'00)*, pp. 116–129, Kyoto, Japan, 2000.

[2] M. Bellare, S. K. Miner, "A forward-secure digital signature scheme", in *Advances in Cryptology (CRYPTO'99)*, pp. 431–448, Santa Barbara, California, USA, 1999.

[3] M. Bellare, B. Yee, "Forward security in private-key cryptography", in *The Cryptographer's Track at RSA Conference (CT-RSA 2003)*, pp. 1–18, San Francisco, CA, USA, 2003.

[4] D. Boneh, X. Boyen, and E. J. Goh, "Hierarchical identity based encryption with constant size ciphertext", in *Advances in Cryptology (EUROCRYPT'05)*, pp. 440–456, Aarhus, Denmark, 2005.

[5] D. Boneh, M. Franklin, "Identity-based encryption from the weil paring", in *Advances in Cryptology (CRYPTO'01)*, pp. 213–229, Santa Barbara, California, USA, 2001.

[6] D. Boneh, E. Goh, and K. Nissim, "Evaluating 2-dnf formulas on ciphertexts", in *The second Theory of Cryptography Confference (TCC 2005)*, pp. 325–341, Cambridge, MA, USA, 2005.

[7] X. Boyen, H. Shacham, and E. Shen, et. al., "Forward-secure signatures with untrusted update", in *The 13th ACM Conference on Computer and Communications Security (CCS'06)*, pp. 191–200, Alexandria, VA, USA, 2006.

[8] R. Canetti, S. Halevi, J. Katz, "A forward-secure public-key encryption scheme", in *Advances in Cryptology (EUROCRYPT'03)*, pp. 255–271, Warsaw, Poland, 2003.

[9] R. Canetti, S. Halevi, and J. Katz, "Chosen-ciphertext security from Identity-based encryption", in *Advances in Cryptology (EUROCRYPT'04)*, pp. 207–222, Interlaken, Switzerland, 2004.
[10] G. Itkis, L. Reyzin, "Forward-secure signatures with optimal signing and verifying", in *Advances in Cryptology (CRYPTO'01)*, pp. 499–514, Santa Barbara, California, USA, 2001.
[11] A. Kozlov, L. Reyzin, "Forward-secure signatures with fast key update", in *Security in Communication Networks*, pp. 241-256, Amalfi, Italy, 2003.
[12] H. Krawczyk, "Simple forward-secure signatures from any signature scheme", in *The 10th ACM Confference on Computer and Communications Security*, pp. 108–115, New York, 2000.
[13] S. Kunwar, C. Pandurangan, A. K. Banerjee, "Lattice based forward-secure identity based encryption scheme", *Journal of Internet Services and Information Security*, vol. 3, no. 1/2, pp. 5-19, 2013.
[14] A. Lewko, B. Waters, "New technique for dual system encryption and fully secure HIBE with short ciphertexts", in *The Seventh Theory of Cryptography Conference (TCC 2010)*, pp. 455–479, Zurich, Switzerland, 2010.
[15] B. Libert, J. Quisquater, M. Yung, "Forward-secure signatures in untrusted update environments: Efficient and generic constructions", in *The 14th ACM Conference on Computer and Communications Security (CCS'07)*, pp. 266–275, Alexandria, VA, USA, 2007.
[16] B. Libert, J. Quisquater, and M. Yung, "Key evolution systems in untrusted update environments", *ACM Transactions on Information and Systems Security*, vol. 13, no. 4, pp. 1-34, 2010.
[17] T. Malkin, D. Micciancio, and S. K. Miner, "Efficient generic forward-secure signatures with an unbounded number of time periods", in *Advances in Cryptology (EUROCRYPT'02)*, pp. 400–417, Amsterdam, The Netherlands, 2002.
[18] J. Nieto, M. Manulis, D. D. Sun, "Forward-secure hierarchical predicate encryption", in *Pairing-Based Cryptography (Pairing'12)*, pp. 83–101, Cologne, Germany, 2012.
[19] Y. L. Ren, Z. H. Niu, X. P. Zhang, "Fully anonymous identity-based broadcast encryption without random oracles", *International Journal of Network Security*, vol. 16, no. 4, pp. 256–264, 2014.
[20] D. Yao, Y. Dodis, and N. Fazio, et. al., "ID-based encryption for complex hierarchies with applications to forward security and broadcast encryption", in *The eleven ACM Conference on Computer and Communication Security*, pp. 354–363, Washington, DC, USA, 2004.
[21] J. Yu, F. Y. Kong, and X. G. Cheng, et. al., "Forward-secure identity-based public-key encryption without random oracles", *Fundamenta Informatica*, vol. 111, no. 2, pp. 241–256, 2011.

Xiujie Zhang biography. Xiujie Zhang is a lecturer in Weifang University. She received her PhD from University of Electronic Science and Technology of China(UESTC). Her research interests include leakage resilient cryptosystems, applied cryptography and information security.

Chunxiang Xu biography. Chunxiang Xu received her BS, MS and PhD from Xidian University, China, in 1985, 1988 and 2004, respectively. She is a professor at UESTC. Her research interests include information security, cloud computing security and cryptography.

Cryptanalysis of Two PAKE Protocols for Body Area Networks and Smart Environments

Mohsen Toorani
Department of Informatics, University of Bergen
P.O. Box 7803, N-5020 Bergen, Norway
(Email: mohsen.toorani@uib.no)

(Received Nov. 23, 2014; revised and accepted May 20 & May 26, 2015)

Abstract

Password-authenticated key exchange (PAKE) protocols enable two or more entities to authenticate each other and share a strong cryptographic key based on a pre-shared human memorable password. In this paper, we present several attacks on two recent elliptic curve-based PAKE protocols that have been suggested for use in body area networks and smart environments. A variant of the first PAKE protocol has been included in the latest standard for body area networks. The second PAKE protocol is a modified variant of the first protocol, and has been proposed for bridging the user interface gap in pervasive computing and smart environments.

Keywords: Dictionary attack, elliptic curves, forward secrecy, impersonation attack, invalid-curve attack

1 Introduction

Authenticated key exchange (AKE) protocols aim to establish a cryptographic session key between legitimate entities in an authenticated manner. Many AKE protocols have been proposed in literature, but some of them have security problems [3, 17, 18]. Password-authenticated key exchange (PAKE) protocols are password-based AKE protocols that use pre-shared human memorable passwords for authentication and establishing a cryptographically strong secret key. Since introduction of the first PAKE protocol in 1992 [2], many PAKE protocols have been proposed. Many of those protocols have been shown to be insecure [7, 11, 12, 16].

Traditionally, PAKE protocols use just a shared password between a client and server. Even though people are always recommended to select strong passwords, many people choose simple passwords. As a countermeasure, many PAKE protocols try to provide multi-factor authentication by combining passwords with other parameters like public keys or symmetric keys.

Several security models have been developed for AKE and PAKE protocols, each of them has a different assumption for capabilities of an adversary. A protocol that is proved to be secure in a security model would be insecure in other security models. It is because of different assumptions on adversarial power and valid attacks.

Several security attributes should be provided by PAKE protocols, and they should withstand well-known attacks. Security requirements for PAKE protocols depend on number of participants and secret parameters that are used for constructing the protocol. Some security requirements of PAKE protocols are common with AKE protocols. This includes mutual authentication, known-key security, forward secrecy, key control, and resilience to impersonation, replay, unknown key-share (UKS), and Denning-Sacco attacks [9, 10, 16]. Furthermore, any PAKE protocol must be resilient to dictionary [5, 16] or password guessing [4] attacks. Such requirement is very subtle because people usually select weak memorable passwords. Then, instead of a brute-force attack, an attacker would use a dictionary of most probable passwords. Based on secret parameters that are used for building a protocol, there are some more requirements that should be satisfied. Those PAKE protocols that use public keys are expected to provide resilience to the key compromise impersonation (KCI) attack and its variants.

The wireless body area network (WBAN) is a wireless network of wearable computing devices [13]. WBAN

has many applications in military, ubiquitous health care, sport, and entertainment. As WBANs are resource-constrained in terms of power, memory, communication rate and computational capability, security solutions proposed for other networks may not be suitable for WBANs. The latest standardization of WBANs is the IEEE 802.15.6 standard [1], but it has security problems [19].

Ho [8] presented four elliptic curve-based key agreement protocols that are designed for different device configurations in WBAN, and can be implemented as a versatile suite through a single implementation. It includes one unauthenticated key exchange protocol, one AKE protocol with out-of-band transfer of public key, one PAKE protocol, and one AKE protocol for devices with numerical display. Variants of those protocols have been included in the IEEE 802.15.6 standard. However, there are two major differences between Ho's protocols and the protocols in the IEEE 802.15.6 standard. The first difference is that Ho's protocols do not consider validation of public keys which makes the protocols vulnerable to some extra attacks, while the protocols in the standard have considered public key validations. The second difference is in sending a masked public key in the corresponding PAKE protocols.

It has been shown [19] that the key agreement protocols in the IEEE 802.15.6 standard are vulnerable to some attacks: The unauthenticated key exchange protocol (Protocol I) is vulnerable to an impersonation attack; the AKE protocol with hidden public key transfer (Protocol II) is vulnerable to a KCI attack; the PAKE protocol (Protocol III) is vulnerable to an impersonation attack and an offline dictionary attack; and the AKE protocol for devices with numerical display (Protocol IV) is vulnerable to an impersonation attack.

All the attacks on Protocols I, II, and IV are applicable to the corresponding protocols in [8], because the protocols are almost the same. However, Ho's PAKE protocol has different vulnerabilities than those of Protocol III in the standard, because the protocols are not the same.

In this paper, we perform a security analysis on Ho's PAKE protocol, and show that it does not provide forward secrecy, although it is argued [8] that the protocol provides perfect forward secrecy. Furthermore, we show that the protocol is vulnerable to an impersonation attack, a KCI attack, and an invalid-curve attack. The impersonation attack on Ho's PAKE protocol is different from the impersonation attack on the corresponding PAKE protocol in the IEEE 802.15.6 standard [19]. By an invalid-curve attack, an adversary is able to extract the private key of another entity. The invalid-curve attack which is presented in this paper on Ho's PAKE protocol, is feasible by an insider adversary. However, it can be shown that any adversary can accomplish a similar invalid-curve attack on Ho's unauthenticated key exchange and numerical display AKE protocols. A variant of the impersonation attack, which is presented in this paper on Ho's PAKE protocol, is also feasible on Ho's AKE protocol with hidden public key transfer. Such extra vulnerabilities are due to not considering public key validations in Ho's protocols.

In this paper, we also perform a security analysis on Unger et al.'s PAKE protocol [25]. The protocol is a variant of the Ho's PAKE protocol, and is proposed for bridging the user interface gap in pervasive computing and smart environments. We show that Unger et al.'s PAKE protocol lacks forward secrecy, and is vulnerable to dictionary and replay attacks. The rest of this paper is organized as follows. We review the protocols in Section 2, and describe their vulnerabilities in Section 3.

2 Review of Two PAKE Protocols

Ho's PAKE protocol [8] and Unger et al.'s PAKE protocol [25] are depicted in Figures 1 and 2, respectively. They use public key cryptography on elliptic curves. The domain parameters consist of an elliptic curve E with cofactor h defined over the finite field $GF(p)$, where $p = q$ or 2^m in which q is a prime number of at least 160 bits, and m is larger than 160. The cofactor h of the elliptic curve is 1, 2 or 4. The base point G in the elliptic curve is of order n where $n \times G = O$ in which O denotes the point at infinity. There are other conditions that should be satisfied by domain parameters of elliptic curves in order to avoid known attacks on elliptic curve-based schemes [22], although they are not mentioned in [8, 25].

The protocols are executed between Alice ($\mathcal{A}$) and Bob ($\mathcal{B}$). $\mathcal{A}$ and $\mathcal{B}$ can be a node and a hub in a WBAN, respectively. $\mathcal{A}$ and $\mathcal{B}$ have self-generated public/private keys. It is specified neither in the IEEE 802.15.6 standard [1] nor in [8] if public keys are accompanied by digital certificates. However, it has been mentioned in [8] that "one of the two parties is likely to be severely constrained by memory, speed, or/and power, and hence cannot store public key certificates or perform digital signature calculations."

