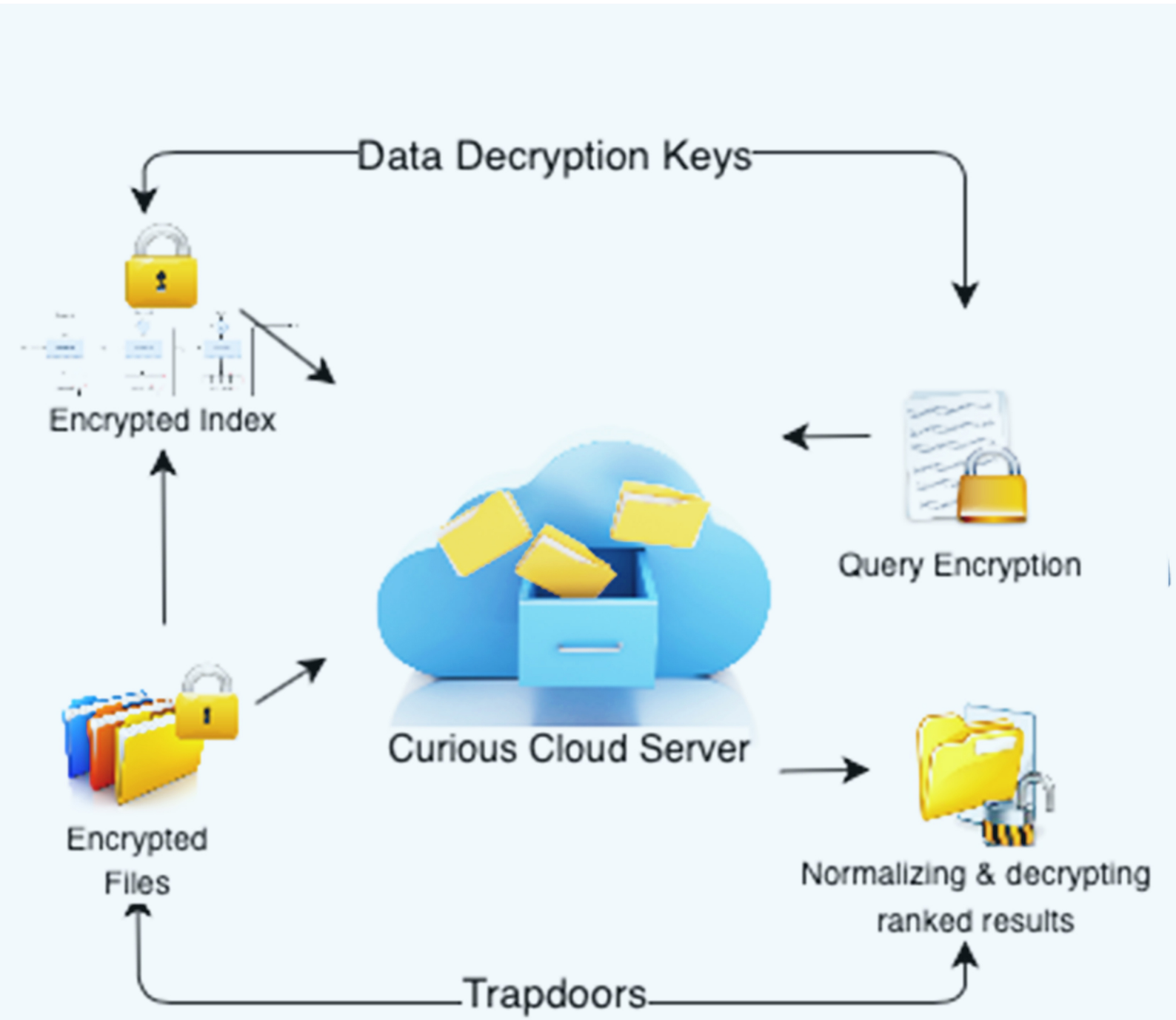
ISSN 2694-1015 (Online)
Volume 1 | Issue 1 | September 2020

Journal of Surveillance, Security and Safety

JSSS



www.jsssjournal.com

OAE Publishing Inc.

EDITORIAL BOARD

GENERAL INFORMATION

About the Journal

Journal of Surveillance, Security and Safety (JSSS), ISSN 2694-1015 (Online), is an open access, peer-reviewed, quarterly online journal which provides a forum for the publication of papers addressing the variety of theoretical, methodological, epistemological, empirical and practical issues concerns reflected in the field of information security, cyber security, machine learning, emerging technologies, and their applications. In particular, the journal encourages articles in the following areas:

AI-based surveillance and security
Privacy protection based on machine learning
Security of machine learning algorithms
Deep learning for attack and defense
Database security
Data-driven cybersecurity incident prediction
Big data security
Cloud/fog computing security
Outsourcing and crowdsourcing security
Security and privacy in pervasive/ubiquitous computing
Cyber-physical systems security
Security, privacy and resilience in critical infrastructures
Multimedia security
Wireless network security
Social networks and IoT security
Information hiding, forensics and security
Theory and applications of cryptography
Identity management, authentication and access control
Security policies, models and architectures
Electronic commerce security
Blockchain and finance security
Intrusion detection
Phishing and spam prevention
Biometrics
Regulation of the security industry
Risk analysis, security measures and management
Evaluations of security measures

Information for Authors

Manuscripts must be prepared in accordance with Instructions to Authors. Please check https://jsssjournal.com/pages/view/author_instructions for details. All manuscripts must be submitted online at www.jsssjournal.com/login.

Copyright

Permissions

Disclaimer

Published by
OAE Publishing Inc.
245 E Main Street ste115, Alhambra, CA 91801, USA
Website: www.oaepublish.com

Contacts
E-mail: editorial@jsssjournal.com
Website: www.jsssjournal.com

Journal of Surveillance, Security and Safety

CONTENTS

Schmidt *et al*. *J Surveill Secur Saf* 2020;1:1-15
DOI: 10.20517/jsss.2019.02

Journal of Surveillance, Security and Safety

Original Article

Open Access

Check for updates

Learning and unlearning from disasters: an analysis of the Virginia Tech, USA shooting and the Lion Air 610 Airline crash

Bianca Schmidt[1], Ashraf Labib[2], Sara Hadleigh-Dunn[3]

[1]Strategy, Enterprise and Innovation, Faculty of Business and Law, University of Portsmouth, Portsmouth PO1 3DE, United Kingdom.
[2]Operations and Systems Management, Faculty of Business and Law, University of Portsmouth, Portsmouth PO1 3DE, United Kingdom.
[3]Strategy, Enterprise and Innovation, Faculty of Business and Law, University of Portsmouth, Portsmouth PO1 3DE, United Kingdom.

Correspondence to: Prof. Ashraf Labib, For Operations and Systems Management, Faculty of Business and Law, University of Portsmouth, Portsmouth PO1 3DE, United Kingdom. E-mail: ashraf.labib@port.ac.uk

How to cite this article: Schmidt B, Labib A, Hadleigh-Dunn S. Learning and unlearning from disasters: an analysis of the Virginia Tech, USA shooting and the Lion Air 610 Airline crash. *J Surveill Secur Saf* 2020;1:1-15. http://dx.doi.org/10.20517/jsss.2019.02

Received: 24 Dec 2019 **First Decision**: 10 Mar 2020 **Revised**: 22 Mar 2020 **Accepted**: 7 Apr 2020 **Available online**: 10 Sep 2020

Academic Editor: Michael G. Pecht **Copy Editor**: Jing-Wen Zhang **Production Editor**: Jing Yu

Abstract

Aim: The aim of this paper is to explore whether and how far organisations learn from failures.

Methods: The paper reviews the current literature about organisational learning and theories of learning from failures, where learning here implies change of practice, and use of modelling techniques to inform recommendations to prevent repetition of similar incidents. Further, it analyses two case studies related to aspects of security and safety: the Virginia Tech Shooting in 2007 and the Lion Air 610 crash in 2018. Both case studies address the concept of learning from failures. In doing so, an analysis of the root causes and vulnerabilities through the methods of Fault Tree Analysis and Reliability Block Diagram is conducted to identify lessons to be learned.

Results: Findings are intended to stimulate organisational learning and improve organisational processes to mitigate disasters from happening again.

Conclusion: The value of this study is that aspects of learning and unlearning from failures have been identified for the cases used. Expectation for future studies is to extend the proposed methodology to other cases in the fields of surveillance, security and safety.

Keywords: Learning from failures, organisational learning, Virginia Tech Shooting, Lion Air 610 crash

1. INTRODUCTION

Failures are often associated with negative implications and illustrated to be something that should be avoided in organisations. However, it has been identified that failures can help improve organisational learning and strengthen an organisation's resilience[1-4]. Further, it is argued that organisations learn more from failures than successes[5]. However, learning from failures is not always achieved by organisations due to factors such as denial of failure, issues of ineffective communication and information sharing, status quo and lack of corporate responsibility[6]. When failures are not detected in time, they can cause a chain reaction resulting in a major failure or disaster. Within organisations, these major failures are often seen as Black Swans as their occurrence is low, but their impact is severe[7]. According to Fortune and Peters[8], failures that have such a destructive impact that they receive widespread media attention and investigation are considered disasters. These impacts are not limited to a number of fatalities and casualties but also related to the wider influence on for example economies, policies and communities. Knight and Pretty's[9] seminal work demonstrated a clear link between an organisation's positive handling of a failure and the potential increase in cumulative annual returns, suggesting a further financial imperative to encourage organisations to better engage with learning from failure. The aim of this paper is to explore whether and how far organisations learn from failures. The main contributions of this paper are two-fold: First, we demonstrate that a common modelling set of tools can be applied to completely different disasters chosen from the two domains of safety and security. Second, we present a taxonomy of failures classification and the role of mental modelling in learning from failures.

2. METHODS

Following the Introduction, the paper reviews the current literature on organisational learning, learning and unlearning from failures as well as explaining failure theories. The paper then analyses two case studies related to aspects of security and safety: the Virginia Tech, USA shooting in 2007 and the Lion Air 610 crash in 2018. Both cases have been chosen to compare efficient learning from failures versus inefficient learning from failures. Further, they have been selected to demonstrate the framework of learning from failures introduced by Labib and Read[1], which addresses three aspects: the aspect of feedback from users to design (i.e., change the status quo), the incorporation of failure analytical tools (new mental models) and the generic lessons learned (i.e., isomorphic learning), highlighting the wider applicability of the approach, independent of industry or event type. The paper concludes with a discussion and a summary.

The case studies are following the framework of Labib[10] by first introducing the case and sequences that resulted in the disaster. After the technical cause and logic of the failure is assessed and consequences addressed, methods such as the Fault Tree Analysis (FTA) and the Reliability Block Diagram (RBD) are used to identify the causes and the vulnerability factor. The case studies are concluded by recommendations and identification of generic lessons to support organisational learning from failure.

2.1 Organisational learning and unlearning

Learning is an important aspect of life as it is can stimulate change and improve actions through better understanding and knowledge[11]. This approach not only influences individuals but also organisations. Organisational learning implements the standard definition and applies it to a wider context. Since the 1980s, the idea of organisational learning has flourished, with numerous definitions and studies into the concept[12]. According to Toft and Reynolds[13], organisational learning is seen as a process in which individuals in an organisation continuously reflect upon and reinterpret their working environment and the experiences encountered in order to improve actions. This definition was supported by Madsen and

Desai[5] who defined organisational learning as "any modification of an organization's knowledge occurring as a result of its experience". To take this definition a step further, it is argued that organisational learning should not only focus on own experiences but also those of others. Learning from others' experiences is defined as vicarious learning[5] or isomorphic learning[13]. Both aspects of learning not only positively influence organisational processes but also enhance organisational resilience and can be an important catalyst in organisational change, as observed by Weinzimmer and Esken[14].

Whilst understanding of the concept developed, a desire to maximise learning for organisations began and, from this, barriers to learning and enablers to learning started to be explored. Debate regarding whether learning from failure or success offered greater benefit started to occupy some of the research within organisational learning and, from this, the identification of second-order effects and ultimately organisational unlearning evolved[15].

Definitions of organisational unlearning could be argued to revolve around "intentionality", that is, whether the organisation intentionally forgot existing knowledge or did so unintentionally. It is here that Tsang and Zahra's[16] insight is useful for organisations, as they clearly defined organisational unlearning as "deliberate discarding of routines"[16] (p. 1437), arguably providing a more useful interpretation for organisations.