The private keys shall be 256-bit random integers, chosen independently from the set of integers $[1, n-1]$. The private key of $\mathcal{A}$ and $\mathcal{B}$ is denoted by SK_A and SK_B, respectively. The corresponding public keys are gen-

Alice ($\mathcal{A}$)		**Bob ($\mathcal{B}$)**
$SK_A \in_R [1, n-1]$		$SK_B \in_R [1, n-1]$
$PK_A = (PK_{AX}, PK_{AY}) = SK_A \times G$		$PK_B = (PK_{BX}, PK_{BY}) = SK_B \times G$
$PK'_A = PK_A - Q(PW)$		
Update SK_A and PK_A if $PK'_A = O$		
Select random 128-bit $Nonce_A$		Select random 128-bit $Nonce_B$
	$\xrightarrow{ID_B \| ID_A \| Nonce_A \| PK'_{AX} \| PK'_{AY} \| Other_A}$	
	$\xleftarrow{ID_A \| ID_B \| Nonce_B \| PK_{BX} \| PK_{BY} \| Other_B}$	
		$PK_A = PK'_A + Q(PW)$
$DH_{Key} = X(SK_A \times PK_B)$		$DH_{Key} = X(SK_B \times PK_A)$
$Temp_1 = RMB_{128}(DH_{Key})$		$Temp_1 = RMB_{128}(DH_{Key})$
$KMAC_{3A} = CMAC(Temp_1, ID_A \| ID_B \| Nonce_A \| Nonce_B \| Other_A, 64)$		$KMAC_{3B} = CMAC(Temp_1, ID_A \| ID_B \| Nonce_A \| Nonce_B \| Other_A, 64)$
$KMAC_{4A} = CMAC(Temp_1, ID_B \| ID_A \| Nonce_B \| Nonce_A \| Other_B, 64)$		$KMAC_{4B} = CMAC(Temp_1, ID_B \| ID_A \| Nonce_B \| Nonce_A \| Other_B, 64)$
	$\xleftarrow{ID_A \| ID_B \| Nonce_B \| PK_{BX} \| PK_{BY} \| Other_B \| KMAC_{3B}}$	
Halt if $KMAC_{3A} \neq KMAC_{3B}$		
	$\xrightarrow{ID_B \| ID_A \| Nonce_A \| PK'_{AX} \| PK'_{AY} \| Other_A \| KMAC_{4A}}$	
		Halt if $KMAC_{4A} \neq KMAC_{4B}$
$Temp_2 = LMB_{128}(DH_{Key})$		$Temp_2 = LMB_{128}(DH_{Key})$
$MK = CMAC(Temp_2, Nonce_A \| Nonce_B, 128)$		$MK = CMAC(Temp_2, Nonce_A \| Nonce_B, 128)$

Figure 1: Ho's PAKE protocol [8]

Alice ($\mathcal{A}$)		**Bob ($\mathcal{B}$)**
$SK_A \in_R [1, n-1]$		$SK_B \in_R [1, n-1]$
$PK_A = (PK_{AX}, PK_{AY}) = SK_A \times G$		$PK_B = (PK_{BX}, PK_{BY}) = SK_B \times G$
$PK'_A = PK_A - Q(PW)$		
Select random $Nonce_A$		
	$\xrightarrow{PK'_A, Nonce_A, ID_A, ID_B}$	
		Select random $Nonce_B$
		$PK_A = PK'_A + Q(PW)$
		$S = SK_B \times PK_A$
		$H_B = CMAC(PW, Other_B \| PK_B)$
	$\xleftarrow{PK_B, Nonce_B, ID_A, ID_B, H_B}$	
$S = SK_A \times PK_B$		
$H'_B = CMAC(PW, Other_B \| PK_B)$		
Verify if $H_B = H'_B$		
$H_A = CMAC(PW, Other_A \| PK_A)$		
$MK = CMAC(S, Nonce_A \| Nonce_B)$		
	$\xrightarrow{H_A}$	
		$H'_A = CMAC(PW, Other_A \| PK_A)$
		Verify if $H_A = H'_A$
		$MK = CMAC(S, Nonce_A \| Nonce_B)$

Figure 2: Unger et al.'s PAKE protocol [25]

erated as $PK_A = (PK_{AX}, PK_{AY}) = SK_A \times G$, and $PK_B = (PK_{BX}, PK_{BY}) = SK_B \times G$. PK_A and PK_B are points on E, and have X-coordinate and Y-coordinate values. The lifecycle of private/public keys are not specified in [8]. Although the key generation is depicted on the top of Figure 1, it does not mean that the key generation should repeat for each protocol execution. It is argued in [8] that all the proposed AKE and PAKE protocols provide the perfect forward secrecy. Reasoning for forward secrecy of the AKE protocols means that private/public keys are not random numbers used in a typical Diffie-Hellman key agreement. The forward secrecy makes sense if there is a static secret value. Furthermore, the private keys are specifically differentiated from nonces in the protocols.

$\mathcal{A}$ and $\mathcal{B}$ are assumed to have a shared password in advance. During protocol executions, $\mathcal{B}$ sends his public key PK_B in clear, but $\mathcal{A}$ sends a password-scrambled public key PK'_A that is masked by a hash of password as $PK'_A = PK_A - Q(PW)$ in which PW is a positive integer, converted through a character encoding from the pre-shared password between $\mathcal{A}$ and $\mathcal{B}$ such that $0 \leq PW < p$. The $Q(.)$ function is a mapping which converts the integer PW to the point $Q(PW) = (Q_X, Q_Y)$ on the elliptic curve in which $Q_X = 2^{32} \times 2h \times PW + M_X$ where M_X is the smallest nonnegative integer such that Q_X becomes the X-coordinate of a point on the elliptic curve. Q_Y is an even positive integer, and is the Y-coordinate of that point. $\mathcal{A}$ shall choose a private key SK_A such that the X-coordinate of PK_A is not equal to the X-coordinate of $Q(PW)$, i.e. we have $PK'_A \neq O$.

$CMAC(K, M, L)$ represents the L-bit output of the *Cipher-based Message Authentication Code* (CMAC), applied under key K to message M. $LMB_L(S)$ and $RMB_L(S)$ designates the L leftmost and the L rightmost bits of the bit string S, respectively. $X(P)$ denotes the X-coordinate of point P on the elliptic curve, i.e. $X(P) = X(P_X, P_Y) = P_X$. The sign $||$ denotes concatenation of bit strings. ID_A and ID_B may be MAC address, IP address, and so on. $Other_A$ and $Other_B$ denotes other public parameters of $\mathcal{A}$ and $\mathcal{B}$, respectively.

3 Security Analysis

In this section, we show that Ho's [8] and Unger et al.'s [25] PAKE protocols that are depicted in Figures 1 and 2, are vulnerable to different attacks. In the rest of this paper, $\mathcal{M}$ denotes an active adversary, and $\mathcal{E}$ denotes a passive adversary.

3.1 Security Problems of Ho's PAKE Protocol

It is argued that the Ho's PAKE protocol provides perfect forward secrecy, and is resilient to impersonation and dictionary attacks [8]. However, we show that the protocol lacks the forward secrecy, and is vulnerable to an impersonation attack, a KCI attack, and an invalid-curve attack.

3.1.1 Impersonation Attack

As mentioned in Section 2, public keys are self-generated by involved parties. It is more likely that public keys are not accompanied by digital certificates due to resource constraints on nodes. As neither $\mathcal{A}$ nor $\mathcal{B}$ checks the validity of the received public key,

- For impersonating $\mathcal{A}$, $\mathcal{M}$ can simply send O as the masked public key of $\mathcal{A}$. If $PK'_A = O$, then $DH_{Key} = O$.
- For impersonating $\mathcal{B}$, $\mathcal{M}$ can simply send O as the public key of $\mathcal{B}$. If $PK_B = O$, then $DH_{Key} = O$.

Based on the encoding used for representation of O, $Temp_1$ and $Temp_2$ will have a known value. The only secret information in calculation of $KMAC_{3A}$, $KMAC_{4A}$, $KMAC_{3B}$, and $KMAC_{4B}$ is $Temp_1$. As $Temp_1$ will have a known value, $\mathcal{M}$ can calculate $KMAC_{3A} = KMAC_{3B}$ and $KMAC_{4A} = KMAC_{4B}$, and bypass the authentication. The only secret information in calculation of the master key MK is $Temp_2$. As $Temp_2$ will have a known value, $\mathcal{M}$ can calculate MK. Then, $\mathcal{M}$ can successfully impersonate either $\mathcal{A}$ or $\mathcal{B}$.

Validation of a public key $PK = (PK_X, PK_Y)$ includes checking the following conditions [6, 15]:

1) $PK \neq O$,
2) $PK_X, PK_Y \in GF(p)$,
3) PK_X, PK_Y should satisfy the defining equation of curve E,
4) $h \times PK \neq O$ where h denotes the cofactor of E.

3.1.2 Key Compromise Impersonation Attack

The KCI attack is a weaker variant of the impersonation attack in terms of requiring knowledge of a private key for

kind of impersonation. The KCI attack is according to a stronger notion of security which has been considered in the eCK security model [10] for AKE protocols. If the private key of an entity $\mathcal{A}$ is compromised, an adversary $\mathcal{M}$ can impersonate $\mathcal{A}$ in one-factor authentication protocols. However, such compromise should not enable $\mathcal{M}$ to impersonate other honest entities in communication with $\mathcal{A}$. Resistance to the KCI attack is an important security attribute which prevents an adversary from actively controlling a compromised entity [23]. The KCI attack makes sense for PAKE protocols if they use public keys.

As Ho's PAKE protocol is vulnerable to an impersonation attack, one would consider discussion on the KCI attack redundant, because the KCI attack has an extra requirement for compromise of a private key. However, discussion on the KCI attack is noteworthy, because the impersonation attack on the protocol could be prevented by adding validation of public keys to the protocol. However, the KCI attack will be feasible even after adding public key validation or having certified public keys from a lightweight PKI [14, 21]. Here is the attack scenario in which $\mathcal{M}$ has SK_A, and impersonates $\mathcal{B}$. $\mathcal{M}$ does not need to have the password PW. As the public key of $\mathcal{B}$ is sent in clear, we can assume that $\mathcal{M}$ has obtained PK_B by eavesdropping a previous protocol run.

- $\mathcal{A}$ selects a 128-bit random number $Nonce_A$, and sends $\{ID_B \mid\mid ID_A \mid\mid Nonce_A \mid\mid PK'_{AX} \mid\mid PK'_{AY} \mid\mid Other_A\}$ to $\mathcal{B}$. $\mathcal{M}$ hijacks the session, and tries to impersonate $\mathcal{B}$.

- $\mathcal{M}$ selects a random number $Nonce_M$, and sends $\{ID_A||ID_B||Nonce_M||PK_{BX}||PK_{BY}||Other_B\}$ to $\mathcal{A}$.

- $\mathcal{M}$ has SK_A. $\mathcal{M}$ computes $DH_{Key} = X(SK_A \times PK_B)$, $Temp_1 = RMB_{128}(DH_{Key})$, $KMAC_{3B} = CMAC(Temp_1, ID_A \mid\mid ID_B \mid\mid Nonce_A \mid\mid Nonce_M \mid\mid Other_A, 64)$, and $KMAC_{4B} = CMAC(Temp_1, ID_B \mid\mid ID_A \mid\mid Nonce_M \mid\mid Nonce_A \mid\mid Other_B, 64)$. $\mathcal{M}$ sends $\{ID_A \mid\mid ID_B \mid\mid Nonce_M \mid\mid PK_{BX} \mid\mid PK_{BY} \mid\mid Other_B \mid\mid KMAC_{3B}\}$ to $\mathcal{A}$.

- $\mathcal{A}$ computes $DH_{Key} = X(SK_A \times PK_B)$, $Temp_1 = RMB_{128}(DH_{Key})$, and $KMAC_{3A} = CMAC(Temp_1, ID_A \mid\mid ID_B \mid\mid Nonce_A \mid\mid Nonce_M \mid\mid Other_A, 64)$. $\mathcal{A}$ verifies that $KMAC_{3A} = KMAC_{3B}$, and computes $KMAC_{4A} = CMAC(Temp_1, ID_B \mid\mid ID_A \mid\mid Nonce_M \mid\mid Nonce_A \mid\mid Other_B, 64)$. $\mathcal{A}$ sends $\{ID_B \mid\mid ID_A \mid\mid Nonce_A \mid\mid PK'_{AX} \mid\mid PK'_{AY} \mid\mid Other_A \mid\mid KMAC_{4A}\}$ to $\mathcal{M}$.

- $\mathcal{A}$ computes $Temp_2 = LMB_{128}(DH_{Key})$, and generates the master key $MK = CMAC(Temp_2, Nonce_A \mid\mid Nonce_M, 128)$.

- $\mathcal{M}$ computes $Temp_2 = LMB_{128}(DH_{Key})$, and generates the master key $MK = CMAC(Temp_2, Nonce_A \mid\mid Nonce_M, 128)$.