Within the context of organisational failures, the concept of learning and unlearning from events becomes even more important, in order to try to prevent a recurrence of the negative incident or minimise the chance for it to happen again. This paper now turns to the specific aspects of learning and unlearning from failures.

2.2 Learning and unlearning from failures

In the work of Madsen and Desai[5], they argued that organisations tend to learn more from failures than successes. However, they also stated that success commonly stabilises organisations, whereas failures require a change of the status quo and challenge an organisation. This latter point was supported by Levitt and March[12] who identified that organisations tend to be more adaptable to learning from successful outcomes than negative actions that should have been avoided. In the work of Weinzimmer and Esken[14], they identified the difference between learning from successes, which requires an organisation to "exploit" this new knowledge, and learning from failures, which they argued requires organisations to engage in deeper "exploration" of the learning opportunity and ultimately offers greater potential benefits, although it is more painful for the organisation. Kunert[17] (p. 19) suggested that "while success and orderliness will arouse little drive to change existing routines, failure is more likely to foster the willingness and urgency to change, and, thus, stimulate action". Sitkin[2] agreed by stating that failure improves learning and resilience. In comparison, Toft and Reynolds[13] identified that learning and respective decision making should address both successes and failures.

Learning from failures mainly depends on the failure being detected and appropriately analysed[8]. Further, organisations often focus on evidence that supports their existing beliefs rather than accepting explanations that challenge the status quo of their companies[6], something that can be a key driver in organisational failure.

Thus, this discussion brings us to the following important questions: Under which conditions are organisations reluctant to learn from failures? How should we motivate them to learn from failures? What is organisational unlearning from failures and how can this help us to better support organisations?

2.2.1 Conditions potentially resulting in reluctance to learn from failure

When an organisational failure occurs due to poor conduct, omission or a need to accept responsibility, organisations can sometimes be seen to engage less willingly in the learning process. Similarly, if a mistake-intolerant culture exists, organisations will be unlikely to engage meaningfully in learning from failure[14]. It

could therefore be suggested that failures that include elements of organisational culpability or reputation risk, particularly due to human failure, could potentially create reluctance to engage and learn from failure; initial responses from BP's CEO to the Deepwater Horizon incident are an example of this[18,19].

2.2.2 How to motivate engagement with learning from failure

Organisational learning is seen by many as a strategic tool for organisational success and to enhance organisations' efficiency[20]. Furthermore, the importance of Knight and Pretty's[9] work is again underlined here through the clear demonstration of the link between how well organisations managed failures and their financial standing. By encouraging organisations to engage with such thinking, providing examples and by supporting use of simulations and scenarios, organisations should begin to understand the value of such learning.

2.2.3 What is unlearning from failures and how can it benefit organisations?

Unlearning from failures can be conceptualised at three levels. Labib[21] used the theoretical lens from Mahler[22] of categories of organisations unlearning to illustrate how this applies to disasters. In Mahler's view, there are three types of lessons that cause unlearning for organisations: (1) lessons not learned; (2) lessons learned only superficially; and (3) lessons learned and then subsequently unlearned. Labib[21] then provided three examples from disasters to illustrate how each of these types of unlearning has occurred.

There are plenty of examples where similar types of incidents keep occurring and repeated incidents occur because organisations have no memory since there are plenty of personnel changes that result in a "brain drain", using Kletz's[23] term. Our proposed modelling, as we will show below, provides a concise visual representation of the causal factors as a simplified mental model, and hence they are easier to remember rather than reading narratives of incident reports that usually run hundreds of pages. Such modelling approach also helps to establish the relationships among the causal factors, and hence provides a visual assessment of the vulnerability (weak or blind spots) in the system, thus informing our analysis of safety barriers.

2.3 Theories of failure

According to Labib[10], learning or unlearning from failure can be linked to a wide range of theories. Organisations can learn from failures through case studies or storytelling. However, they might be confronted with the concept of narrative fallacy[7]. This theory indicates that humans often search for explanations to the point where they manufacture them. Two other approaches address aspects of decision making. Organisations can either be too risk-averse or too risk-seeking. This is perfectly illustrated by the Swiss Cheese Model (SCM) introduced by Reason[24]. This conceptual model provides a simple illustration of the function in place where, if all guards fail, the whole system fails. The basic idea of SCM is that the layers/slices of cheese represents numerous system barriers that exist in the organisation in the form of procedures, check-lists and human checks for preventing hazards, while defects or loopholes in the system are represented by holes in the cheese layers. This model visualises incidents as the result of accumulation of multiple failures in barriers, or defences, represented as an alignment of holes in successive slices; hence, it is a simplified model to show the dynamics of accident causation. In other words, the failure occurred due to alignment of holes or simultaneous failures (loopholes in the system) of safety barriers. By safety barriers, this is analogous to the body's auto-immune system. More details about SCM and its evolution and limitations can be found in[25]. The model indicates cheese slices as barriers of protection concluding that the number of cheese slices identifies the level of risk aversion. Such modelling is easy to understand but its simplicity has also been criticised in that it does not represent adequately the relationship between different causal factors. Our proposed approach is intended to address this by providing more insight into causal relationships. Thus, the FTA, on the other hand, works its way to understand, or predict, what can cause the final unwanted event to happen, by working from the undesired event at the top of the FTA,

and drilling down to the most basic events that are associated together through logic gates to examine the relationships among causal factors.

Further, organisations can learn from specific and/or generic lessons of disasters. They can decide to adopt either the approach of Normal Accident Theory (NAT) or high-reliability theory. NAT, which was introduced by Perrow[26], claims that complexity and lack of prevention measures will unavoidably result in a disaster. Thus, the main claim of this argument is that accidents cannot be predicted or prevented and hence are "normal" and unavoidable. In contrast, the approach of High-Reliability Organisation (HRO) theory states that organisations can contribute to the prevention of disasters[27]. Hence, the emphasis here is not how accidents happen, but what successful organisations do to promote and ensure safety in complex systems. NAT and HRO have created two schools of thought in the literature related to failure theories. For a comprehensive and a balanced account of both schools of thought, the reader is directed to the works of Saleh *et al.*[28] and Rijpma[29].

3 CASE STUDIES

3.1 Rationale and methods applied

This paper focuses on two case studies of different backgrounds; one relates to security and the other to safety. The first case analyses the Virginia Tech Shooting in 2007 relating to failure management from a security perspective. As the incident has been extensively researched and highly influenced policymakers such as universities and government in the U.S. in improving regulations, policies, and laws[30-33], it is considered as a good example of learning from failure. The second case study concerns the Lion Air 610 airplane crash, which occurred in October 2018. This case study was chosen due to its current relevance and relation to aspects of safety. Further, the case study reflects on aspects of not fully learning from failures. This poor example of organisational learning, or "unlearning", is evident by the subsequent crash of Ethiopian Air shortly afterwards, which ultimately led Boeing to decide to halt the production of this type of aircraft. However, it needs to be acknowledged that the case studies are limited due to being secondary information collected by others and potentially being biased.

The two methods applied in this paper are FTA and RBD. Both methods complement each other as the RBD is constructed based on the structure of the FTA[10]. Combining both methods, as a hybrid model approach, can help to identify failures leading to a disaster, optimise the allocation of resources to address safety gaps and thereby mitigate consequences for future disasters[10].

An FTA identifies, models and evaluates the unique interrelationship of events leading to: (1) failure; (2) undesired events; or (3) unintended events. Those events are on the top of the FTA resulting from the input events indicated in the fault tree. Events are connected by "AND" and "OR" gates. An "OR" gate indicates that one or more events must occur to trigger the output event. In comparison, an "AND" gate is used when all failures indicated in connection with the output event must occur at the same time. The RBD gathers the events from the "AND" gate identified in the FTA into a parallel structure and the "OR" gate into a series one[10]. This method is used to identify vulnerabilities and gaps.

Addressing vulnerabilities can be an iterative and recursive process to help better understand and modify the original modelling in the form of a fault tree. This kind of analysis can be achieved either algebraically using the operational research method of minimum cut sets (a cut set is a combination of failure events, causing the top event in a fault tree) or by simply examining all possible failure scenarios of boxes in the RBD that will cause "cut-through" of the model. Such exercises build up critical mental problem-solving muscles, instead of simply reading a narrative of a report. It also helps to examine the logical combination of safety barriers that can mitigate against potential similar hazard.

Since fault trees are hierarchical structures, where the top of the tree is the undesirable incident (the disaster), and the bottom events are causal factors (root causes), there has been very little research on the middle levels of the hierarchical structure, especially the level just below the top event. This is particularly important as framing the problem dictates how the scope of the analysis will be developed into causal factors and subsequent recommendations. It has been proposed that the middle part of the fault tree can resemble the remit of the middle managers in humanitarian logistics organisations in terms of learning from rare events[34]. In addition, there has been a variation in terms of ways of classifying reasons of failure. For example, the factors can be grouped into "direct" and "indirect" causes such as the work of Labib and Read[1] in the analysis of Hurricane Katrina disaster. The same approach is followed in this paper as it helps to focus on both short and long terms recommendations. In addition, such taxonomy helps to realise both single- and double-loop learning.

Argyris (p. 68)[11] proposed the concept of single-loop learning, which can be defined as: "an error is detected and corrected without questioning or altering the underlying values of the system". It can be argued that this is related to the "direct" causes in an FTA model. Conversely, Argyris (p. 68)[11] defined double-loop learning as: "mismatches are corrected by first examining and altering the governing variables, and then reviewing the actions". This can be attributed to be among the "indirect" causes in the FTA modelling. Triple-loop learning has also been proposed in the literature, but this is beyond the scope of our paper. Other variations to classify failures can be as sociological and technical, or human and technology related issues.

3.2 Case study 1: Virginia Tech Shooting, USA, 2007

3.2.1 Background

One of the deadliest shootings in US history took place on 16 May 2007 at Virginia Tech University, Blacksburg. The shooting comprised two attacks that took place in two different locations on campus, which are illustrated in Figure 1[35]. The first shooting took place at 07:15 at West Ambler Johnston Hall dormitory, killing two people. Since the police associated the shooting with a domestic incident and assumed the attacker had already left campus, they did not shut down the campus[35]. About two and a half hours later, the gunman started the second shooting in Norris Hall building. Referring to witnesses, the gunman entered various classrooms and started randomly shooting at everyone[35]. As the gunman realised that the police were rushing into the building, he shot himself.

3.2.2 Logic and technical cause of failure

As this incident has been fully investigated and to reduce biased information, the causes of failure identified have been obtained from the amended Review Panel Report[36] of the Virginia Tech Shooting (2009). The failures have been identified as followed:

(1) Since his childhood, Cho exhibited mental health issues and received psychiatric treatment and counselling for selective mutism and depression. His mental instability worsened during his junior year at university as the university's care team failed to provide support to Cho.

(2) After he mentioned suicidal remarks to his roommates, his mental health was evaluated by psychologists. However, this was done inadequately resulting in insufficient treatment of Cho.

(3) With his history of mental health instability, he would not have been allowed by federal law to purchase the two guns which he used in the shooting. However, his data were never entered into the federal database used for background checks when purchasing firearms.

(4) There were communication errors among the university entities involved with Cho's situation and the incidents encountered by other students and faculty. Further, laws concerning the privacy of health and education records have been misinterpreted.

(5) Students and staff were not notified of the first shooting due to a misinterpretation of the incident and ineffective warning systems in place.

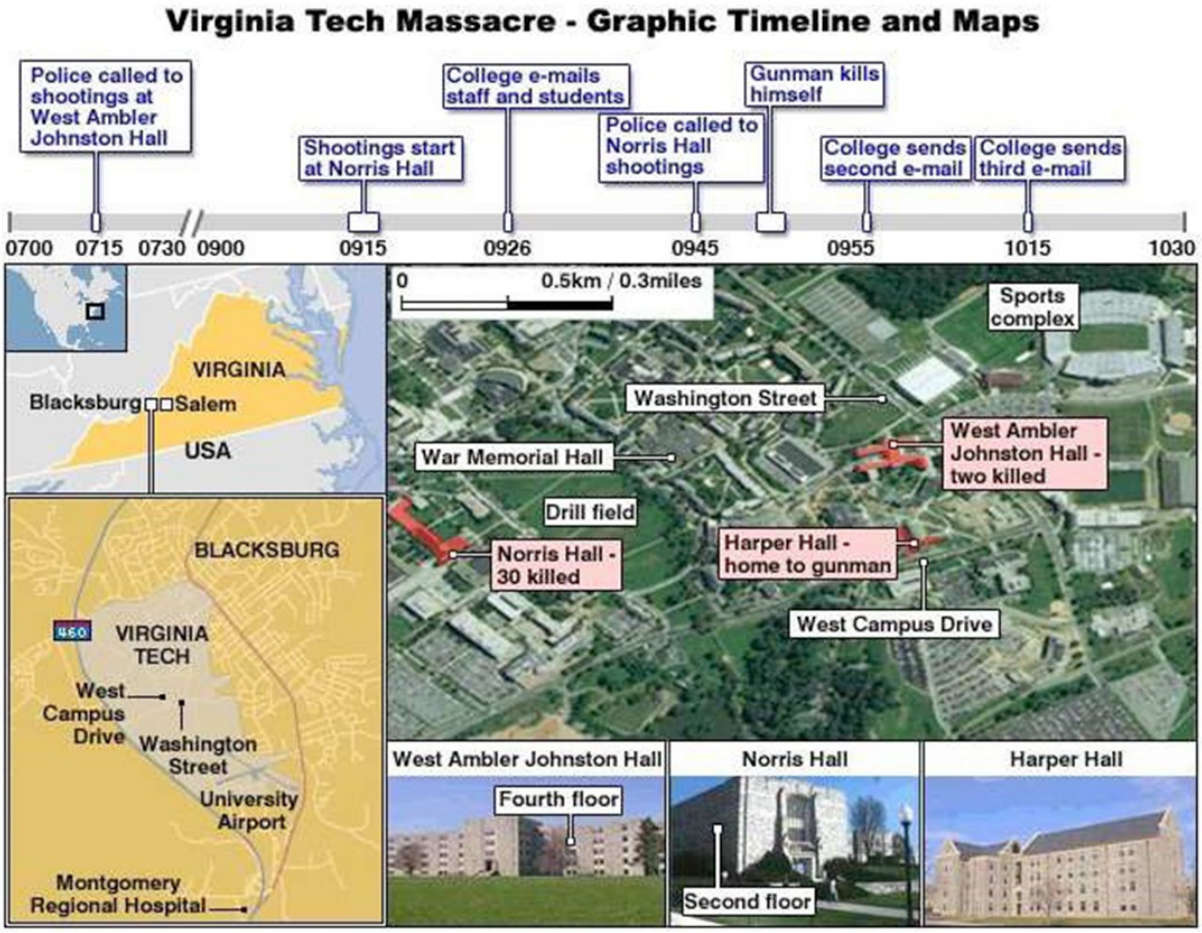


Figure 1. Graphic timeline of the incident and campus map (source: ref.[35])

(6) The university emergency plan did not include scenarios of a shooting and the assignment of a threat assessment team.
(7) The university had insufficient security systems such as cameras in dormitories and entrances to buildings as well as locks on classrooms.

3.2.3 Consequences

Twenty-seven students and five teachers were killed, and 17 people wounded by 23-year-old Virginia Tech student Seung-Hui Cho[35,36].

3.2.4 FTA and RBD

The FTA illustrated in Figure 2 identifies the direct causes and contributing factors leading to the Virginia Tech Shooting. The equivalent RBD is then cited in Figure 3, where every OR is series and every AND is parallel structure. The first direct cause defined is gunman Cho. Insufficient treatment, worsening mental health and a lacking supporting system were linked with an AND- gate to demonstrate their almost simultaneous occurrence. The insufficient treatment resulted from an inadequate evaluation of his mental health and lack of reporting. Another direct cause is linked to Cho's ability to purchase the guns due to reporting failures about the unstable mental health of Cho and a gap in gun laws. The third direct cause relates to the university and its lack of internal communication, the misinterpretation of the mental health and education laws, as well as the overall insufficient evaluation of Cho's situation. The contributing factors are linked with an AND-gate addressing the deficiency in responding to the incident and the lack of

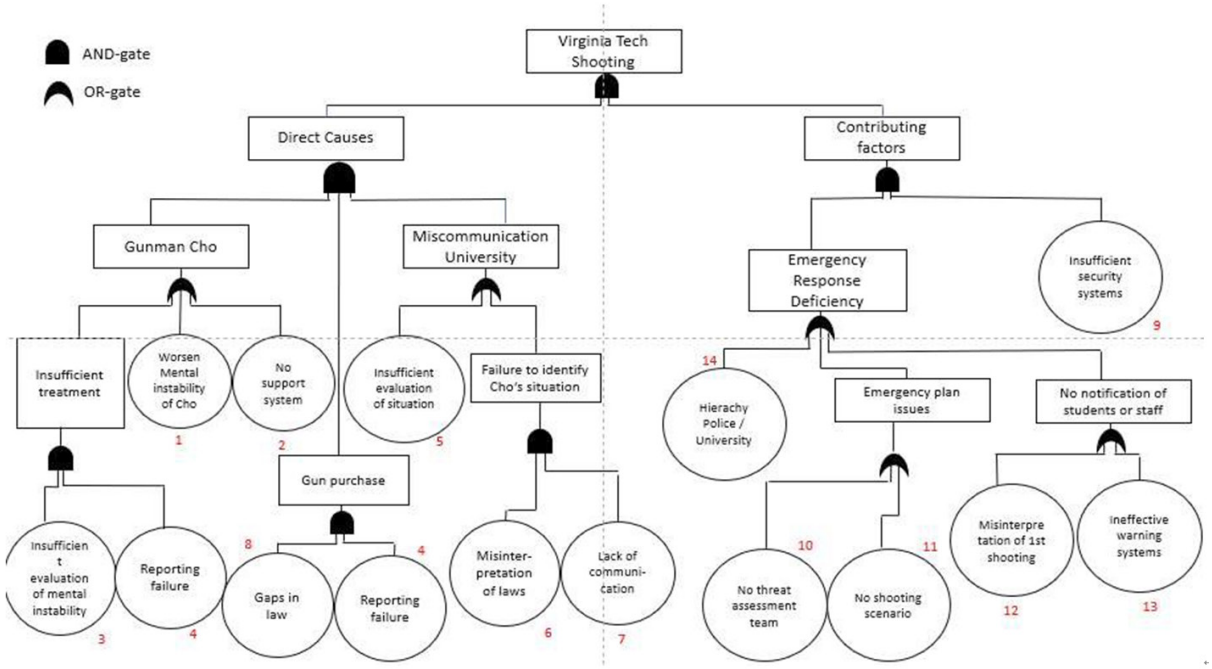


Figure 2. Fault Tree Analysis of the Virginia Tech Shooting

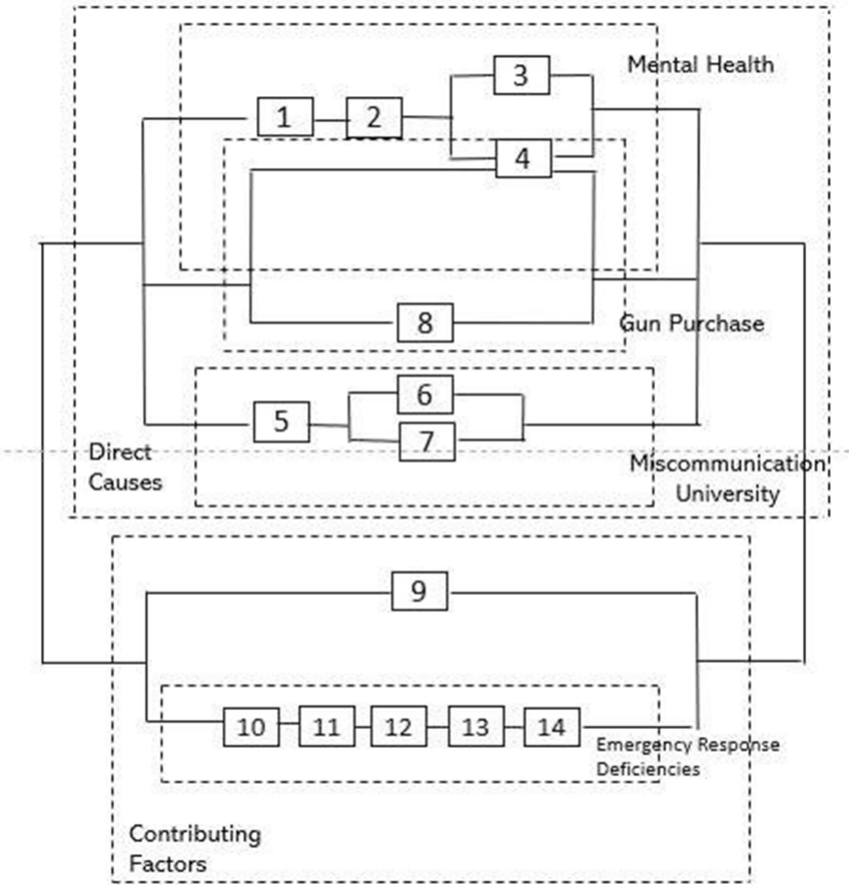


Figure 3. Reliability Block Diagram of the Virginia Tech Shooting

security systems around campus. Lacking information in the emergency plan, the decision to not notify staff and students as well as hierarchical issues between the police and the university all influenced the overall response to the incident. Causes 10 and 11 relate to the inefficient emergency plan. Causes 12 and 13 linked with an OR-gate lead to the decision to not notify people on campus.

3.2.5 Recommendations and generic lessons

Lack of information sharing, misinterpretation and insufficient evaluation of situations, as well as lack of communication, can be considered the main causes for the shooting. Universities and learning facilities should promote the sharing of knowledge internally, in particular when it concerns safety and health. Policies and laws should be carefully reviewed and unclarities identified and addressed. Further, learning institutions should allocate more resources to their care teams to improve support of students struggling with mental health issues. Emergency plans should be reviewed and updated regularly. They should implement a wide range of "what if" scenarios such as shootings. In addition, introducing a wide variety of communication channels and connecting them via one system can help to effectively warn students and staff in case of incidents.

3.3 Case study 2: Lion Air 610 Airplane Crash, 2018

3.3.1 Background

On 29 October 2018, a Boing Max 737-8 operated by Lion Air was scheduled to fly from Jakarta to Pangkal Pinang, as illustrated in Figure 4[34].

The scheduled departure time for flight LNI610 was 05:45. Two pilots, six crew members, and 181 passengers were on board the airplane[38]. Shortly after its departure, the pilots faced problems with indicating the altitude and the airspeed of the plane due to critical sensors registering different readings[39]. To identify the correct information, they contacted air traffic control. Shortly after, the airplane dropped over 700 feet as the aircraft's safety system MCAS, which was triggered by the falsified information of altitude, had forced the plane to nose down[39]. The pilots were able to correct and recover from the drop. However, the MCAS continued to push the plane's nose down even after pilots proceeded with counteractions. The plane went up and down more than a dozen times before disappearing from the radar[39].

3.3.2 Logic and technical cause of failure

As the investigation of this disaster is still ongoing and the final report is due to be released in August this year, the technical causes were based on the Preliminary Aircraft Accident Investigation Report (2018) of the Indonesian Transport Committee (KNKT)[38] and news articles related to the subject. The causes identified can be summarised as follows:

(1) The Angle of Attack (AoA) sensor falsely indicated that the airplane's nose was too high and that the airplane was stalling. The information obtained by the AoA sensor triggered the automatic safety system Maneuvering Characteristics Augmentation System (MCAS), which forced the airplane's nose down[39].

(2) The MCAS overrode the pilots' response as they were trying to correct the problem by lifting the plane's nose back up[38].

(3) Pilots were not aware of the existence of MCAS. They seemed to have not received any training for this feature and no information was added in the manual[39].

(4) The airplane did not have the optional warning light that would have indicated the problem's root. Issues with previous flights of the airplane and response actions by pilots to overcome those have not been carefully evaluated and communicated properly[35].

(5) The MCAS had a poor system redundancy by being able to be triggered by a single sensor, even though there are two AoA sensors on every airplane[40].

Figure 4. Expected route of Lion Air Flight 610 and the location of the crash (source: ref.[37])

3.3.3 Consequences

The airplane crashed at about 5000 feet with a speed of 450 miles per hour into the Java Sea[39]. All people on board died in the airplane crash. It is the second deadliest airplane accident in Indonesia.