$\mathcal{M}$ and $\mathcal{A}$ compute the same MK. $\mathcal{M}$ could successfully impersonate $\mathcal{B}$.

3.1.3 Invalid-curve Attack

In Ho's protocols, neither $\mathcal{A}$ nor $\mathcal{B}$ consider validation of public keys, received from the other party. Validation of static and ephemeral public keys is very important in elliptic curve cryptography. An invalid-curve attack would be feasible if an EC-based protocol does not consider validation of static or ephemeral public keys [6, 20]. By an invalid-curve attack, an attacker may extract the private key of another entity [24].

In [8], the elliptic curve is defined over $GF(p)$ where $p = q$ or 2^m. For an elliptic curve defined over a finite field $GF(q)$ of prime order $q > 3$, the Weierstrass equation is $y^2 = x^3 + ax + b$ where $a, b \in GF(q)$. For non-singularity, we require that $4a^3 + 27b^2 \neq 0 \mod q$. For the binary finite fields $GF(2^m)$, the Weierstrass equation is $y^2 + xy = x^3 + ax^2 + b$ where $a, b \in GF(2^m)$ with $b \neq 0$. There is another kind of Weierstrass equation over $GF(2^m)$ which gives supersingular curves, but they are cryptographically weak. If G, the base point of the elliptic curve, is of order n, then h the cofactor of the elliptic curve is defined as $h = \#E(GF(p))/n$ in which $\#E(.)$ is called the order of the elliptic curve E, and it denotes the number of points on the elliptic curve (including O).

The idea behind an invalid-curve attack is that for two elliptic curves over $GF(q)$ (or two curves over $GF(2^m)$) whose defining equations have the same a coefficient but different b coefficients, the addition formulaes are the same, and they do not involve the coefficient b. For the general case, let $y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6$ be the generalized Weierstrass equation of an elliptic curve E defined over the finite field $GF(p)$ where $p = q$ or 2^m. An *invalid-curve* (relative to E) is an elliptic curve E' defined over $GF(p)$ with the Weierstrass equation $y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a'_6$ where $a_6 \neq a'_6$. Note that $E(GF(p)) \cap E'(GF(p)) = O$.

If $PK_M \in E'(GF(p))$ and $PK_M \neq O$, then there is not any private key SK_M such that $PK_M = SK_M \times G$. The addition formulaes on E and E' does not involve a_5

and a'_5 coefficient, respectively. Let $PK_M \in E'(GF(p))$, and SK_B be the private key of $\mathcal{B}$. If $\mathcal{B}$ uses the addition formulae for E in any point multiplication algorithm for computing $K = SK_B \times PK_M$, then $\mathcal{B}$ will indeed obtain a point on E', and we have $K \in E'(GF(p))$. If PK_M is a point of a small order, and $\mathcal{M}$ receives a feedback from $\mathcal{B}$ which includes some calculations based on K, this would be used for finding SK_B, the private key of $\mathcal{B}$.

For Ho's PAKE protocol, the invalid-curve attack can be accomplished by an insider adversary that has knowledge of the shared password PW. However, the same attack can be done by any adversary on Ho's unauthenticated key exchange protocol (Protocol I) and display AKE protocol (Protocol IV). Here is the attack scenario on Ho's PAKE protocol in which $\mathcal{A}$ performs an invalid-curve attack against $\mathcal{B}$, and finds SK_B:

- $\mathcal{A}$ selects an invalid curve E' such that $E'(GF(p))$ contains a point $PK_{Ai} = (PK_{AiX}, PK_{AiY})$ of small order t_i. $\mathcal{A}$ computes $PK'_{Ai} = (PK'_{AiX}, PK'_{AiY}) = PK_{Ai} - Q(PW)$, selects a random number $Nonce_A$, and sends $\{ID_B \,||\, ID_A \,||\, Nonce_A \,||\, PK'_{AiX} \,||\, PK'_{AiY} \,||\, Other_A\}$ to $\mathcal{B}$. Note that PK'_{Ai} most likely resides on neither E nor E'.

- $\mathcal{B}$ selects a 128-bit random number $Nonce_B$, and sends $\{ID_A \,||\, ID_B \,||\, Nonce_B \,||\, PK_{BX} \,||\, PK_{BY} \,||\, Other_B\}$ to $\mathcal{A}$.

- $\mathcal{B}$ computes $PK_{Ai} = PK'_{Ai} + Q(PW)$, $DH_{Key} = X(SK_B \times PK_{Ai})$, $Temp_1 = RMB_{128}(DH_{Key})$, $KMAC_{3B} = CMAC(Temp_1, ID_A \,||\, ID_B \,||\, Nonce_A \,||\, Nonce_B \,||\, Other_A,\ 64)$, and $KMAC_{4B} = CMAC(Temp_1,\ ID_B \,||\, ID_A \,||\, Nonce_B \,||\, Nonce_A \,||\, Other_B,\ 64)$. $\mathcal{B}$ sends $\{ID_A \,||\, ID_B \,||\, Nonce_B \,||\, PK_{BX} \,||\, PK_{BY} \,||\, Other_B \,||\, KMAC_{3B}\}$ to $\mathcal{A}$.

- $\mathcal{A}$ receives $KMAC_{3B}$, and halts the protocol execution. There are t_i possible values for $SK_B \times PK_{Ai}$ because PK_{Ai} is of order t_i. Then, there are $t_i/2$ possible values for $DH_{Key} = X(SK_B \times PK_{Ai})$ and $Temp_1 = RMB_{128}(DH_{Key})$. $\mathcal{A}$ tries all possible values of $Temp_1$ in $KMAC_{3A} = CMAC(Temp_1, ID_A \,||\, ID_B \,||\, Nonce_A \,||\, Nonce_B \,||\, Other_A,\ 64)$ until she finds a value for $Temp_1$ for which $KMAC_{3A} = KMAC_{3B}$. Then, with at most $t_i/2$ trials, $\mathcal{A}$ finds an equation $d_i^2 \equiv SK_B^2 \bmod t_i$ in which t_i and d_i are known, and SK_B is unknown.

$\mathcal{A}$ repeats the above attack for different points PK_{Ai} of pairwise relatively prime order t_i, i.e. we should have $gcd(t_i, t_j) = 1$, $\forall t_i \neq t_j$. Such points can be selected from the same or different invalid curves. Using the *Chinese remainder theorem*, $\mathcal{A}$ combines the mentioned equations, and finds $SK_B^2 \equiv d \mod N$ for some $N > n^2$. Since $SK_B^2 < n^2 < N$, we have $d = SK_B^2$, and $\mathcal{A}$ computes $SK_B = \sqrt{d}$. $\mathcal{A}$ finds the private key of $\mathcal{B}$, while $\mathcal{B}$ is unaware that such an attack has taken place.

3.1.4 Lack of Forward Secrecy

Forward secrecy is an important security attribute in key exchange protocols. If an entity's private key has been compromised, it should not affect the security of session keys that have been established before the compromise. The notion of *perfect forward secrecy* (PFS) is a bit stronger than the forward secrecy. PFS means that the established session keys should remain secure even after compromising the private keys of all the entities that are involved in the protocol. For public key-based AKE protocols, the forward secrecy is defined with respect to compromise of the private key. For PAKE protocols, the forward secrecy is defined with respect to compromise of the password. For PAKE protocols that use both public keys and passwords, the forward secrecy can be defined according to compromise of either a private key or a password.

In [8], it is argued for the PFS. However, we show that the protocol provides neither PFS nor forward secrecy. As PK_B, $Nonce_A$ and $Nonce_B$ are sent in clear, we can assume that they are eavesdropped and saved by $\mathcal{E}$. If SK_A is compromised, $\mathcal{E}$ computes $DH_{Key} = X(SK_A \times PK_B)$, $Temp_2 = LMB_{128}(DH_{Key})$, and obtains the master key $MK = CMAC(Temp_2, Nonce_A || Nonce_B, 128)$. Then, the protocol does not provide the forward secrecy.

3.2 Security Problems of Unger et al.'s PAKE Protocol

In this section, we show that Unger et al.'s PAKE protocol lacks the forward secrecy, and is vulnerable to dictionary and replay attacks.

3.2.1 Dictionary Attacks

It is crucial for PAKE protocols to be resilient to dictionary attacks. A PAKE protocol should not provide an adversary with a verifier which can be used for guessing the password. This is not the case for Unger et al.'s protocol. For an offline dictionary attack, it is sufficient for an adversary $\mathcal{E}$ to eavesdrop on messages exchanged between $\mathcal{A}$ and $\mathcal{B}$ in a protocol run. $\mathcal{E}$ then obtains H_B and PK_B that are sent

in clear. As values of $H_B = CMAC(PW, Other_B||PK_B)$, PK_B and $Other_B$ are known, it can be used as a verifier in a password guessing attack. $\mathcal{E}$ can try most probable passwords from a dictionary of passwords in the verifier.

Alternatively, instead of eavesdropping on a protocol execution, an adversary may interact with an entity, and obtain the verifier. Here is a scenario in which an adversary $\mathcal{M}$ impersonates $\mathcal{A}$, and obtains a verifier which can be used for finding the password:

- $\mathcal{M}$ selects random numbers SK_M and $Nonce_M$. $\mathcal{M}$ computes $PK''_M = SK_M \times G$, and sends $\{PK''_M, Nonce_M, ID_A, ID_B\}$ to $\mathcal{B}$.
- $\mathcal{B}$ selects a random number $Nonce_B$, computes $PK_M = PK''_M + Q(PW)$, $S = SK_B \times PK_M$, and $H_B = CMAC(PW, Other_B||PK_B)$. $\mathcal{B}$ sends $\{PK_B, Nonce_B, ID_A, ID_B, H_B\}$ to $\mathcal{M}$.
- $\mathcal{M}$ gets H_B, and halts the protocol execution. $\mathcal{M}$ performs an offline dictionary attack to find a password which satisfies $H_B = CMAC(PW, Other_B||PK_B)$ in which H_B, PK_B and $Other_B$ are known. $\mathcal{B}$ does not detect any attack or suspicious activity.

3.2.2 Replay Attack

Unger et al.'s protocol is vulnerable to a replay attack. In the protocol, authentication of $\mathcal{A}$ and $\mathcal{B}$ is done through $H_A = CMAC(PW, Other_A||PK_A)$ and $H_B = CMAC(PW, Other_B||PK_B)$, respectively. H_A and H_B does not contain any fresh information, and they will be the same in all protocol executions between $\mathcal{A}$ and $\mathcal{B}$, as long as they do not change their public keys or passwords. For the resource-constrained situation that has been considered in [25], $\mathcal{A}$ and $\mathcal{B}$ may perform private/public key generation once. Even if they change their public key, $\mathcal{B}$ always sends his public key PK_B in clear. $\mathcal{A}$ sends a password-scrambled public key PK'_A. If the values sent for PK'_A are different in different protocol runs, it notifies a change in PK_A or PW. Otherwise, it means that they are more likely unchanged which indicates feasibility of a replay attack. Of course, $\mathcal{M}$ cannot establish a new master key MK through a replay attack, but can bypass the authentication.

3.2.3 Lack of Forward Secrecy

As PK_B, $Nonce_A$ and $Nonce_B$ are sent in clear, we can assume that they are eavesdropped and saved by $\mathcal{E}$. If SK_A is compromised, $\mathcal{E}$ computes $S = X(SK_A \times PK_B)$, and obtains the master key $MK = CMAC(S, Nonce_A||Nonce_B)$. Then, the protocol does not provide the forward secrecy.

4 Conclusion

In this paper, we performed a security analysis on Ho's PAKE protocol [8] and Unger et al.'s PAKE protocol [25] that are proposed for body area networks and smart environments, respectively. Both protocols use elliptic curve cryptography. We showed that Ho's PAKE protocol lacks the forward secrecy and is vulnerable to impersonation, KCI and invalid-curve attacks. Furthermore, we showed that Unger et al.'s protocol lacks the forward secrecy, and is vulnerable to dictionary and replay attacks. The invalid-curve attack, which is presented in this paper on Ho's PAKE protocol, is feasible by an insider adversary where the adversary can extract the private key of another participant. However, it can be shown that any adversary can accomplish a similar invalid-curve attack on Ho's unauthenticated key exchange and numerical display AKE protocols [8]. A variant of the impersonation attack, which is presented in this paper on Ho's PAKE protocol, is also feasible on Ho's AKE protocol with hidden public key transfer [8]. Such extra vulnerabilities are due to not considering public key validations in Ho's protocols.

Acknowledgment

The author would like to thank anonymous reviewers for their comments.

References

[1] The IEEE Standards Association, "IEEE P802.15.6 Standard for Wireless Body Area Networks," 2012.