3.3.4 FTA and RBD

The FTA illustrated in Figure 5, and its equivalent RBD in Figure 6 identify the direct and indirect causes leading to the Lion Air 610 crash, and the overall vulnerability analysis. A technical error was defined as the first direct cause. The AoA sensor providing wrong information about altitude and speed as well as triggering the MCAS, the MCAS overriding the pilots' response and the missing optional warning light were all indicated as the root causes for the technical failure of the machine. As they occurred simultaneously (and they were both needed and sufficient for the outcome to be realised), they were linked with an AND-gate. Another failure included the response of the pilots whose countermeasures were inefficient. Further, they failed to identify the problem because MCAS was a new feature, pilots did not

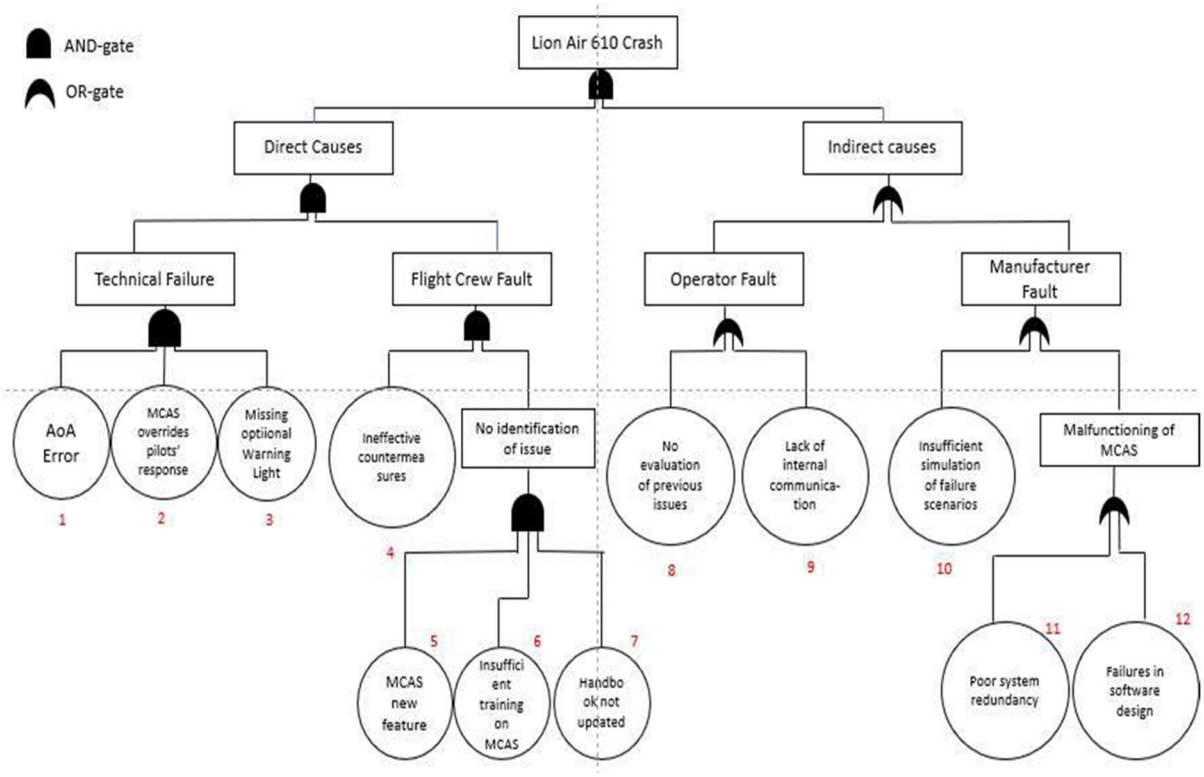


Figure 5. Fault Tree Analysis of the Lion Air 610 airplane crash

receive any training on MCAS and the handbook was not updated. Indirect causes are linked to failures of the operator Lion Air and the manufacturer Boing. Failures by the operator included that the airline did not evaluate the previous issues in relation to the AoA sensor and the MCAS as well as the missing internal communication. On the other side, failures in the software and poor system redundancy resulted in the malfunctioning of the MCAS. Additionally, simulations did not include potential failure scenarios. These causes are linked to Boing.

3.3.5 Recommendations and generic lessons

The main lessons to be learned are that lack of communication and lack of training are often the root causes of airplane crashes. By informing and training pilots appropriately about new features implemented in the airplanes, failures in responding can be reduced. Further airplane manuals should be updated regularly. By sharing information on near misses internally and improving communications, airlines can help to mitigate disasters.

In addition, airlines should carefully consider the optional safety features that can be purchased from the manufacturers for a relatively low cost. Manufacturers such as Boing should carefully evaluate potential failures of software and design before implementing new features. To increase the redundancy of the MCAS, Boeing should link the feature to two sensors instead of just one. Depending on the outcomes of investigations, airlines and authorities should work collaboratively to address the safety of passengers and flight crews - even if that means the grounding of airplanes and financial losses.

4. DISCUSSION

Implementing the FTA and RBD in the analysis of the two case studies in this paper helped to understand the root causes and to recognise the vulnerability gaps. In a wider context, the methods identified

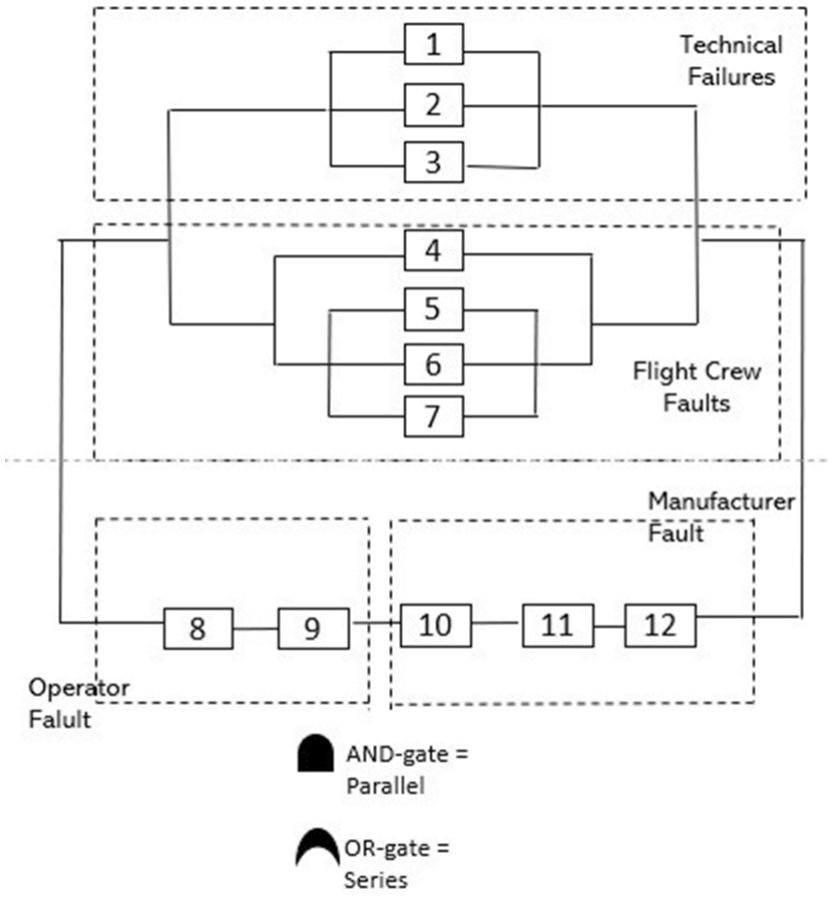


Figure 6. Reliability Block Diagram of Lion Air 610 airplane crash

similarities between the two cases in relation to failures associated with the disasters. These include failures in information sharing, communication and lack of knowledge. Further, both cases presented unlearning from near misses before the disasters occurred. However, the cases differ in their level of learning from the disasters. In the case of the Virginia Tech Shooting, the failures have been detected, analysed and lessons learned implemented nationwide through updated regulations, laws and policies. However, in the case of the Lion Air 610 crash, the failures have only been partly detected as investigations have not been concluded yet. Hence, the organisations involved have the just begun with learning from the failure. This is also reflected in the crash of the Ethiopian Airlines last month (March, 2019), which was identified to have similar causes of failure seemingly not having been completely addressed yet.

Regarding the case of the Virginia Tech Shooting, the RBD identified gaps in the emergency response. Referring to the Swiss Cheese Model, it can be indicated that a lack of barriers resulted in the emergency response to be the most vulnerable part of the system. In comparison, the RBD for the Lion Air 610 crash identified gaps in the design of the MCAS and internal communication among airline entities.

As the case study of the Lion Air 610 crash mainly addresses systematic failures, the FTA proved to be easily applicable. However, the FTA of the Virginia Tech Shooting showed limitations due to the complexity of the case. The authors are aware that Cho's personality might be seen more as a symptom rather than a cause. However, in this case, it was interpreted as a cause since this particular incident would not have happened without Cho and the root causes were related to issues in communication and missing supporting features.

There are two main perceived criticisms of our proposed approach. The first relates to the limited value added, whereas the latter is concerned with assumption of simplicity of cause and effect. Thus, the first point is about the true added value of using FTA and RBD, which is claimed to be limited, as these tools did not discover new lessons. In response to this criticism, one can say that FTA helps to organise relationships between factors and such mental model in the form of a diagram might be easy to recall and hence saves much time in going through the large number of pages that is typical of any incident investigation report. Regarding the second point, of being constrained by a strict cause and effect relationship that may not capture complexity of the incident, one can argue that such limitation is in a way a blessing, as, by following such logical way of thinking that is strictly relying on just a couple of logic gates, this approach provides a rational way of thinking to reach logical conclusions. It is not the intention of this paper to offer new insight into these tragic cases but instead to present approaches that will help to develop and support organisational learning, highlighting the diversity of cases for application of the approach and, crucially, offering a simplified method to capture key information rather than extended narrative, which can dilute learning for organisations.

In conclusion, the results of integrating the information obtained through the methods of FTA and RBD can support the various stakeholders of the events to appropriately allocate their resources, improve processes in terms of standard operating procedures and routines and thereby mitigate future disasters. Thus, organisational learning is stimulated and organisational resilience can be improved. Again, we stress here that learning implies change of behaviour to avert similar incidents from occurring and accordingly unlearning implies abandoning such practices. However, the learning process varied between both cases due to having investigations closed and failures detected in the case of the Virginia Tech Shooting and ongoing investigations in the case of the Lion Air 610 crash.

Thus, can the analysis be the motivation for organisational learning? If we define organisational "learning" from failures as consisting of three main streams: (1) feedback from users to design; (2) use of advanced modelling and analysis tools; and (3) incorporation of multidisciplinary and generic lessons, as proposed by Labib[10], then, using this lens, the answer is yes. Both cases showed that there is potential to learn from the disasters as: (1) users' feedback to design through specific lessons and actions identified; (2) the integration of failure analysis tools such as FTA and RBD; and (3) the generic lessons learned have been applied in both safety and security domains.

DECLARATIONS

Acknowledgments

We would like to acknowledge the feedback comments received by the reviewers.

Authors' contributions

Conceptualisation, data analysis, formal analysis and writing: Schmidt B
Advise on investigation, methodology and supervision: Labib A
Validation, visualisation, review and editing: Hadleigh-Dunn S

Availability of data and materials

Not applicable.

Financial support and sponsorship

None.

Conflicts of interest

All authors declared that there are no conflicts of interest.

Ethical approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Copyright

REFERENCES

1. Labib A, Read M. Not just rearranging the deckchairs on the Titanic: Learning from failures through Risk and Reliability Analysis. Safety Sci 2013;51:397-413.
2. Sitkin S. Learning through failure: the strategy of small losses. Res Organizational Behave 1992;14:231-66.
3. Cannon MD, Edmondson AC. Failing to learn and learning to fail (Intelligently): how great organizations put failure to work to innovate and improve. Long Range Planning 2005;38:299-319.
4. Edmondson AC. Strategies for learning from failure. Harvard Bus Rev 2011;89:48-55.
5. Madsen PM, Desai V. Failing to learn? The effects of failure and success on organizational learning in the global orbital launch vehicle industry. Academy Management J 2010;53:451-76.
6. Smith D, Elliott D. Exploring the barriers to learning from crisis: organizational learning and crisis. Management Learning 2007;38:519-38.
7. Taleb N. The Black Swan. London, England: Penguin Books Ltd; 2007.
8. Fortune J, Peters G. Learning from failure: the systems approach. Chichester: John Wiley & Sons Ltd; 1995.
9. Knight R, Pretty D. The impact of catastrophes on shareholder value. The Oxford Executive Research Briefings. Templeton College: University of Oxford; 1997.
10. Labib A. Learning from failures: decision analysis of major disasters. Elsevier Butterworth-Hein; 2014.
11. Argyris C. On organizational learning. 2nd ed. Oxford: Blackwell Publishing; 1999.
12. Levitt B, March JG. Organizational learning. Annu Rev Sociol 1998;14:319-38.
13. Toft B, Renolds S. Learning from disasters: a management approach. 3rd ed. Basingstoke: Palgrave Macmillan; 2005.
14. Weinzimmer LG, Esken CA. Learning from mistakes: how mistake tolerance positively affects organizational learning and performance. J Appl Behav Sci 2017;53:322-48.
15. Starbuck WH. Unlearning ineffective or obsolete technologies. Int J Technol Management 1996;11:725-37
16. Tsang EWK, Zahra SA. Organizational unlearning. Human Relations 2008;1435-62.
17. Kunert S. Strategies in failure management: scientific insights, case studies and tools. Cham: Springer International Publishing; 2018.
18. Pranesh V, Palanichamy K, Saidat O, Peter N. Lack of dynamic leadership skills and human failure contribution analysis to manage risk in Deep Water Horizon oil platform. Safety Sci 2016;92:85-93.
19. Balmer JMT, Powell SM, Greyser SA. Explicating ethical corporate marketing. Insights from the BP deepwater horizon catastrophe: the ethical brand that exploded and then imploded. J Business Ethics 2011;102.
20. Saadat V, Saadat Z. Organizational learning and as a key role of organizational success. Procedia Social Behav Sci 2016;230:219-25.
21. Labib A. Learning (and unlearning) from failures: 30 years on from Bhopal to Fukushima an analysis through reliability engineering techniques. Process Safety Environmental Protection 2015;97:80-90.
22. Mahler JG. Organizational learning at NASA: the Challenger and Columbia accidents. Georgetown University Press; 2009.
23. Kletz TA. Lessons from disaster: how organizations have no memory and accidents recur. IChemE; 1993.
24. Reason J. Managing the risk of organizational accidents. Aldershot, Hants, Ashgate; 1997.
25. Reason J, Hollnagel E, Paries J. Revisiting the Swiss cheese model of accidents. J Clini Engineering 2006;27:110-5.
26. Perrow C. Normal accidents: living with high-risk technologies. New York: Basic Books; 1984.
27. Rochlin GI, La Porte TR, Roberts KH. The self-designing high reliability organization. 1987. Reprinted in Naval War College Review 1998;51:17.
28. Saleh JH, Marias KB, Bakolas E, Cowlagi RV. Highlights from the literature on accident causation and system safety: review of major ideas, recent contributions, and challenges. Reliability Engineering System Safety 2010;95:1105-16.
29. Rijpma JA. Complexity, tight-coupling and reliability: connecting normal accidents theory and high reliability theory. J Contingencies Crisis Management 1997;5:15-23.
30. Vieweg S, Palen L, Liu SB, Hughes AL, Sutton JN. Collective intelligence in disaster: Examination of the phenomenon in the aftermath of the 2007 Virginia Tech shooting. Boulder, CO: University of Colorado; 2008.
31. Palen L, Vieweg S, Liu SB, Hughes AL. Crisis in a networked world: Features of computer-mediated communication in the April 16, 2007, Virginia Tech event. Social Sci Computer Rev 2009;27:467-80.
32. Hong JS, Cho H, Lee AS. Revisiting the Virginia Tech shootings: an ecological systems analysis. J of Loss and Trauma 2010;15:561-75.
33. Virginia Tech Review Panel. Mass shootings at Virginia Tech. Addendum to the report of the review panel; 2009. Available from: https://scholar.lib.vt.edu/prevail/docs/April16ReportRev20091204.pdf [Last accessed on 24 Apr 2020]
34. Labib A, Hadleigh-Dunn S, Mahfouz A, Gentile M. Operationalising learning from rare events: framework for middle humanitarian

operations managers. Production Operations Management 2019;28.
35. Caruso K. Virginia Tech Massacre. Available from: http://www.virginiatechmassacre.com/index.html [Last accessed on 24 Apr 2020]
36. CNN (2018). Virginia Tech Shooting Fast Facts. Available from: https://edition.cnn.com/2013/10/31/us/virginia-tech-shootings-fast-facts/index.html [Last accessed on 24 Apr 2020]
37. BBC. Lion Air JT610 crash: What the preliminary report tells us. Available from: https://www.bbc.co.uk/news/world-asia-46373125 [Last accessed on 24 Apr 2020]
38. Komite Nasional Keselamatan Transportasi. Preliminary Aircraft Accident Report. Available from: https://reports.aviation-safety.net/2018/20181029-0_B38M_PK-LQP_PRELIMINARY.pdf [Last accessed on 24 Apr 2020]
39. Gröndahl M, McCann A, Glanz J, Migliozzi B, Syam U. In 12 minutes, everything went wrong. How the pilots of Lion Air Flight 610 lost control. Available from: https://www.nytimes.com/interactive/2018/12/26/world/asia/lion-air-crash-12-minutes.html [Last accessed on 24 Apr 2020]
40. Learmount D. Lion Air lessons. Available from: https://www.aerosociety.com/news/lion-air-lessons/ [Last accessed on 24 Apr 2020]

Xu *et al. J Surveill Secur Saf* 2020;1:16-33
DOI: 10.20517/jsss.2020.04

Journal of Surveillance, Security and Safety

Original Article

Open Access

Check for updates

Big data analytics of crime prevention and control based on image processing upon cloud computing

Zheng Xu[1], Cheng Cheng[1,2], Vijayan Sugumaran[3]

[1]School of Computer Science, Shanghai University, Shanghai 201142, China.
[2]Cernter of IoT, The third research institute of the ministry of public security, Shanghai 200335, China.
[3]Department of Decision and Information Sciences, Oakland University, Rochester, MI 48309, USA.

Correspondence to: Prof. Zheng Xu, School of Computer Science, Shanghai University, Shanghai 201142, China.
E-mail: zhengxu@shu.edu.cn

How to cite this article: Xu Z, Cheng C, Sugumaran V. Big data analytics of crime prevention and control based on image processing upon cloud computing. *J Surveill Secur Saf* 2020;1:16-33. http://dx.doi.org/10.20517/jsss.2020.04

Received: 5 Mar 2020 **First Decision**: 21 Apr 2020 **Revised**: 5 Jun 2020 **Accepted**: 12 Aug 2020 **Available online**: 12 Sep 2020

Academic Editor: Yelena Yesha **Copy Editor**: Cai-Hong Wang **Production Editor**: Jing Yu

Abstract

Aim: Current crime behavior observation has the problem of not being real time, thus criminal behavior cannot be promptly controlled. To improve the control of criminal behavior, this study was based on cloud computing image processing, and adopted data mining for criminal behavior.

Methods: This study obtained many criminal behavior characteristics through data collection and combined the rapid response capability of cloud computing to adopt data processing. In addition, to improve the accuracy of criminal behavior recognition, the identification method for criminal behaviors in selected populations was studied, and the image processing technology was combined to identify individual crimes and subject segmentation.

Results: Our work used statistical methods to collect the characteristics of criminal behavior, and we designed experiments to verify the effectiveness of the algorithm. The experimental research shows that the algorithm has high accuracy in identifying abnormal behavior.

Conclusion: The research shows that the accuracy of the algorithm for identifying abnormal behavior is relatively high, and it has high practical value, which can meet the accuracy and real-time requirements of security systems.

Keywords: Cloud computing, cloud storage, data mining, crime prevention and control, image capture

1 INTRODUCTION

Current crime behavior observation has the problem of not being real time, thus criminal behavior cannot be promptly controlled. To improve the control of criminal behavior, this study was based on cloud computing image processing, and adopted data mining for criminal behavior. his study obtained many criminal behavior characteristics through data collection and combined the rapid response capability of cloud computing to adopt data processing.

Criminal behavior has always affected the stability of societies, and it is difficult to control crimes in real time through monitoring and observation. In recent years, with the continuous advancement of society, the level of material and spiritual living has continuously improved, and people have begun to pay more attention to the safety of their lives and property. As a settlement of the population, the city has a more urgent need to ensure that life and property are not invaded. Due to the high population density and complex personnel structure, urban management becomes more and more difficult, and various public security incidents are more likely to occur. How to effectively control crimes is an important research content of current social management, and the analysis of big data crime behavior is an effective way.

The abnormality detection object in this paper is a random group in ordinary public places, i.e., the people in the crowd are not unified and purposeful. The direction of movement of the crowd is irregular when there is no abnormality. This paper uses the anomaly detection algorithm with improved acceleration characteristics to detect the abnormal escape behavior of the crowd. Firstly, the motion vector field is processed by block processing, then the image is filtered to reduce the influence of noise, and the mean filtering is adopted, and then the algorithm is used to extract the foreground of the image sequence. This kind of operation not only facilitates the extraction of motion features, but also reduces the disadvantages of large computational complexity. The experimental research shows that the algorithm has high accuracy in identifying abnormal behavior and has high practical value, which can meet the accuracy and real-time requirements of the security system.

2 RALATED WORK

In recent years, with the continuous deepening of the research on abnormal behavior detection of people, related technologies have gradually matured. According to the degree of occurrence of abnormal events in the surveillance video picture, they can be divided into global abnormalities and local abnormalities. A global exception means that an abnormal behavior occurs in the entire monitored image (even if part of the area is normal). A local anomaly is an abnormal behavior in which a local area in a surveillance video is distinguished from a surrounding area. In addition, the focus of the research method is different for the different characteristics of global anomalies and local anomalies[1]. Scholars have proposed some more classical methods to promote the continuous development of population anomaly detection technology.

For the global exception, it is necessary not only to detect whether the scene is abnormal, but also to judge the start and end of the abnormality and the intermediate transition phase. In general, the global anomaly detection method is to analyze the change of the event based on the motion estimation of the entire video picture.

Chen and Huang[2] proposed a graph analysis algorithm based on eigenvalues. In their paper, each isolated area in the video picture is regarded as a vertex, and the whole group is regarded as a picture. Moreover, topological changes are simulated by local features (based on feature subgraph analysis and trigonometric transformation) and global features (time, *etc*.). Finally, the simulated topological variation characteristics are used to analyze the presence or absence of abnormal events in the population.

Wu *et al*.[3] proposed a population abscess detection method based on Bayesian model. The method is mainly based on the optical flow extraction motion feature, directly simulates the crowd movement with the conditional density function, and uses the Bayesian classification formula to determine whether there is an abnormal crowd escape event. Their experiments show that this method can accurately detect the abnormal behavior of the crowd, but it is not appropriate for the crowded scene. The global exception can only determine whether there is an abnormality in the monitoring screen. In actual applications, it is often necessary to locate the specific location where the abnormality occurs. Based on this, many scholars have proposed a local anomaly detection method, which usually divides the video picture into many small areas and locates the specific location of the abnormality through the abnormal situation of all small areas.

Biswas *et al*.[4] proposed an abnormal event detection method based on the social force model. In the social force model, everyone in the group is simultaneously influenced by the individual's desired force and social interaction. The direction in which the individual expects the force indicates the direction of the movement desired by the individual, and the direction of the social interaction indicates the direction in which the environment, the pedestrian, *etc*. influence the individual. The direction of the two forces is the actual direction of the individual in the crowd. A particle flow calculation method based on optical flow is proposed to calculate the interaction force and solve the problem that the force calculation in the crowd is difficult due to serious crowding and occlusion. After modeling the crowd, the LDA model is used to determine the normal and abnormal frames in the video.

Cong *et al*.[5] proposed anomalous event detection for the sparse reconstruction cost (SRC) model. In this method, three different types of multi-scale optical flow histograms are extracted for different local anomalous behaviors and global anomalous behaviors. After extracting features from normal frame images, the feature sets are composed of the extracted features, and the redundant information in the dictionary set is eliminated by an optimized method to form an optimal dictionary set. At the same time, the method uses the best dictionary set to judge whether each frame of the test set has abnormal behavior through the SRC method. Li *et al*.[6] proposed a mixed dynamic texture model (MDT) to detect anomalies in dense populations. The MDT model performs a time-space block on a video sequence to detect whether an exception has occurred. In the time anomaly detection, the local distribution of the image intensity is simulated based on the foreground extraction of the Gaussian mixture model. In the spatial anomaly detection, the local region where an abnormality may occur is discriminated based on the principle of image saliency (the spatial position of the abnormality is higher than a certain threshold).

Song *et al*.[7] proposed a chaotic invariant algorithm. This paper proposes a human flow model that is applicable in both structured and unstructured scenarios. First, the particles are advected based on the optical flow, and the trajectory of the human flow is represented by the trajectory of the aggregated particles. Then, all representative trajectories are quantified using a blunt invariant, and a model is trained using the quantized chaotic set. Finally, the maximum likelihood estimation is used to identify abnormalities and normal behaviors in the population.