[2] S. M. Bellovin and M. Merritt, "Encrypted key exchange: Password-based protocols secure against dictionary attacks," in *Proceedings of the IEEE Symposium on Security and Privacy*, pp. 72–84, 1992.

[3] Q. Cheng, "Cryptanalysis of a new efficient authenticated multiple-key exchange protocol from bilinear pairings," *International Journal of Network Security*, vol. 16, no. 6, pp. 494–497, 2014.

[4] Y. Ding and P. Horster, "Undetectable on-line password guessing attacks," *ACM SIGOPS Operating Systems Review*, vol. 29, pp. 77–86, Oct. 1995.

[5] R. Dutta and R. Barua, "Password-based encrypted group key agreement," *International Journal of Network Security*, vol. 3, no. 1, pp. 23–34, 2006.

[6] D. Hankerson, S. Vanstone, and A. J. Menezes, *Guide to elliptic curve cryptography*, Springer, 2004.

[7] D. He, Y. Zhang, and J. Chen, "Cryptanalysis of a three-party password-based authenticated key exchange protocol," *International Journal of Network Security*, vol. 16, no. 5, pp. 393–396, 2014.

[8] J. M. Ho, "A versatile suite of strong authenticated key agreement protocols for body area networks," in *IEEE 8th International Conference on Wireless Communications and Mobile Computing (IWCMC'12)*, pp. 683–688, 2012.

[9] H. Krawczyk, "HMQV: A high-performance secure Diffie-Hellman protocol," in *Advances in Cryptology (CRYPTO'05)*, pp. 546–566, 2005.

[10] B. LaMacchia, K. Lauter, and A. Mityagin, "Stronger security of authenticated key exchange," in *Provable Security*, pp. 1–16, 2007.

[11] C. C. Lee, S. T. Chiu, and C. T. Li, "Improving security of a communication-efficient three-party password authentication key exchange protocol," *International Journal of Network Security*, vol. 17, no. 1, pp. 1–6, 2015.

[12] C. C. Lee, C. H. Liu, and M. S. Hwang, "Guessing attacks on strong-password authentication protocol," *International Journal of Network Security*, vol. 15, no. 1, pp. 64–67, 2013.

[13] M. Mana, M. Feham, and B. A. Bensaber, "Trust key management scheme for wireless body area networks," *International Journal of Network Security*, vol. 12, no. 2, pp. 75–83, 2011.

[14] S. Misra, S. Goswami, C. Taneja, and A. Mukherjee, "Design and implementation analysis of a public key infrastructure-enabled security framework for ZigBee sensor networks," *International Journal of Communication Systems*, Article first published online: 10 NOV 2014.

[15] M. Toorani, "SMEmail - a new protocol for the secure e-mail in mobile environments," in *Proceedings of the IEEE Australian Telecommunications Networks and Applications Conference (ATNAC'08)*, pp. 39–44, 2008.

[16] M. Toorani, "Cryptanalysis of a new protocol of wide use for email with perfect forward secrecy," *Security and Communication Networks*, vol. 8, no. 4, pp. 694–701, 2015.

[17] M. Toorani, "Cryptanalysis of a protocol from FC'10," in *Proceedings of Financial Cryptography and Data Security*, Jan. 2015.

[18] M. Toorani, "On continuous after-the-fact leakage-resilient key exchange," in *Proceedings of the Second ACM Workshop on Cryptography and Security in Computing Systems (CS2'15)*, pp. 31–34, 2015.

[19] M. Toorani, "On vulnerabilities of the security association in the IEEE 802.15.6 standard," in *Proceedings of Financial Cryptography and Data Security Workshops – 1st Workshop on Wearable Security and Privacy (Wearable'15)*, Jan. 2015.

[20] M. Toorani and A. Beheshti, "Cryptanalysis of an efficient signcryption scheme with forward secrecy based on elliptic curve," in *Proceedings of 2008 IEEE International Conference on Computer and Electrical Engineering (ICCEE'08)*, pp. 428–432, 2008.

[21] M. Toorani and A. Beheshti, "LPKI - a lightweight public key infrastructure for the mobile environments," in *Proceedings of the 11th IEEE International Conference on Communication Systems (ICCS'08)*, pp. 162–166, Nov. 2008.

[22] M. Toorani and A. Beheshti, "A directly public verifiable signcryption scheme based on elliptic curves," in *Proceedings of the 14th IEEE Symposium on Computers and Communications (ISCC'09)*, pp. 713–716, 2009.

[23] M. Toorani and A. Beheshti, "An elliptic curve-based signcryption scheme with forward secrecy," *Journal of Applied Sciences*, vol. 9, no. 6, pp. 1025–1035, 2009.

[24] M. Toorani and A. Beheshti, "Cryptanalysis of an elliptic curve-based signcryption scheme," *International Journal of Network Security*, vol. 10, no. 1, pp. 51–56, 2010.

[25] S. Unger and D. Timmermann, "Bridging the UI gap for authentication in smart environments," in *Proceedings of the 19th IEEE Symposium on Computers and Communications (ISCC'14)*, pp. 1–6, July 2014.

Mohsen Toorani received the B.S. degree in Communications Engineering and M.S. degree in Secure Communications both from Iran University of Science and Technology in 2005 and 2008, respectively. He is a PhD candidate at Department of Informatics, University of Bergen. His research interests include cryptology and information security.

A Component Histogram Map Based Text Similarity Detection Algorithm

Huajun Huang, Shuang Pang, Qiong Deng, and Jiaohua Qin
(Corresponding author: Huajun Huang)

College of Computer and Information Engineering, Central South University of Forestry and Technology
498 Shaoshan South Road, CHangsha, Hunan Province 410004, China
(Email: hhj0906@163.com)

(Received Apr. 10, 2015; revised and accepted May 16 & May 24, 2015)

Abstract

The conventional text similarity detection usually use word frequency vectors to represent texts. But it is high-dimensional and sparse. So in this research, a new text similarity detection algorithm using component histogram map (CHM-TSD) is proposed.This method is based on the mathematical expression of Chinese characters, with which Chinese characters can be split into components. Then each components occurrence frequency will be counted for building the component histogram map (CHM) in a text as text characteristic vector. Four distance formulas are used to find which the best distance formula in text similarity detection is. The experiment results indicate that CHM-TSD achieves a better precision, recall and F1 than cosine theorem and Jaccard coefficient.

Keywords: Component histogram map, distance calculation, text similarity detection

1 Introduction

As a branch of natural language processing, text similarity detection is more and more important for information security. It has been used in many fields such as information retrieval (IR), duplicated detection, Data clustering and classification [3]. In general, there are two ways for text similarity detection, one is that based on semantic similarity, and the other one is non-semantic. Semantic similarity detection usually based on dictionary computation like HowNet [13] and WordNet [4]. Huang has ever proposed a method that combined the external dictionary with TF-IDF to compute text similarity [5]. Some people also use a large-scale corpus for semantic similarity detection [7], but its uncommon because of its disadvantages. Non-semantic similarity detection mostly uses word frequency statistics and string comparison two methods. The most common used methods of word frequency statistics are VSM [11, 12] the text similarity can be computed through cosine [14] theorem or Jaccard coefficient [10]. In the other hand, Shingling [15] and maximum string matching algorithm [6] is often used for string comparison. All of the methods above performance well in certain situations, but there are also some shortcomings. For examples, the semantic method based on dictionary is too depending on person and the knowledge library to express the sense of a word exactly. Word frequency statistics is very high-dimensional and sparse [8].

From the above, a new Chinese text similarity detection method was proposed. This method used CHM (component histogram map) to avoid high-dimensional and sparse problem. Mathematical expression of Chinese characters [9], used to split Chinese characters into components was the basic theorem for this method. And the components were taken as research object. Components are correlated with each other to compose Chinese characters, so these components are correlative. CHM was built with each components occurrence frequency. Then the distance between text and duplicate text is calculated with Bhattacharyya formula. From the results, we can see that CHM-TSD performance better than cosine theorem and Jaccard coefficient.

2 Related Theories

In the process of text duplicate detection, text feature representation and similarity detection are two very important steps [12]. VSM is the most common method for text feature representation. Assuring d_i is the i-th text, W_{ij} is the weight of the j-th word of d_i, then the i-th text can be represented as $\vec{d_i} = (W_{i,1}, W_{i,2}, \cdots, W_{i,3})$, so all the texts in the experiment can compose a vector space $D = (\vec{d_1}, \vec{d_2}, \cdots, \vec{d_n})$. The similarity of each pair of text can be computed as two vectors distance through cosine

theorem. The formula is as follows:

$$\begin{aligned} sim(d_i, d_j) &= \cos(\theta) \\ &= \frac{\vec{d_i} \cdot \vec{d_j}}{\left\|\vec{d_i}\right\| \times \left\|\vec{d_j}\right\|} \\ &= \frac{\sum_{t=1}^{n} W_{i,t} \times W_{j,t}}{\sqrt{\sum_{t=1}^{n} W_{i,t}^2} \times \sqrt{\sum_{t=1}^{n} W_{j,t}^2}}, \end{aligned}$$

where $\left\|\vec{d_j}\right\|$ is norm of d_i. The value of cosine similarity between two vectors is between 0 and 1, 0 indicates the two texts are different and 1 indicates they are the same.

In the Chinese character library, there are 6763 common Chinese characters which encoded with Gb-2312. And all these Chinese characters can be combined to thousands of words and even more. For example, the word segmentation software of Chinese Academy of Science (ICTCLAS), has extracted 130,000 commonly used words from the corpus provided by Sogo lab [8]. So it is obvious that the number of Chinese word in a corpus needs to be counted is quite big. This leads to high-dimensional and sparse vectors space. Therefore, a new text representation method based on component relation map has been proposed.

A mathematical expression of Chinese characters of a Chinese character is a formula for splitting Chinese characters into components. It composes of operators and components. Component is a part of a Chinese character and composes of strokes. Every component has a corresponding number as its identifier. There are two kinds of components, one is the ordinary components, and the other called composed components consist of two or more components by certain structural rules. As shown in Figure 1. Chinese characters are formed with the components by different structural rules [9].

There are six operators of the mathematical expression of Chinese characters, $lr(left\ right)$, $ud(up\ down)$, $we(whole\ embody)$, $lu(left\ up)$, $ld(left\ down)$, $ru(right\ up)$. All these operators represent the combination mode of components. As shown in Figure 1, the rectangle A and B are components [9]. $A\ lr\ B$ means that A is on the left and B is on the right. It has two results, a composed component and a Chinese character.

As mentioned above, Chinese characters are compose of components and correlated rules. In this research, we have selected 505 components which can form all the common used Chinese characters as research objects. As Chinese characters increasing, the number of each component will increase clearly, but the number of kinds of components won't. Figure 2 gives some examples of mathematical expression of Chinese characters.

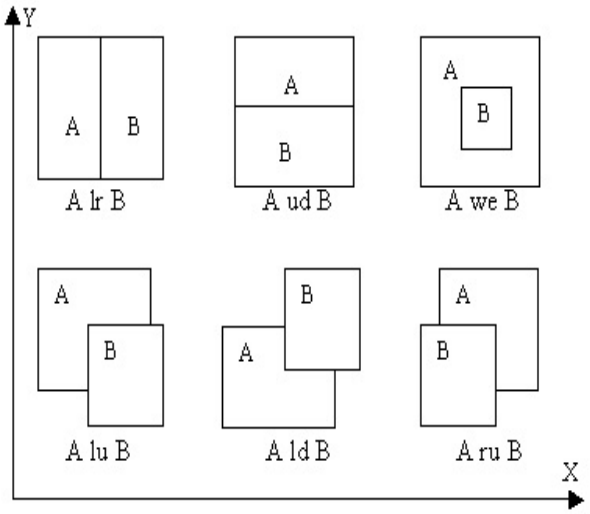


Figure 1: Intuitive description of the operators

Chinese Characters	Mathematical Expression
膘	124 lr(203 ud 142)
彬	86 lr 86 lr 435
勰	447 lr(5 ud 303 ud 67)lr 16
丙	1 ud 100
蚕	95 ud 209

Figure 2: Mathematical expression of Chinese characters

3 Text Duplicate Detection Model

3.1 Detection Model

Text duplicate detection model divides into three modules: 1)text preprocessing. 2) build the component histogram map. 3) Calculate the distance between text in database and detected text. The framework of this model is as shown in Figure 3. When two texts are prepared, the number, English characters, and the stop and useless words are deleted first. So there are only Chinese characters retained. After the preprocessing, all Chinese characters in texts are split into components through the mathematical expression of Chinese characters. Then the occurrence frequency of each component will be counted for building the histogram maps. At last, all the component histogram maps of each pair of texts are matched to get the text similarity. The core module of this model is

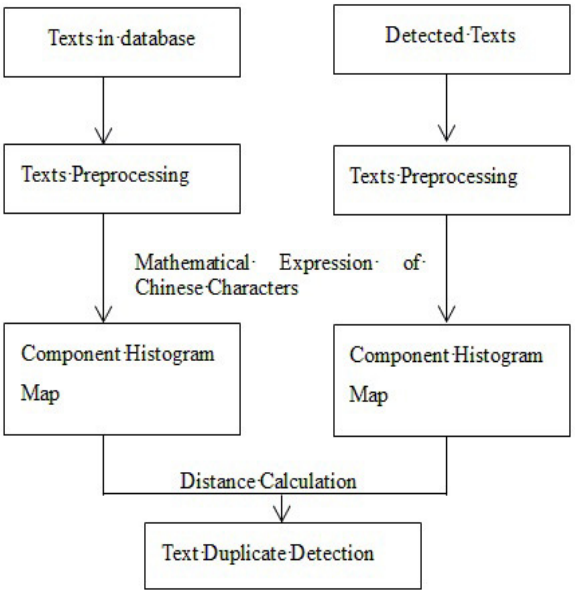


Figure 3: The text duplicate detection model

text duplicate detection using component histogram map.