Kratz and Nishino[8] proposed a framework for modeling local temporal and spatial motion behaviors in dense population scenarios based on Hidden Markov Models (HMM). In the training phase, the temporal relationship between local motion patterns is extracted by a distribution-based HMM, and the spatial relationship is modeled by a coupled HMM. In the test phase, the anomaly event is the statistical deviation in the same scene in the video sequence. Their experiments show that HMM is suitable for analyzing more intensive scenes, and an HMM is established for each small area, which indicates that the method is only appropriate for a limited variety of normal behavior.

Despite the increasing popularity of video surveillance equipment and the maturity of intelligent surveillance technology, real-time problems are faced in practical applications. At present, although many

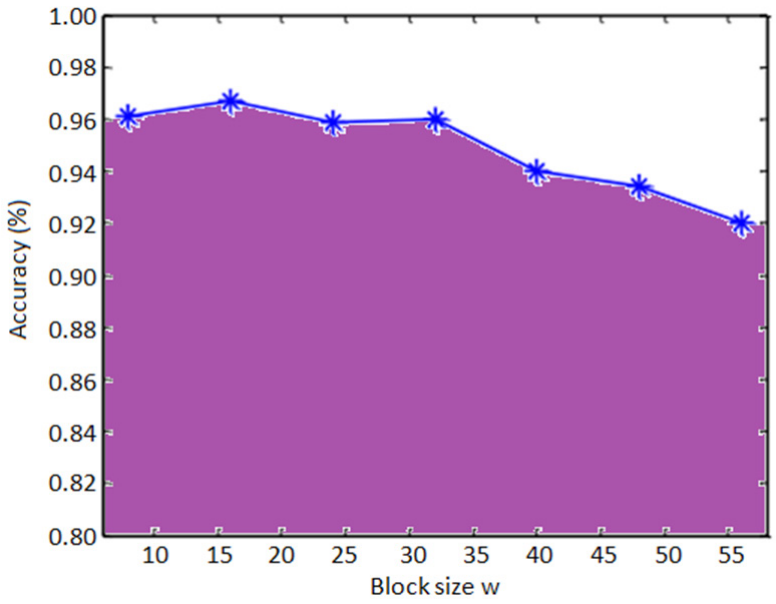


Figure 1. The algorithm's detection results are accurate under different block sizes[9]

algorithms achieve high accuracy on public datasets, it is difficult to meet real-time requirements due to, e.g., computational complexity. Thus, based on data mining, this study combined cloud computing image processing technology for real-time crime behavior recognition.

3 METHODS

3.1 Population abnormal behavior detection

To analyze the behavior of the crowd, each field is divided into a set of patches (blocks), and the population density in the image field determines the size of the block for the field. The purpose of this statistical partitioning of images by patches is mainly to better avoid interference from other factors. The proposed detection algorithm was tested on the public dataset UMN (University of Minnesota) to find the best block size. Figure 1 shows that, when the scale is higher than 40, the accuracy becomes larger with the block size, and the progress is basically stable. When the scale is higher than 40, the accuracy begins to decrease. However, it should be noted that, if the block size is too large, the motion state cannot be described well. Therefore, the block size selected in this section is 24×24. The schematic is shown in Figure 2[10].

This paper uses a foreground extraction algorithm based on K-means clustering. When the crowd escapes in the scene, the acceleration of the crowd changes. Therefore, this paper uses the magnitude of the acceleration to extract the foreground. When the amplitude of a position in the scene is greater than the threshold, the position is judged as the foreground. In this paper, the K-means algorithm, with $k = 2$, is used to calculate the threshold value by randomly calculating the partial vector selected from the sample set[11].

Since the performance of abnormal behaviors is diverse, the definitions of exceptions are different in different scenarios. For any video image describing the behavior of the crowd, the sequence is described by the acceleration and velocity vectors. The basic steps of detecting the abnormal escape behavior of the crowd are as follows: (1) a given video image is smoothed by an image preprocessing method to remove noise; (2) the velocity and acceleration vectors are calculated using a modified acceleration algorithm for the processed grayscale image sequence. The improved acceleration algorithm combines the changes in the characteristics of the movement of the crowd and the changes in the distribution of the population.

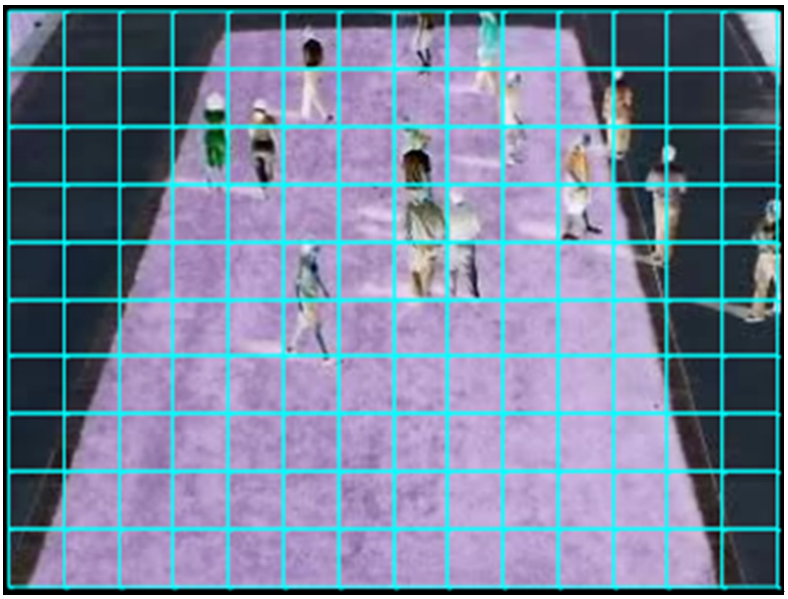


Figure 2. Schematic diagram of block processing of video frames

The neighborhood size for solving the acceleration in the experiment was 5 × 5; (3) regarding anomaly classification, most algorithms are based on empirical thresholds. The threshold ε in this paper is mainly the continuous actual detection and analysis of different datasets; and (4) an optimal behavior determination based on an optimal threshold is applied to a given video sequence. If it is greater than the threshold, it is considered that there is an abnormal situation; otherwise, it is considered a normal video frame. Represent threshold as Ai. The detection process can be described as[12]:

$$F(A_i)=\begin{cases} normal, A_i < \varepsilon \\ anomaly, A_i \geq \varepsilon \end{cases} \quad (1)$$

In real video images, the crowd leisurely walks in the direction they want to go. However, when there are unexpected events, people will run around because of psychological panic. At this point, people usually get away from danger quickly. This results in a very significant change in the speed of the human body, thus the range of motion of the moving object is also large.

It should be noted that both normal behavior and abnormal behavior are continuous; they cannot occur at a certain moment, and then disappear immediately. Therefore, occasional short-lived anomalous frames that appear in normal video frames are still considered normal video frames. Conversely, several frames of normal frames that occur by chance in an abnormal video frame are still determined as abnormal video frames[13].

3.2 Escape center

The anomaly detection object in this paper is a random group in ordinary public places. That is to say, the movement direction of the crowd is irregular when there is no abnormality. Usually, sudden abnormal situations for ordinary people will bring fear to people and then move away from the point where the anomaly occurs. Therefore, according to this principle, the idea of escaping from the center is introduced, and the place where the abnormality may occur is determined by determining the escape center. As shown in Figure 3, one or more anomalies may occur in a scene. The squares in the figure represent moving targets, the arrows on the squares indicate the direction of motion of the moving targets, and the blue dots represent the locations where anomalies may occur[14].

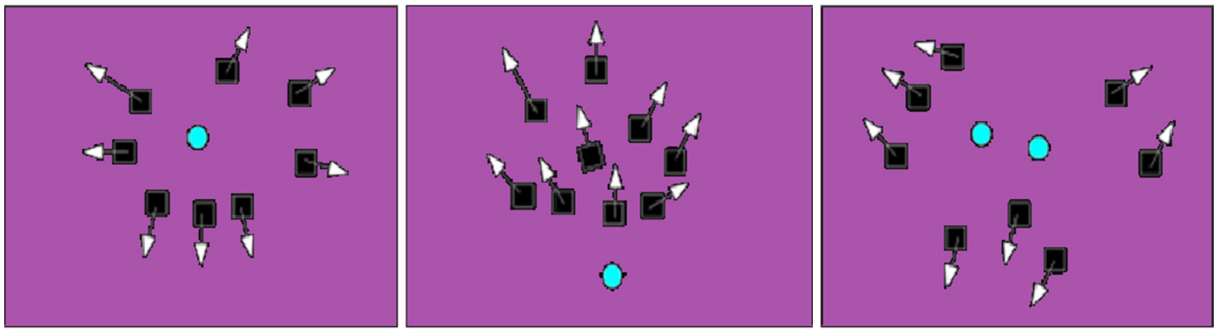


Figure 3. Schematic diagram of the escape center

3.3 Anomaly localization algorithm based on single escape center

According to the introduction to the escape center in the previous section, the escape center can be obtained from the movement information of the crowd in the specific case studied in this paper. The population studied in this paper is mainly people who have fled after being affected by emergencies in public places. Moreover, anomaly detection and localization for this situation can be implemented by combining the acceleration features with the escape center. First, the acceleration feature extraction is performed through the optical flow field of the video image to determine whether an abnormality has occurred. If there is a possibility of an abnormality, the escape center position is calculated. Moreover, it is determined that the center position can be escaped by using the intersection of all the acceleration vector inverse extension lines obtained by detecting the position and direction information of the acceleration vector when the abnormality is detected and using K nearest neighbor search. Simulation experiments show that the method can well determine the possible location of anomalies in simple places where anomalies occur[15].

The anomaly localization algorithm in this paper is based on the analysis of the motion vector of the moving target in the abnormality detection process to locate the position where the abnormality may occur. The process of locating mainly consists of counting all the intersections of the inverse extension lines of the motion vector and using the K nearest neighbor search method to determine the most likely abnormal position[16].

The algorithm is as follows: (1) by framing the video, the image sequence is obtained, and then the motion feature vector is extracted; (2) each frame of the image is subjected to improved acceleration signature calculations, and an improved acceleration threshold is obtained through a number of experiments. The acceleration vector of this frame image is retained if the improved acceleration value is greater than the threshold; (3) according to the obtained acceleration vector $A(a_x, a_y)$ of the image, the corresponding one-dimensional linear equation parameter is calculated. When ignoring special cases, the parameter expression is as follows[17]:

$$\begin{aligned} k_l &= a_{y(i,j)} \big/ a_{x(i,j)} \\ b_l &= i - k_l \times j \end{aligned} \tag{2}$$

where i and j represent the positions of the corresponding pixels in the image; (4) According to the linear equation of the acceleration vector, the intersection point $P = \{p_1, p_2, \ldots p_s\}$ is calculated. The main calculation is the intersection of all the two straight lines, and the mathematical derivation is as follows[18]:

$$\begin{aligned} x &= (b_2 - b_1)/(k_1 - k_2) \\ y &= \left((b_2 - b_1)/(k_1 - k_2)\right) \times k_1 + b_1 \end{aligned} \tag{3}$$

(5) there are many repetitions in the process of calculating the intersection set. The intersections that do not match the actual situation are collectively referred to as wild intersections. Therefore, to improve the

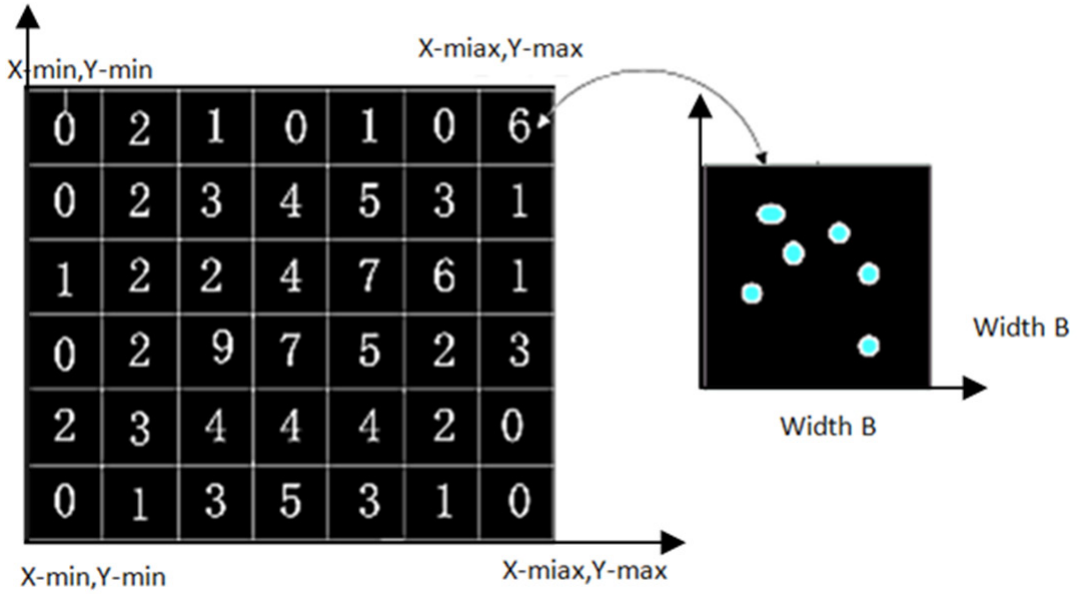


Figure 4. Block diagram of the principle of graphic method

accuracy of the algorithm, it is necessary to remove wild intersection points from the intersection point set; and (6) by further analyzing the intersection set, the escape center is determined. Since the intersection set referred to in this paper belongs to a simple dataset, the K nearest point search method can be used.

The K nearest point search method is also known as the K nearest neighbor search method. In the process of determining the escape center, this search method calculates the distance between the intersections in the neighborhood, selects the Kth minimum distance in the distance set of each intersection, and compares it with other intersections to determine whether it is the escape center. This search method is suitable for the classification of rare events. Since the escape center in this section is an infrequent event, and this search method is easy to implement without training, this paper uses the K nearest neighbor search method to determine the escape center of abnormal populations[19].

In determining the escape center, the determination of the intersection of the lines of the improved acceleration vector in the image plays a very important role in the overall algorithm. In the process of determining the intersection, all intersections are calculated and the wild intersections are removed. The calculation method of the intersection point is introduced above, and the removal process of the intersection point is introduced here.

The wild intersection removal method in this paper adopts the graphical removal method, as shown in Figure 4. The specific steps of the algorithm are as follows[20]: (1) the intersection set $P = \{p_1, p_2, \ldots p_s\}$ solved by the improved acceleration vector is taken as an input; (2) the overall scope of the graphical search is determined; and (3) a small search window is designed to count the number of intersections in the search range determined in the previous step. The size of the small search window is determined before the start of the detection and is mainly determined according to the number of moving objects in the image and the size of the image.

The graphic method removes the wild intersection points mainly based on the number of intersection points in the search window to make the intersection point, which is mainly determined by the particularity of the research object of the algorithm. Moreover, the algorithm is mainly used to locate the

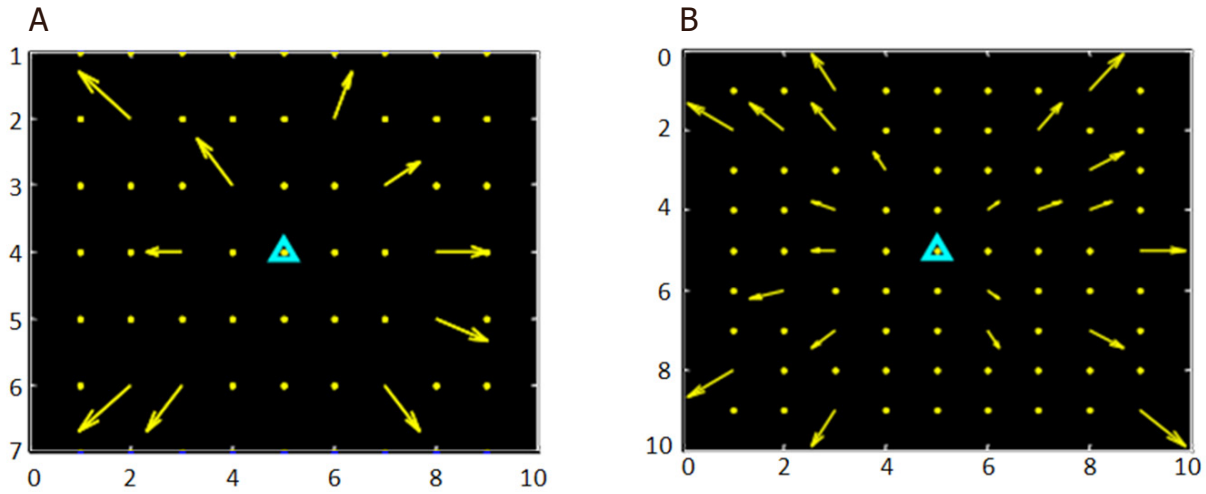


Figure 5. Escape center obtained in the simulation data experiment. A: center test result 1; B: center test result 2

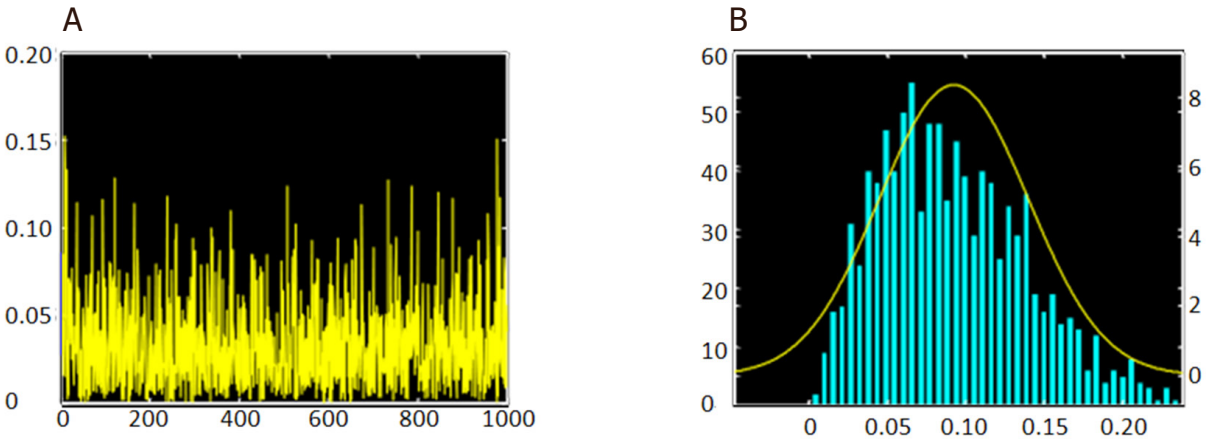


Figure 6. Curve image of deviation. A: variation of deviation; B: Histogram and Gaussian fitting curve of deviation

escape center of the crowd, and the escape center is defined as the position where the intersection of the motion vector is the densest. Therefore, the removal of wild intersections should remove a smaller number of intersections in the search window. The graphical method used in this paper is obtained through many experiments, and the empirical threshold is used to measure the intensity of intersections to assist the removal of wild intersections.

The limitations of individual motion in the population detected herein are often referred to as directional noise, and such noise is subject to a Gaussian distribution. Therefore, according to the actual situation of the research object in this paper, a set of random numbers with five-degree direction noise is designed to be added to the synthesized data, so that the synthesized data are a set of known ideal data of the escape center close to the actual situation. The test results are shown in Figure 5.

The accuracy of the method is further tested by the deviation of the statistical positioning results, as shown in Figure 5.

In Figure 6A, the abscissa indicates the number of experiments, and the ordinate indicates the mean square error. By calculating the average deviation of the 1000 experiments, the average deviation of the proposed

algorithm in the experimental data of adding direction noise is 0.0292. This shows that the algorithm has high positioning accuracy and strong anti-interference ability. It can be seen in Figure 6B that the deviation generally obeys a right-skewed distribution. Through the above simulation experiments, it can be seen that the abnormal positioning of a single escape center is highly accurate in the ideal synthetic data experiment.

The optical flow field of the video image is used to perform the acceleration feature extraction to determine whether an abnormality occurs. If there is an abnormality, there is escape from the center location. The position and direction information of the acceleration vector obtained when the abnormality is detected is used to obtain all the acceleration vector reverse extension lines Intersection, and then K nearest neighbor search is used to determine the possible escape from the center position.

The anomaly localization algorithm in this paper is based on the analysis of the motion vector of the moving target in the anomaly detection process to locate where the anomaly may occur. The process of locating is mainly by counting all the intersections of the inverse extension lines of the motion vector, and using the K nearest neighbor search method to determine the most likely abnormal location. The algorithm is as follows:

(1) Obtain the image sequence by framing the video, and then extract the motion feature vector.

$$A_X^n = \left\{ a_{x1}^{(n)}, a_{x1}^{(n)}, \ldots, a_{xk}^{(n)} \right\}$$
$$A_Y^n = \left\{ a_{y1}^{(n)}, a_{y1}^{(n)}, \ldots, a_{yk}^{(n)} \right\} \quad (4)$$

(2) Calculate the improved acceleration characteristics for each frame of image, and obtain the improved acceleration threshold through many experiments. If the improved acceleration value is greater than the threshold, the acceleration vector of this frame of image is retained.

(3) Calculate the corresponding univariate linear equation parameters according to the acceleration vector $A(a_x, a_y)$ of the obtained image; ignoring the special case, the parameter expression is as follows:

$$k_l = a_{y(\mathrm{i,j})} / a_{x(\mathrm{i,j})}$$
$$b_l = i - k_l \times j \quad (5)$$

where i and j represent the positions of corresponding pixels in the image.

(4) Calculate the intersection point set $P = \{p_1, p_2, \ldots p_s\}$ according to the straight-line equation of the acceleration vector. In order for the intersection point set to include all intersection points, the intersection points of all two different straight lines are mainly calculated. The mathematical derivation is as follows:

$$x = \left(b_2 - b_1\right) / \left(k_2 - k_1\right)$$
$$y = \left(b_2 - b_1\right) / \left(k_2 - k_1\right) \times k_l + b_1 \quad (6)$$

(5) In the process of calculating the intersection set, there are many repetitions or intersections that are inconsistent with the actual situation, which are collectively called wild intersections. Therefore, to improve the accuracy of the algorithm, wild intersections need to be removed from the intersection set.

(6) Determine the escape center by further analyzing the intersection set. Since the intersection set involved in this article belongs to a simple dataset, the K nearest point search method can be used. The K nearest point search method is also known as the K nearest neighbor search method. In the process of determining the escape center, this search method is to calculate the distance between intersections in the neighborhood, select the Kth smallest distance in the distance set of each intersection, and then compare them with other intersections to determine if they escaped the center. This search method is suitable for the classification of rare events. Since the escape center in this section is an infrequent event, and this search method is easy to implement without training, this paper uses the K nearest neighbor search method to determine the escape center of the abnormal crowd.

The graphical method of removing wild intersections is mainly based on the number of intersections in the search window. Intersections are rounded off, which is mainly determined by the particularity of the

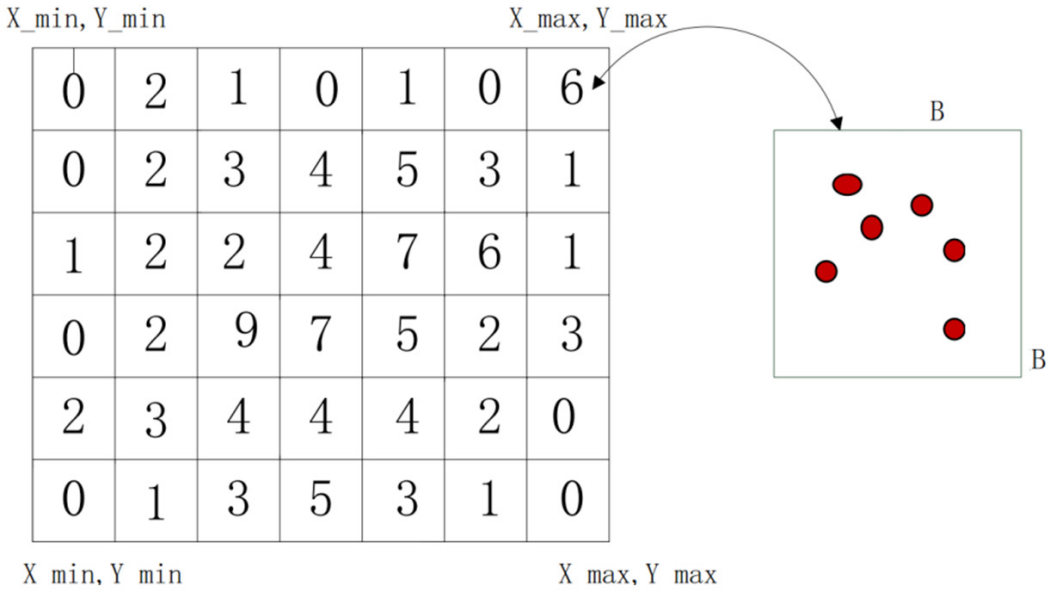


Figure 7. Schematic block diagram

research object of this algorithm. The algorithm is mainly used to locate the escape center of the crowd, and the escape center is defined as the position where the intersection of the motion vector is the densest. Therefore, the removal of wild intersections should remove a smaller number of intersections in the search window. The graphical method used in this paper uses many experiments to obtain empirical thresholds to measure the denseness of intersections to assist in removing wild intersections [Figure 7].