3.2 Component Histogram map

Rule 1. Assuring that c denotes component and T is a text, then c_i is the ith component and the text T can be regarded as a set of components, so $T = \{c_1, c_2, c_3, \cdots, c_i, \cdots, c_n\}$.

Rule 2. T is the preprocessing text, W is a set of words appearing in text T. Ω is a basic component. w_i is an element of W. $t(c, w_i)$ is the number of c appear in word w_i. $N(c)$ denotes component c appear in text T:

$$N(c) = \sum_{w_i \in \Omega} N(w_i) \times t(c, w_i).$$

Definition 1. *Component histogram map is defined as* $H = \{f_{c_1}, f_{c_2}, \cdots, f_{c_n}\}$, *where* f_{c_i} *is the frequency of the* i*th component in text* T. *Figure 4 is a CHM of the text extracted from experiment data corpus.* f_{c_i} *is counted in the following.*

$$f(c_i) = \frac{N(C_i)}{\sum_{i=1}^{n} N(c_i)}.$$

From the definition of component histogram map, there are some properties.

Property 1. A component histogram map only reflects components frequency in a text. The location of a component appear in the text do not depict in the map.

Property 2. The mapping relation between component histogram map and text is many to one. A text only has a component histogram map, but different texts may have the same component histogram map.

Property 3. A sub component histogram map in a text can into the whole map.

As the properties are shown in the former section, this method may bring a false negative, which a text is not a duplicated text, but it is detected as a duplicated one.

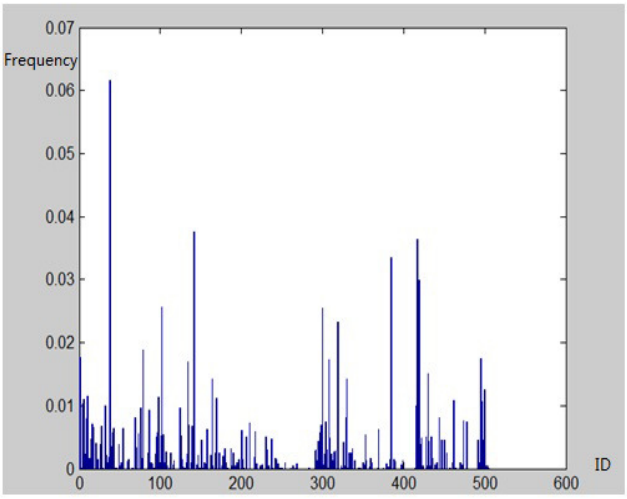


Figure 4: Component histogram map

3.3 Distance Calculated

Text feature representation and similarity detection are two very important steps in the process of text duplicate detection. We use the component histogram map as feature representation. Now, we take the similarity detection into account. To two feature vectors, the common method is distance calculation between the two vectors. So we choose distance calculation to measure the similarity between the two texts.

Assuming text $T1$ and text $T2$, the component histogram map is denote $H1$ and $H2$ by each. So the distance between $T1$ and $T2$ can be denoted as follows.

$$D = Dis(H1, H2).$$

If the value of D is equal to 0, the texts are complete similarity. If the value of D is equal to 1, the texts are completely different. Others may use a threshold α to determine the texts are belonging to. In order to reduce the false positive and false negative, we select fours distance calculation formulas, Correlation, Chi-Square, Intersection, and Bhattacharyya. In the following section, we will show which is the best distance calculation formula used in our method. The four formulas are shown as follows.

Correlation:

$$D(H_1, H_2) = 1 - \frac{\sum_{i=1}^{n}(H_1(i)-\bar{H_1})(H_2(i)-\bar{H_2})}{\sqrt{\sum_{i=1}^{n}(H_1(i)-\bar{H_1})^2)\sum_{i=1}^{n}(H_2(i)-\bar{H_2})^2)}}.$$

Chi-Square:

$$D(H_1, H_2) = \sum_{i=1}^{n}\frac{(H_1(i)-H_2(i))^2}{H_1(i)+H_2(i)}.$$

Intersection:

$$D(H_1, H_2) = 1 - \frac{\sum_{i=1}^{n} min[H_1(i), H_2(i)]}{\sum_{i=1}^{n} H_1(i)}.$$

Bhattacharyya:

$$D(H_1, H_2) = \sqrt{1 - \frac{\sum_{i=1}^{n}\sqrt{H_1(i)\cdot H_2(i)}}{\sqrt[n]{\bar{H_1}\bar{H_2}}}},$$

where $\bar{H_k} = \frac{\sum_{i=1}^{n} H_k(i))}{n}$.

4 Experiments Results and Performance Analysis

4.1 Performance Analysis

The experiment text corpus includes 200 pair entries collected from the Internet. 200 entries are collect from Baidu [1] and the same entries come from Baike [2]. The experiment tools include MATLAB 7.0 and C#.

In this research, we use precision P, recall R and $F1$-Measure for analyzing the results of experiment. This three indexes are most commonly used in the field of information retrieval and natural language processing. Precision is the fraction of retrieved instances that are relevant, while recall is the fraction of relevant instances that are retrieved. $F1$ is a synthesis evaluation parameter of precision and recall. The specific calculation formulas are as follows:

$$\begin{aligned} P &= \frac{truepositives}{truepositives + falsepositives} \\ R &= \frac{truepositives}{truepositives + falsenegatives} \\ F1 &= \frac{2 * P * R}{P + R}. \end{aligned}$$

4.2 Experiment Results And Analysis

Firstly, we use the four distance formulas to calculate the 200 pair texts entries and select ten entries of them shown in Table 1. As shown above, the distance value is smaller, the two texts are similarity. When the two texts are different from the content, the distance value will larger. So, we can see from the table, the distance value of text pair number 5 equal to 0, the content of the text are similarity. We analyzes the two texts by manual, and found the two texts are similarity. Another number 3, the distance value is larger to 0, so the two texts are different from each other in contend. This is fit to our manual analysis.

The threshold α is important criteria in our detection algorithm. The criteria will affect the parameters of our detection algorithm. So, it is important to select appropriate threshold α. Firstly, we select distance formula *Bhattacharyya* to show the threshold α effect the parameters of algorithm. The parameters of precision P, recall R and $F1$-Measure with different threshold α is shown in Figure 5. From this graphic, with the threshold α larger, the parameters P, R and $F1$ are close to 1. When the threshold α is 0.3, the three parameters equal to 1. But if the threshold α is becoming smaller, the parameters P, R and F1 reduce very fast. So, we choose the threshold $\alpha = 0.1$ to test the four distance formula. The experiment result is shown in Figure 6.

In order to get the best performance, we think about the threshold α and $F1$. The experiment result is list in Table 2. From this table, when the distance formula is *Bhattacharyya* and the threshold $\alpha = 0.1$, $F1 = 0.9$.

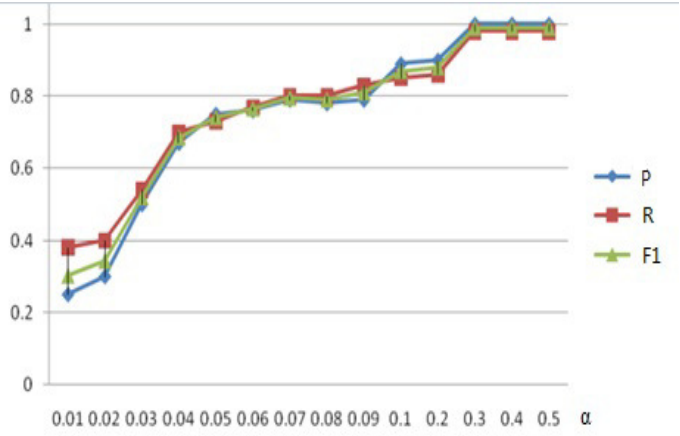


Figure 5: Threshold α

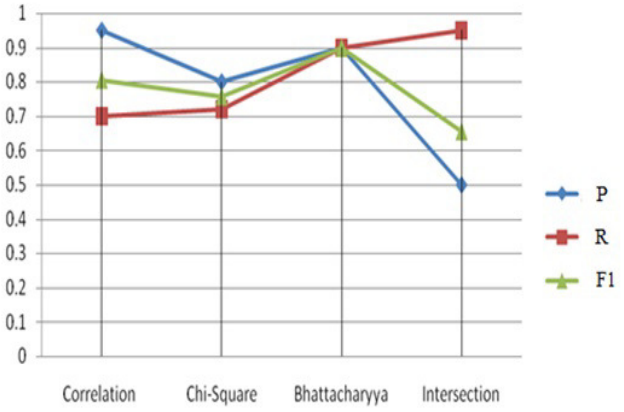


Figure 6: Distance formulas

Table 1: The distance of four formulas

Pairs of Numbers	Correlation	Chi-Square	Intersection	Bhattacharyya
1	0.045617	0.11053	0.16319	0.18374
2	0	0.00020945	0.006156	0.0072509
3	0.071436	0.14858	0.20078	0.22477
4	0.025078	0.076275	0.13112	0.15592
5	0	0	0	0
6	0.030073	0.096455	0.14462	0.17483
7	0.011809	0.03041	0.087298	0.092447
8	0.001564	0.0052661	0.03533	0.036437
9	0.014378	0.046964	0.0998042	0.11998
10	0.04337	0.15721	0.18893	0.22385

Table 2: Threshold α and F1

Distance formula	α	F1
Correlation	0.2	0.6
Chi-Square	0.15	50.8
Bhattacharyya	0.08	0.8
Intersection	0.1	0.9

After the parameter of our detection algorithm is found. We select *Cosine* algorithm and *Jaccard* method proposed in literature to prove ours are better than the two methods in many areas.The parameters P, R, $F1$ is shown in Figure 7. From this graphic, ours method is better than Cosine and *Jaccard*.

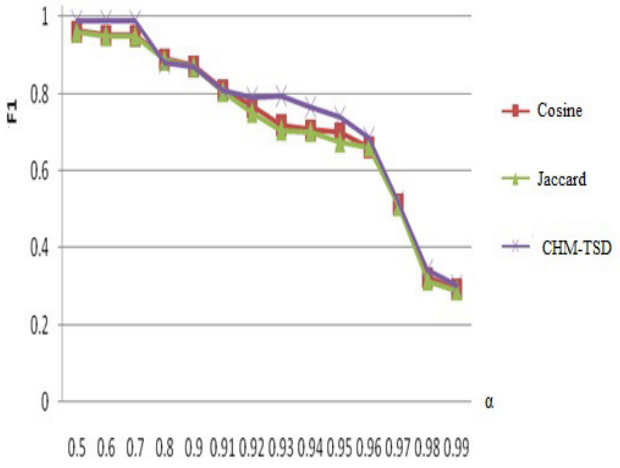


Figure 7: F1 to different methods

5 Conclusion

Text duplicate detection is mainly used for copy detection and webpage de-duplicate. It is also an effective ways for maintaining information quality. This paper put forward a new algorithm of text duplicate detection after the analysis and research on the structure of Chinese characters. CHM-TSD starts a new view of Chinese text similarity detection research. Chinese characters in text are split into components to build CHM. The texts similarity is obtained by computing the distance of all text CRM and duplicated text CHM. The experimental results show that CHM-TSD performs better than cosine theorem and *Jaccard* coefficient.

This paper provides a new idea of the natural language processing. The method can be used for text information processing and duplicated webpages deletion. In our future work, we will improve the efficiency of component decomposing and enhance the precision of the algorithm on the detection of the two texts that have a large variation on the number of words.

Acknowledgment

This study is supported by National Natural Science Foundation of China (No. 61304208, 61202496), Hunan Province Natural Science Foundation of China (No. 13JJ2031), and Youth Scientific Research Foundation of Central South University of Forestry & Technology (No. QJ2012009A).

References

[1] "Baidu," `http://www.baidu.com`, 2014. [Online; accessed 3-MARCH-2014].

[2] "Baike," `http://www.baike.com`, 2014. [Online; accessed 3-MARCH-2014].

[3] J. P. Bao, J. Y. Shen, and X. D. Liuand, Q. B. Song, "A survey on natural language text copy detection," *Journal of Software*, vol. 14, no. 10, pp. 1753–1760, 2003.

[4] C. L. Chen, F. S. C. Tseng, and T. Liang, "An integration of fuzzy association rules and wordnet for document clustering," *Journal of Knowledge and Information Systems*, vol. 28, no. 4, pp. 687–708, 2011.
[5] C. H. Huang, J. Yin, and F. Hou, "A text similarity measurement combining word semantic information with TF-IDF method," *Chinese Journal of Computers*, vol. 34, no. 5, pp. 856–864, 2011.
[6] X. G. Peng, S. M. Liu, and Y. Song, "A copy detection tool for chinese documents," in *Proceedings of the 2nd International Conference on Education Technology and Computer*, pp. 125–129, 2010.
[7] J. Shi, Y. F. Wu, L. K. Qiu, and X. Q. Niu, "Chinese lexical semantic similarity computing based on large-scale corpus," *Journal of Chinese Information Processing*, vol. 27, no. 1, pp. 1–6, 2013.
[8] C. N. Sun, C. Zhang, and Q. S. Xia, "Chinese text similarity computing based on lda," *Journal of Computer Technology and Development*, vol. 23, no. 1, pp. 217–220, 2013.
[9] X. M. Sun and J. P. Yin, "On mathematical expression of a Chinese character," *Journal of Computer Research and Development*, vol. 39, no. 5, pp. 707–711, 2004.
[10] M. V. Thada and M. S. Joshi, "A genetic algorithm approach for improving the average relevancy of retrieved documents using jaccard similarity coefficient," *International Journal of Research in IT Management*, vol. 11, no. 3, pp. 50–55, 2011.
[11] L. X. Wang, H. T. Gong, K. Sun, and X. Zhang, "Auto-detection technology of text divulgence based on natural language processing," *Computer Engineering and Design*, vol. 32, no. 8, pp. 2600–2603, 2011.
[12] Z. G. Wang and M. Wu, "Similarity checking algorithm in item bank based on vector space model," *Computer System Application*, vol. 19, no. 3, pp. 213–216, 2011.
[13] P. Y. Zhang, "A hownet-based semantic relatedness kernel for text classification," *TELKOMNIKA Indonesian Journal of Electrical Engineering*, vol. 11, no. 4, pp. 1909–1915, 2013.
[14] X. C. Zhang, W. Xu, and L. Gao, "Combining content and link analysis for local web community extraction," *Journal of Computer Research and Development*, vol. 49, no. 11, pp. 2352–2358, 2012.
[15] D. P. Zhao, L. J. Cai, and P. Li, "A similar text detection algorithm based on newshingling," *Journal of Shenyang Jianzhu University (Natural Science)*, vol. 27, no. 4, pp. 771–775, 2011.

Huajun Huang is currently a faculty member in the college of Computer and Information Engineering at Central South University of Forestry & Technology.His overall research area include of Webpage information hiding and hidden information detection, XML Watermarking, Anti-phishing, Mobile Device Forensics. Dr. Huang received his Ph.D. from Hunan University in 2007, M.S. degrees from Hunan University in Software Engineering (2004), and a B.A. in Applied Physics from Yunnan University (2001).

Shuang Pang is currently a postgraduate student in the college of Computer and Information Engeering at Central South University of Foresty &Technology.Ms Pang received her B.A.in Software Engeineer from Central South University of Foresty & Technology in 2014.

Qiong Deng is currently a postgraduate student in the college of Computer and Information Engeering at Central South University of Foresty &Technology.Ms Deng received her B.A.in Software Engeineer from Central South University of Foresty & Technology in 2014.

Jiaohua Qin received her BS in mathematics from Hunan University of Science and Technology, China, in 1996, MS in computer science and technology from National University of Defense Technology, China, in 2001, and PhD in computing science from Hunan University, China, in 2009. She a professor in College of Computer Science and Information Technology, Central South University of Forestry and Technology, China. Her research interests include network and information security, image processing and pattern recognition.

A Meaningful Scheme for Sharing Secret Images Using Mosaic Images

Shengyun Zhai[1], Fan Li[1], Chin-Chen Chang[2,3] and Qian Mao[1]
(Corresponding author: Chin-Chen Chang)
School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology[1]
516, Jungong Road, Yangpu, Shanghai 200093, P. R. China
Department of Computer Science and Information Engineering, Asia University[2]
500, Lioufeng Road, Wufeng, Taichung 41354, Taiwan
Department of Information Engineering and Computer Science, Feng Chia University[3]
100, Wenhwa Road, Seatwen, Taichung, 40724, Taiwan
(Email: alan3c@gmail.com)
(Received Dec. 2, 2014; revised and accepted Feb. 8 & Mar. 7, 2015)

Abstract

Secret image sharing (SIS) technique protects an image by sharing it among many users. Most existing SIS schemes produce meaningless shadow images, which tends to cause attackers' suspicion. In this paper, a meaningful secret image sharing scheme is proposed that, for the first time, uses mosaic images as the shadow images. The secret image is divided into several parts, and each part is transformed into a mosaic image according to a host image. The indices of the secret tiles of the mosaic image are permuted according to the secret key, and they are hidden in the mosaic images to provide security for the SIS scheme. The experimental results showed that the proposed SIS scheme can reconstruct the original secret image and provide high security.

Keywords: Indices hiding, mosaic image, random numbers, secret image sharing

1 Introduction

Secret image sharing (SIS) is a technique that ensures the security of a secret image by sharing it with several users. The secret image can be reconstructed only when a certain number of users cooperate together. The principle of secret sharing, which is based on the (r, n) threshold, was proposed by Shamir [13] and Blakley [1] independently in 1979. The secret is divided into n shadow images $W_i(i = 1, 2, ..., n)$, and any r or more than r shadow images can reconstruct the secret image cooperatively; none of the secret information can be obtained unless at least r shadow images cooperate together. Many secret image sharing (SIS) methods have been proposed based on this principle. Lee and Chiu [5] obtained $n - 1$ meaningful natural shadow images and one noise-like shadow image after sharing the secret image into $n - 1$ natural images. Then, the noise-like shadow image was hidden through steganography and quick-response code (QR code) techniques, which can reduce the risk of transmission. Using steganographic method, Lin and Chan [6] obtained the lossless secret image and the original host image from the shadow images. A $(2, n)$ matrix-based SIS scheme was proposed by Rey [12]. However, Elsheh *et al.* [4] pointed out that the threshold property of Rey's scheme was compromised, which means that the secret information could be reconstructed from only one shadow. To solve this problem, Yang *et al.* [20] presented a new method using random binary matrices. On the basis of pixel division and XOR operation, Bhattacharjee *et al.* [2] proposed a $(2, n)$ secret image sharing scheme to reconstruct the secret image correctly. In order to improve the security of the secret image and the quality of the reconstructed secret image, Chen [3] proposed a new sharing scheme using linear equations of Hill cipher and random grid. Wang *et al.* [17] proposed a creative scheme, which can share more than one secret images, using matrix transformation. Latif *et al.* [7] utilized random grids, error diffusion, and chaotic encryption to propose a new sharing scheme that can generate meaningful shadow images. Lin and Wang [8] presented a (t, n) scalable secret image sharing method, in which the size of the shadow image is $(2n - t)/n^2$ times that of the original secret image. Based on block truncation coding (BTC), discrete wavelet transform (DWT) and vector quantization (VQ), Le *et al.* [9] proposed an SIS scheme that also can generate small shadow images. However, the shadow images of [8] and [9] are noise-like images, which can increase the risk associated with the transmission of the shadow images.

Recently, many algorithms associated with hiding images have been proposed. According to vector quanti-

zation, Shie *et al*. [14] proposed a visually-imperceptible image hiding scheme. Through this method, multiple secret images can be hidden into the cover image. Based on phase-truncation and phase retrieval in the fractional Fourier domain, a method of hiding the color image into the host image was proposed by Wang *et al*. [18]. A data hiding algorithm was proposed in [19] that uses block patterns to hide a large amount of data into a binary image without attracting the attention of a hacker. Based on LSB substitution and pixel difference, Tsai *et al*. [15] proposed a new scheme to realize information hiding.

In [10], a scheme for hiding secret images was proposed, in which the meaningful cover image can be obtained by assembling the tiny fragments of the secret image. Lee and Tsai [11] used pixel color transformation to improve the quality of the mosaic image provided by the scheme proposed in [10]. Based on color, a scheme of reassembling the fragments of the image rapidly and efficiently was proposed by Tsamoura [16]. In [21], it was shown that potential matching can be generated according to the geometry and color of the fragments. However, if the two aspects of the fragments are similar, this scheme cannot achieve an accurate match.

In the previous research, the generated shadow images are not ideal for most of them are noise-like images, which may arouse the attention of hackers. In this paper, a new SIS scheme is proposed that improves the security and visualization of the shadow images by combining the benefits of traditional secret image sharing and the image mosaicing technique. The contributions of the proposed scheme are that:

1) For the first time, it uses the image mosaicing technique to share the secret image, achieving high-quality shadow images.

2) The sequence of the indices is encrypted by seed key K, which improves the security of the scheme.

3) The shadow images are meaningful, so they do not attract the hacker's attention.

The rest of this paper is organized as follows. The proposed scheme is introduced in Section 2. The experimental results and analyzes are presented in Section 3. The conclusions are discussed in Section 4.

2 Proposed SIS Scheme Based on Image Mosaicing

The proposed SIS scheme is comprised of three phases, i.e., secret image sharing, tile-indexes sharing, and secret image restoration.

2.1 Secret Image Sharing

The size of the secret image is $m_S \times n_S$. First, the secret image is divided into n blocks, each of which has the size of $\frac{m_S}{n} \times n_S$. Here, the parameter n is chosen so that m_S is divisible by it. Then, each block is divided into many adjacent tiles. All of the tiles have the same shape and the same size, i.e., $a \times b$. a, b and n are fixed, which are known to both sender and receiver. Figure 1 shows the process for generating the tiles.

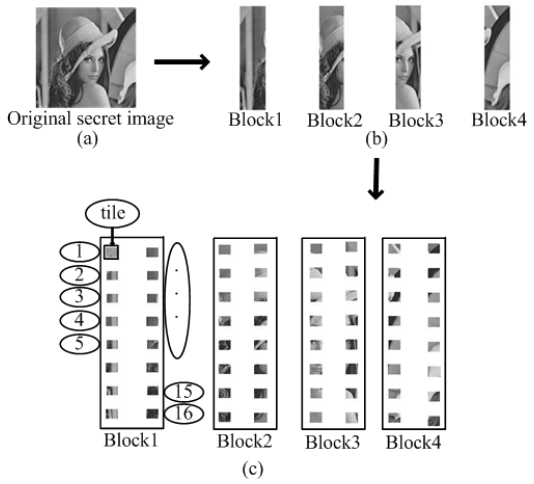


Figure 1: Tiles generation

Following that, the n host images with sizes of $m_H \times n_H$ are chosen, in which $m_H = k_1 m_S$ and $n_H = k_2 n_S (k_1, k_2 = 1, 2, ..., l)$($l$ is an integer). In the phase in which the shadow images are generated, the ith secret block will be hidden in the ith host image, where $i = 1, 2, ..., n$. Each host image is divided into adjacent tiles with the same shape and size, i.e., $a \times b$. The number of tiles, L_H, in each host image is:

$$L_H = \frac{m_H \times n_H}{a \times b}. \tag{1}$$

Similarly, the number of tiles in the secret image block is $L_S = \frac{m_S \times n_S}{n \times a \times b}$. For the ith tile ($i = 1, 2, ..., L_S or L_H$)($L_H \geqslant L_S$) of either the secret image block or the host image, the feature value is computed by the following function:

$$f_i = p \times m_i + q \times d_i, \tag{2}$$

$$s_{ij} = |f_i^S - f_j^H|, \tag{3}$$

where m_i is the average of the pixels' gray values in the ith tile, d_i is the standard deviation, $i \in \{1, 2, 3, ..., L_S\}$, $j \in \{1, 2, 3, ..., L_H\}$. The symbols p and q are the weighting factor of m_i and d_i, respectively. The mosaic image changes as the weighting factor changes. In the experiment, p equals to 0.99 and q equals to 0.01, which are the empirical values. In this case, the quality of the mosaic image was acceptable. For the ith tile in a secret image block and in a host image, the feature value f_i can be obtained by Equation (2). The secret tile's feature value f_i was called f_i^S and the host tile's feature value f_i was called f_i^H.

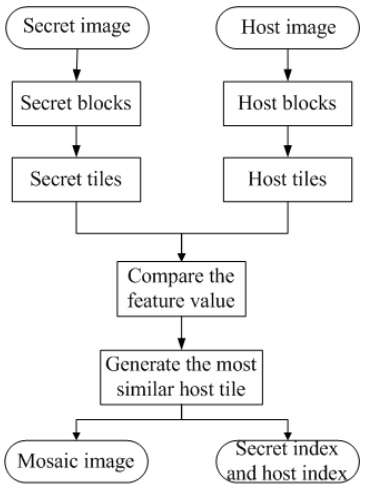


Figure 2: Process of the secret image sharing

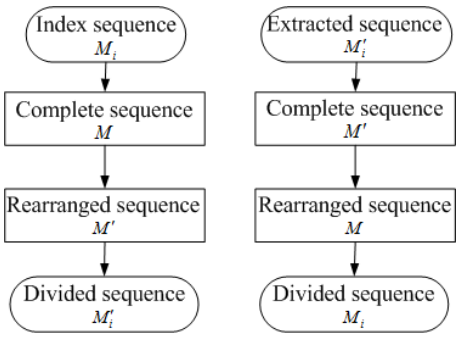


Figure 3: Process of indices sequence

Conducting the operation as shown in Equation (2) for the entire secret image and the entire host image, the feature sequences, $F^S = f_1^S, f_2^S, ..., f_{L_S}^S$ and $F^H = f_1^H, f_2^H, ..., f_{L_H}^H$, can be obtained. The order of scanning the secret tile increases monotonically. Then, according to the Equation (3), the similarity value between the secret tile and the host tile can be obtained. The first secret block and the first host image are considered for the image mosaicing. The first feature value of secret tile f_1^S and all of the whole feature values of the host tiles are scanned. And the similarity value, s_{1j}, between the feature of the first secret tile and the feature of the jth host tile, is obtained, where $j = 1, 2, ..., L_H$. The smaller s_{ij} is, the more similar between the ith secret tile and the jth host tile will be. So, the smallest similarity value, s_{1,j_1}, is chosen among the L_H similarity values. That is to say, the host tile H_{j_1} is the tile that is the most similar to the first secret tile. After that, for the second feature value of the secret tile f_2^S, all of the host feature values, except for $f_{j_1}^H$, are scanned. The similarity values between f_2^S and each of the rest of the feature values of the host tile are created and referred to as called s_{2j}. The smallest value s_{2,j_2}, among the $L_H - 1$ similarity values indicates that the host tile, H_{j_2}, is the most similar to the second secret tile. Similarly, for the ith secret feature value and the remaining $L_H - i + 1$ host tiles, the $L_H - i + 1$ similarity values are created and the smallest one is obtained. Thus, the host tile, H_{j_i}, which is the most similar to the ith secret tile is identified.

After that, for all of the tiles $S_1, S_2, ..., S_i, ...$, in the first secret block, the most similar tiles $H_{j_1}, H_{j_2}, ..., H_{j_i}, ...$, in the first host image are chosen. Thus, the indices sequence M, including $M_S = 1, 2, ...$(the order of secret tiles in one block) and $M_H = j_1, j_2, ...$(the order of the chosen host tiles), is constructed. Moving the secret tiles $S_1, S_2, ...$to the corresponding position of the host tiles $H_{j_1}, H_{j_2}, ...$, respectively, the preliminary shadow image is obtained, i.e., the mosaic image. The vector M_H is restored as the indices to be used in reconstructing the secret. The same operation is conducted for all of the secret blocks and host images. Figure 2 shows the flowchart that is used to generate shadow images.

2.2 Index Sharing

After generating the shadow images, n indices sequences $M_i (i = 1, 2, ..., n)$ are obtained. Based on the order of the secret blocks, the indices sequences M_i are connected to form a complete indices map, called M, $M = M_1||M_2||...||M_n$. Now, indices sequence M is encrypted by seed key K and a permutated sequence, M_r, is obtained. The key K is fixed, which is known to both sender and receiver. Divide M_r into n segments and transform them into a binary string. Using the least significant bit (LSB) method, the binary string of the ith segment is hidden into the ith mosaic image. Therefore, the shadow images are obtained. Figure 3 shows the flowchart that is used to generate indices sequence.

2.3 Secret Image Restoration

When reconstructing the secret image, first, the binary indices sequence is extracted from the shadow images and transformed into decimal numbers, $M_i^{'}$. In the order of the host images, the components of the sequence $M_i^{'}$ are connected with each other to construct the complete indices sequence $M^{'} = M_1^{'}||M_2^{'}||...||M_n^{'}$. The original order of the indices sequence $M^{'}$ has been rearranged by the seed key K, then the sequence M is divided into n segments, called $M_i (i = 1, 2, ..., n)$. An empty image is defined, which size is identical to the size of the original secret image. The empty image and the shadow images are divided, respectively, as the division of original secret image and host image. The tiles of the shadow images are moved to the corresponding positions of tiles of the empty image according to indices sequence M_i. Thus, the original secret image is reconstructed.

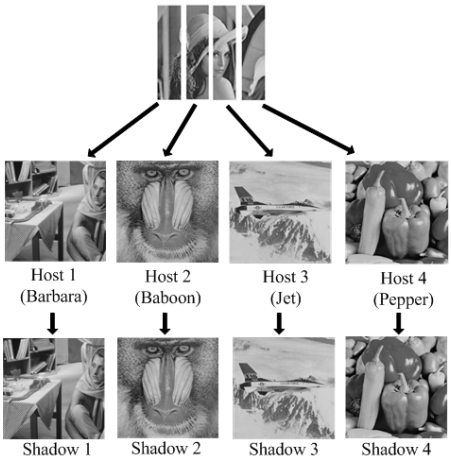


Figure 4: Secret image sharing

Figure 5: Secret images

Table 1: PSNR between host images and shadow images

Images (384×384)	PSNR of shadow images (dB)
Barbara	39.52
Baboon	37.26
Jet	37.41
Pepper	39.64

3 Experimental Results and Analyzes

3.1 Experimental Results

An example of sharing a secret image by this scheme is shown in Figure 1 and Figure 4. In Figure 1, the original secret image is divided into four blocks, and each block is divided into many tiles. The *ith* block is hidden into the *ith* host image as shown in Figure 4, where $i = 1, 2, 3, 4$. In the following, the secret image, Lena, as shown in Figure 1, is hidden in four host images, as shown in Figure 4. For the sake of convenience, the size of the secret image is 128×128. The secret image is divided into four blocks. The size of each host image is 384×384. The secret blocks and host images are divided into 2×2 tiles. The experimental results are shown in Figure 4 and Table 1. The reconstructed secret image is shown in Figure 5. The experimental results indicated that the shadow images were meaningful and that their quality was acceptable, which proved the usefulness of the proposed visualization scheme in the field of secret image sharing.

3.2 Performance Analyzes

The relationship between the quality of the shadow image and the size of the host image was analyzed first. The sizes of the host images were varied from 128×128 to 256×256 and to 384×384. The size of the secret image was 128×128, and the size of tile was 2×2. It is obvious that the quality of the shadow image improved as the size of the host image increased, as shown in Figures 6 and 7.

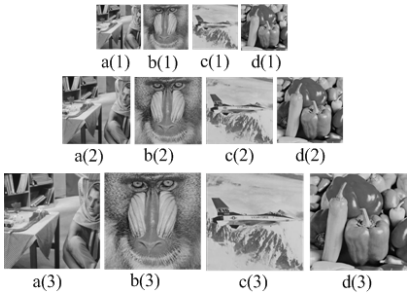


Figure 6: Different sizes of shadow images: a(1)∼d(1): 128×128, a(2)∼d(2): 256×256, a(3)∼ d(3): 384×384.

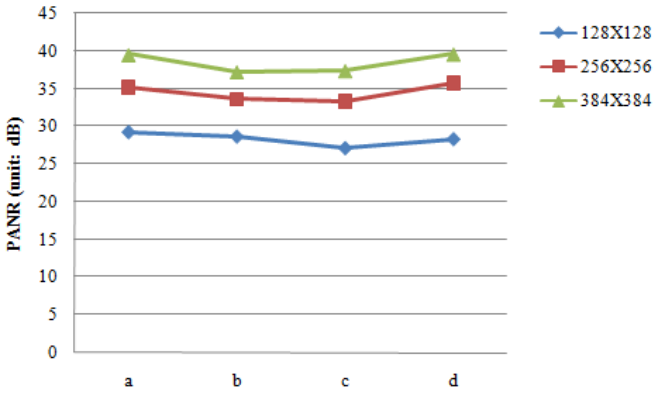


Figure 7: PSNR between host images and shadow images with different sizes

The general images are chose as the secret images and host images, as shown in Figure 8. In the experiments, PSNR between general host images and shadow images with different size are shown in Figure 9. For the same size secret image, the larger the size of the host image is, the better the quality of the shadow image will be. This is because a larger host image provides more choices for identifying the similarity value for each tile of the secret image. PSNR between mosaic images and host images with different sizes of tiles are shown in Table 2, while PSNR between shadow images and host images with different sizes of tiles are shown in Table 3. It can be seen

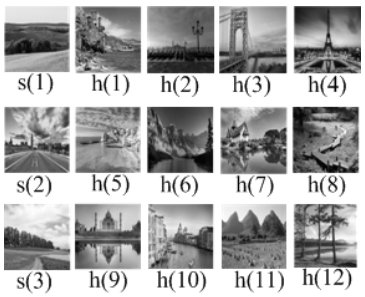


Figure 8: s(3)∼s(3) the three secret images, h(1)∼h(4) the host images to s(1), h(5)∼h(8) the host images to s(2), h(9)∼h(12) the host images to s(3)

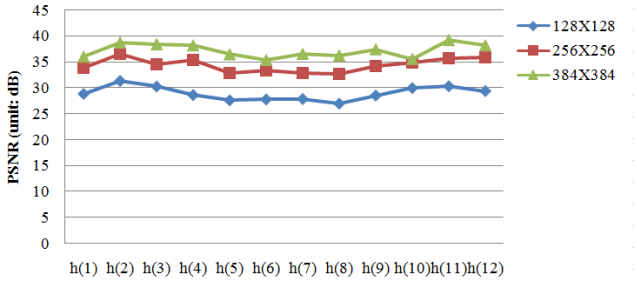


Figure 9: PSNR between general host images and shadow images with different sizes

from Table 2, the PSNR values decrease as the sizes of tiles increase, that is to say, the size of tile have an important impact on the image quality. By the comparison between Table 2 and Table 3, for the same size of tiles, the PSNR values decrease slightly, i.e., the impact of the data hiding to the image quality is tiny. The PSNR values between secret images and restored secret images are shown in Figure 10, which illustrates that the PSNR values increase as the sizes of tiles increase. Meanwhile, the quality of restored secret image is also acceptable when the size of tile is the smallest, i.e., 2×2. In practice applications, users can adapt the sizes of tiles to obtain suitable shadows according to the requirement. It can be seen from the Table 4 that the computational time is increased when the size of tile is decreased. Especially when the size of tile is 2×2, the computational time is increased sharply.

The experimental results of this method and other three methods are shown in Table 5. The shadow images created by Lin and Wang's scheme [8] and Bhattacharjee *et al.*'s method [2] are meaningless, so they likely to attract the attention of hackers. The algorithm proposed by Latif *et al.* [7] can produce meaningful shadow images by encoding the secret image into natural host images. However, this encryption technology changes the pixels of original image, so the shadow images can be identified during transmission. In this study, the shadow images are meaningful and the pixels of the original image are

Table 2: PSNR between mosaic images and host images with different sizes of tiles (unit: dB)

Mosaic images	Size			
	2×2	4×4	8×8	16×16
h(1)	36.04	30.87	27.79	25.90
h(2)	38.77	36.14	31.88	28.82
h(3)	38.46	33.14	30.90	28.75
h(4)	38.24	33.74	29.88	30.00
h(5)	36.52	32.10	29.73	27.73
h(6)	35.40	30.60	28.89	27.66
h(7)	36.58	32.39	29.60	28.24
h(8)	36.28	31.63	28.87	25.89
h(9)	37.48	32.70	30.22	29.71
h(10)	35.61	29.78	28.23	28.67
h(11)	39.27	34.43	31.68	29.54
h(12)	38.23	34.95	31.84	31.23

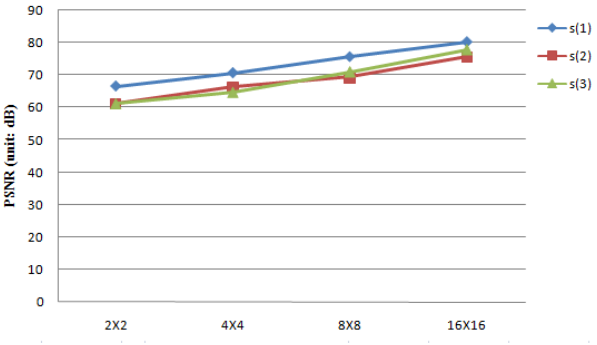


Figure 10: PSNR between general secret images and restored secret images with different sizes of tiles

Table 3: PSNR between shadow images and host images with different sizes of tiles (unit: dB)

Shadow images	Size			
	2×2	4×4	8×8	16×16
h(1)	36.03	30.87	27.80	25.91
h(2)	38.74	36.13	31.88	28.82
h(3)	38.43	33.14	30.90	28.75
h(4)	38.21	33.74	29.88	30.00
h(5)	36.50	32.10	29.73	27.73
h(6)	35.39	30.60	28.89	27.66
h(7)	36.56	32.38	29.60	28.24
h(8)	36.27	31.63	28.87	25.89
h(9)	37.46	32.70	30.22	29.71
h(10)	35.60	29.78	28.22	28.67
h(11)	39.24	34.43	31.68	29.54
h(12)	38.20	34.95	31.83	31.23

Table 4: Computational time of the general secret image sharing (unit:second)

Secret images	Size			
	2×2	4×4	8×8	16×16
s(1)	66.6	8.4	4.0	3.3
s(2)	63.0	8.4	3.9	3.4
s(3)	55.3	8.2	4.0	3.4

Table 5: The thresholds and shadow images descriptions

Schemes	(t,n)	Description
Bhattacharjee *et al.*'s scheme [2]	$t = 2$	Meaningless shadow images
Latif *et al.*'s scheme [7]	$t < n$	Meaningful shadow images
Lin and Wang's scheme [8]	$t < n$	Meaningless shadow images
Proposed scheme	$t = n$	Meaningful shadow images

unchanged. In addition, this scheme can be used to share other forms of secret images, such as color images, texture images.

4 Future Work

In this paper, the reconstructed secret image is lossy because the irreversible hiding method was used in the embedding of the indices sequence. In the future, the research work is directed to developing a reversible hiding method to hide the indices sequence and reconstruct the lossless secret image. In addition, a more accurate method of calculating the similarity of the tiles is necessary, which can improve the quality and security of the shadow images.

5 Conclusions

A new kind of sharing scheme, called meaningful secret image sharing with a mosaic image, was proposed in this work. In the scheme, it is the first time that the mosaic technique has been used in sharing secret images, and a good quality of the shadow images is achieved. And also, the generated shadow images are meaningful. Furthermore, a seed key K is used to generate the rearranged order, which is the order of the indices, to ensure the security of the algorithm. The comparisons between the proposed scheme and other existing secret image sharing techniques demonstrate a good performance of the proposed algorithm.

References

[1] G. R. Blakley, "Safeguarding cryptographic keys," *AFIPS Conference Proceedings*, vol. 48, pp. 313–317, 1979.

[2] T. Bhattacharjee, J. P. Singh and A. Nag, "A novel $(2, n)$ secret image sharing scheme," *Procedia Technology*, vol. 4, pp. 619–623, 2012.

[3] W. K. Chen, "Image sharing method for gray-level images," *Journal of Systems and Software*, vol. 86, no. 2, pp. 581–585, 2013.

[4] E. Elsheh and A. B. Hamza, "Comments on matrix-based secret sharing scheme for images," in *Progress in Pattern Recognition, Image Analysis and Applications*, LNCS 6419, pp. 169–175, 2010.

[5] K. H. Lee and P. L. Chiu, "Digital image sharing by diverse image media," *IEEE Transactions on Information Forensics and Security*, vol. 9, no, 1, pp. 88–98, 2014.

[6] P. Y. Lin and C. S. Chan, "Invertible secret image sharing with steganography," *Pattern Recognition Letters*, vol. 31, no. 13, pp. 1887–1893, 2010.

[7] A. A. E. Latif, X. H. Yan, L. Li, N. Wang, J. L. Peng and X. M. Niu, "A new meaningful secret sharing scheme based on random grids, error diffusion and chaotic encryption," *Optics and Laser Technology*, vol. 54, pp. 389–400, 2013.

[8] Y. Y. Lin and R. Z. Wang, "Scalable secret image sharing with smaller shadow images," *IEEE Signal Processing Letters*, vol. 17, no. 3, pp. 316–319, 2010.

[9] T. H. N. Le, C. C. Lin, C. C. Chang and H.B. Le, "A high quality and small shadow size visual secret sharing scheme based on hybrid strategy for grayscale images," *Digital Signal Processing*, vol. 21, no. 6, pp. 734–745, 2011.

[10] I. J. Lai and W. H. Tsai, "Secret-fragment-visible mosaic image-a new computer art and its application to information hiding," *IEEE Transactions on Information Forensics and Security*, vol. 6, no. 3, pp. 936–945, 2011.

[11] Y. L. Lee and W. H. Tsai, "A new secret image transmission technique via secret-fragment-visible mosaic images by nearly reversible color transformations," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 4, pp. 695–703, 2014.

[12] A. M. Rey, "A matrix-based secret sharing scheme for images," in *Progress in Pattern Recognition, Image Analysis and Applications*, LNCS 5197, pp. 635–642, 2008.

[13] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, no. 11, pp. 612–613, 1979.

[14] S. C. Shie, S. D. Lin and J. H. Jiang, "Visually imperceptible image hiding scheme based on vector quantization," *Information Processing and Management*, vol. 46, no. 5, pp. 495–501, 2010.

[15] Y. Y. Tsai, J. T. Chen and C. S. Chan, "Exploring LSB substitution and pixel-value differencing for block-based adaptive data hiding," *International*

Journal of Network Security, vol. 16, no. 5, pp. 363–368, 2014.

[16] E. Tsamoura and I. Pitas, "Automatic color based reassembly of fragmented images and paintings," *IEEE Transactions on Image Processing*, vol. 19, no. 3, pp. 680–690, 2010.

[17] Z. H. Wang, H. B. Jin, X. B. Wang and C. C. Chang, "An adaptable (n, n) secret image sharing mechanism based on boonlean operation," *International Journal of Network Security*, vol. 16, no. 6, pp. 487–493, 2014.

[18] Q. Wang, Q. Guo and J.Y. Zhou, "Color image hiding based on phase-truncation and phase retrieval technique in the fractional Fourier domain," *Optik-International Journal for Light and Electrics Optics*, vol. 124, no. 12, pp. 1224–1229, 2013.

[19] C. C. Wang, Y. F. Chang, C. C. Chang, J. K. Jan and C. C. Lin, "A high capacity data hiding scheme for binary images based on block patterns," *Journal of Systems and Software*, vol. 93, pp. 152–162, 2014.

[20] C. N. Yang, C. C. Wu, Y. C. Lin and C. Kim, "Enhanced matrix-based secret image sharing scheme," *IEEE Signal Processing Letters*, vol. 19, no. 12, pp. 789–792, 2012.

[21] K. Zhang and X. Li, "A graph-based optimization algorithm for fragmented image reassembly," *Graphical Models*, vol. 76, no. 5, pp. 484–495, 2014.

Shengyun Zhai was born in Henan Province, China, in 1990. She received the B.S. degree in Mechanical and Automotive Engineering from Nanyang Institute of Technology, Henan, China, in 2013. She is currently working toward the M.S. degree in Instrumentation Engineering from University of Shanghai for Science and Technology, Shanghai, China. Her research interests include information sharing and image processing.

Fan Li was born in Shandong Province, China, in 1990. She received the B.S. degree in Electronic and Information Engineering from Shandong Institute of Business and Technology, Shandong, China, in 2012. She is currently working toward the M.S. degree in Signal and Information Processing from University of Shanghai for Science and Technology, Shanghai, China. Her research interests include information hiding and image processing.

Chin-Chen Chang received the B.S. degrees in Science in Applied Mathematics and M.S. degree in Science in computer and decision sciences. Both were awarded in National Tsing Hua University, Taiwan. He received his Ph.D. degree in computer engineering from National Chiao Tung University, Taiwan.

His current title is Chair Professor in Department of Information Engineering and Computer Science, Feng Chia University, from Feb. 2005. He is currently a Fellow of IEEE and a Fellow of IEE, UK. His current research interests include database design, computer cryptography, image compression and data structures.

Since his early years of career development, he consecutively won Outstanding Talent in Information Sciences of the R. O. C., AceR Dragon Award of the Ten Most Outstanding Talents, Outstanding Scholar Award of the R. O. C., Outstanding Engineering Professor Award of the R. O. C., Distinguished Research Awards of National Science Council of the R. O. C., Top Fifteen Scholars in Systems and Software Engineering of the Journal of Systems and Software, and so on. On numerous occasions, he was invited to serve as Visiting Professor, Chair Professor, Honorary Professor, Honorary Director, Honorary Chairman, Distinguished Alumnus, Distinguished Researcher, Research Fellow by universities and research institutes.

Qian Mao was born in Shanxi Province, China, in 1978. She received the B.S. degree in Mechanical Engineering and Automation Science from Nanjing University of Aeronautics and Astronautics, Jiangsu, China, in 2000, the M.S. degrees in Traffic Information Engineering and Control from Shanghai Ship and Shipping Research Institute, Shanghai, China, in 2003, and the Ph.D. degree in Traffic Information Engineering and Control from Tongji University, Shanghai, China, in 2006.

Since 2006, she has been with the faculty of the School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai, China, where she is currently a lecturer. She is also a post-doctoral researcher of Asia University, Taiwan. Her research interests include information security, image processing, and information theory and coding.

Guide for Authors
International Journal of Network Security

IJNS will be committed to the timely publication of very high-quality, peer-reviewed, original papers that advance the state-of-the art and applications of network security. Topics will include, but not be limited to, the following: Biometric Security, Communications and Networks Security, Cryptography, Database Security, Electronic Commerce Security, Multimedia Security, System Security, etc.

1. Submission Procedure

Authors are strongly encouraged to submit their papers electronically by using online manuscript submission at http://ijns.jalaxy.com.tw/.

2. General

Articles must be written in good English. Submission of an article implies that the work described has not been published previously, that it is not under consideration for publication elsewhere. It will not be published elsewhere in the same form, in English or in any other language, without the written consent of the Publisher.

2.1 Length Limitation:

All papers should be concisely written and be no longer than 30 double-spaced pages (12-point font, approximately 26 lines/page) including figures.

2.2 Title page

The title page should contain the article title, author(s) names and affiliations, address, an abstract not exceeding 100 words, and a list of three to five keywords.

2.3 Corresponding author

Clearly indicate who is willing to handle correspondence at all stages of refereeing and publication. Ensure that telephone and fax numbers (with country and area code) are provided in addition to the e-mail address and the complete postal address.

2.4 References

References should be listed alphabetically, in the same way as follows:
For a paper in a journal: M. S. Hwang, C. C. Chang, and K. F. Hwang, ``An ElGamal-like cryptosystem for enciphering large messages," *IEEE Transactions on Knowledge and Data Engineering*, vol. 14, no. 2, pp. 445--446, 2002.

For a book: Dorothy E. R. Denning, *Cryptography and Data Security*. Massachusetts: Addison-Wesley, 1982.

For a paper in a proceeding: M. S. Hwang, C. C. Lee, and Y. L. Tang, ``Two simple batch verifying multiple digital signatures," in *The Third International Conference on Information and Communication Security (ICICS2001)*, pp. 13--16, Xian, China, 2001.

In text, references should be indicated by [number].

Subscription Information

Individual subscriptions to IJNS are available at the annual rate of US$ 200.00 or NT 7,000 (Taiwan). The rate is US$1000.00 or NT 30,000 (Taiwan) for institutional subscriptions. Price includes surface postage, packing and handling charges worldwide. Please make your payment payable to "Jalaxy Technique Co., LTD." For detailed information, please refer to http://ijns.jalaxy.com.tw or Email to ijns.publishing@gmail.com.