4 EXPERIMENTAL EVALUATION

Step 1: Select the video. In the experiments, 10 segments of video were selected to test three kinds of abnormal behaviors. The test video mainly came from a bank monitoring video, and some videos were from the crime scene monitoring. Moreover, the surveillance video was recorded by Haikang or Dahua cameras. Some test videos were taken by digital cameras, and these video formats were different, including AVl, MP4, DAV, *etc*. In addition, the system unified the video sources to AVI format for easy processing. The number of pedestrians appearing in the selected video, the time of appearance of the pedestrian, the video resolution, *etc*. were all random and there was no law. In addition, because shadow removal is not the focus of this paper, to reduce the impact of shadow on the algorithm, only video with less obvious shadow was selected when selecting video.

Step 2: Parameter configuration. First, the camera corresponding to the selected video was calibrated, and after the coordinate conversion was completed, the actual coordinate matrix was saved as an XML file (the file name is the camera name, and the file suffix is .xml). Then, the estimated target imaging size, frame interval, monitoring area, abnormal behavior type, etc. were saved in the PAR file (the file has the same name as the test video file and the file suffix is .par).

Step 3: The result of manual statistics. The video was observed by the human eye, and the manual statistical result was recorded in the MTR file (the file has the same name as the test video file, and the file suffix is .mtr). Manual statistics were needed to record the type of abnormal behavior, start time, and end time. Finally, the number of samples with abnormal behavior (abnormality) and no abnormal behavior (normal) in the selected video were counted according to the behavior type. The statistical results are shown in Tables 1-3.

Table 1. Number of manual statistical samples of regional invasive behavior

	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	Total
Abnormal	14	8	26	12	17	8	8	5	18	8	124
Normal	3	8	0	5	5	3	3	0	3	3	33

Table 2. Number of manual statistical samples of trailing behavior

	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	Total
Abnormal	6	8	14	11	8	3	2	3	8	5	68
Normal	3	5	5	6	2	3	0	3	0	3	30

Table 3. Number of manual statistical samples of defamation behavior

	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	Total
Abnormal	8	6	11	3	2	5	8	9	6	12	70
Normal	6	2	5	0	2	3	5	5	3	2	33

Figure 8. Identification of illegal intrusion crimes

In the detection of regional intrusion behavior, the algorithm does not use actual coordinates, but directly judges according to pixel coordinates. Therefore, there is no difference between the algorithm and the traditional algorithm in the abnormal behavior analysis stage. However, because the algorithm in this paper is better than the traditional algorithm in the target detection and target tracking stage, the false negative rate of the algorithm is greatly reduced, and finally higher accuracy is achieved. For the analysis of the video with false positives, it is easy to have false positives in the following cases: Pedestrians walk outside the edge of the surveillance area, and the feet do not enter the surveillance area but are closer to the boundary of the surveillance area, as shown in Figure 8.

One key to detecting trailing behavior is the setting of the relative distance and relative distance threshold between two pedestrians. The traditional algorithm is processed based on the pixel coordinate trajectory, thus the relative distance and the distance threshold can only be calculated by the number of pixel points, which is only an estimated value and is not accurate. However, the proposed algorithm is based on the actual coordinate trajectory, and the actual distance between the targets can be calculated, which is used as the distance threshold. In the test, the distance threshold of the algorithm was T_d = 1000 cm, and the distance threshold of the traditional algorithm was T_d = 100 pixels, as shown in Figure 9.

Awkward behavior is a relatively complex behavior that requires not only the calculation of the speed of pedestrian movement, but also the calculation of the angle of pedestrian movement. While the traditional algorithm is based on the pixel coordinate trajectory, the proposed algorithm is based on the actual coordinate trajectory [Figure 10].

Figure 9. Trailing criminal behavior identification

Figure 10. Identification of wandering criminal behavior

To further verify the performance of the algorithm, the UMN video library and PETS200 video library were used to perform simulation experiments on algorithm performance verification. First, the feasibility of the algorithm was verified at the theoretical level by using the set ideal dataset for simulation experiments.

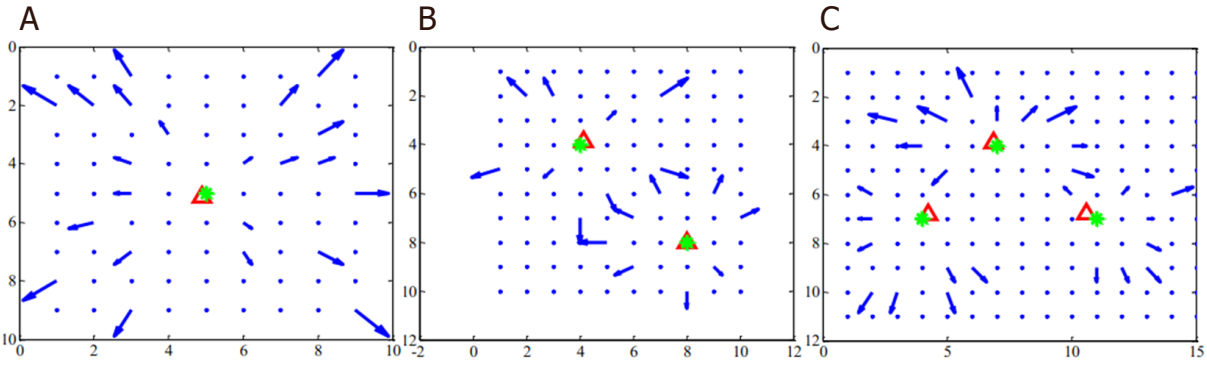


Figure 11. Results of escaping from the center position in the synthesized vector field. A: one escape center; B: two escape centers; C: three escape centers

As shown in Figure 11, the algorithm was used to perform different positioning experiments on different numbers of escape centers. Considering that it is difficult to have more than three sudden escape events in actual scenes, the research situation was a simple incident.

Since the data used in the simulation experiment were specifically designed, the red mark and green mark in Figure 11, respectively, represent the detected escape center and the ideally preset escape center. The detection of ideal data can verify that the algorithm is theoretically feasible. To make the designed data closer to an actual dataset, the same method as the previous simulation experiment to add Gaussian direction noise was chosen. It was found through experiments that the addition of directional noise did not affect the accuracy of the algorithm much. However, from the above simulation experiments, it can be seen that, for the three scenarios of escaping the center, the selection of K value by the KNN search method during the process of escaping the center and removing the wild intersections has a certain impact on the performance of the entire algorithm. These two factors that may affect the performance of the algorithm were analyzed further, as shown in Figures 12-14.

As shown in Figure 12, the overall positioning performance of the algorithm is still very high. Figures 13 and 14 show the change of the mean square error of the positioning error caused by the selection of different K values during the algorithm's search process for escaping the center and the process of removing wild points. It can be seen from the fitting curve in Figure 13 that the accuracy of selecting an appropriate K-value algorithm corresponding to different directional noise angles can be higher. Figure 14 is a comparison diagram between the searching wild intersection removal method and the graphical method. It can be seen in the figure that properly selecting the value of K under a fixed noise angle can make the performance of the algorithm in this section better.

In order to test the effect of this research algorithm in the actual crime detection, through the intelligent identification of 60 sets of crime surveillance videos, the identification effect of crime actions is counted. The crime recognition accuracy rate is shown in Table 4 and Figure 15.

As shown in Figure 15, this algorithm has a high recognition rate for criminal actions, and has certain practical significance. It can be applied to practice.

5 DISCUSSION

This study verified the accuracy of the abnormal positioning method of the single escape center through simulation experiments. First, this study verified the accuracy of positioning under synthetic data, which mainly proved the feasibility of the theory. Then, this study validated the positioning method using

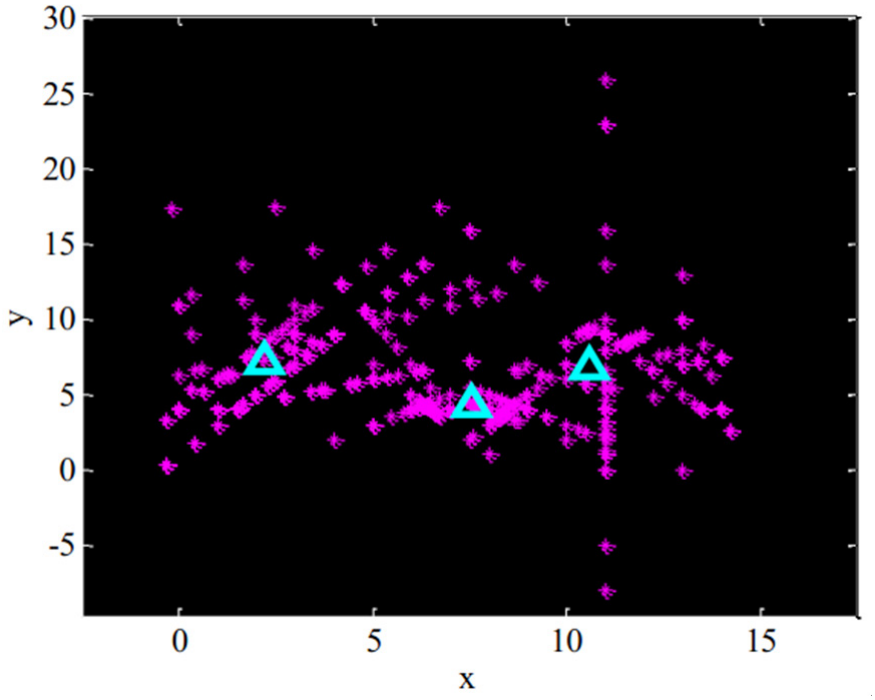


Figure 12. Location of the simulated escape center in the intersection diagram

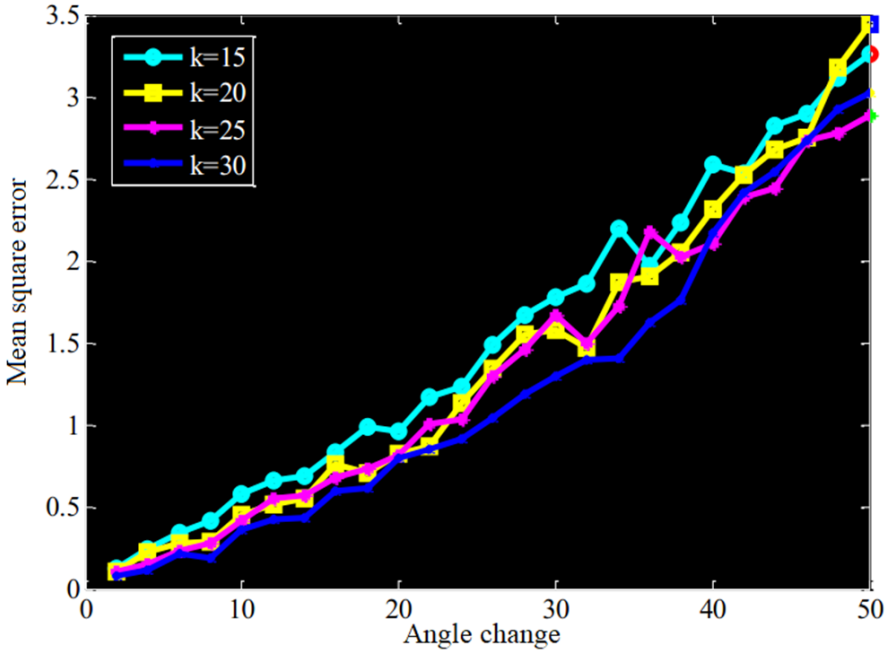


Figure 13. Influence of K value on detection results

the UMN dataset. This experiment was mainly to verify the accuracy of the algorithm in the actual environment.

The concept of "escape center" is used to approximate the abnormal scattered behavior of the crowd. According to the state of the crowd distribution index and acceleration characteristics on the playing field,

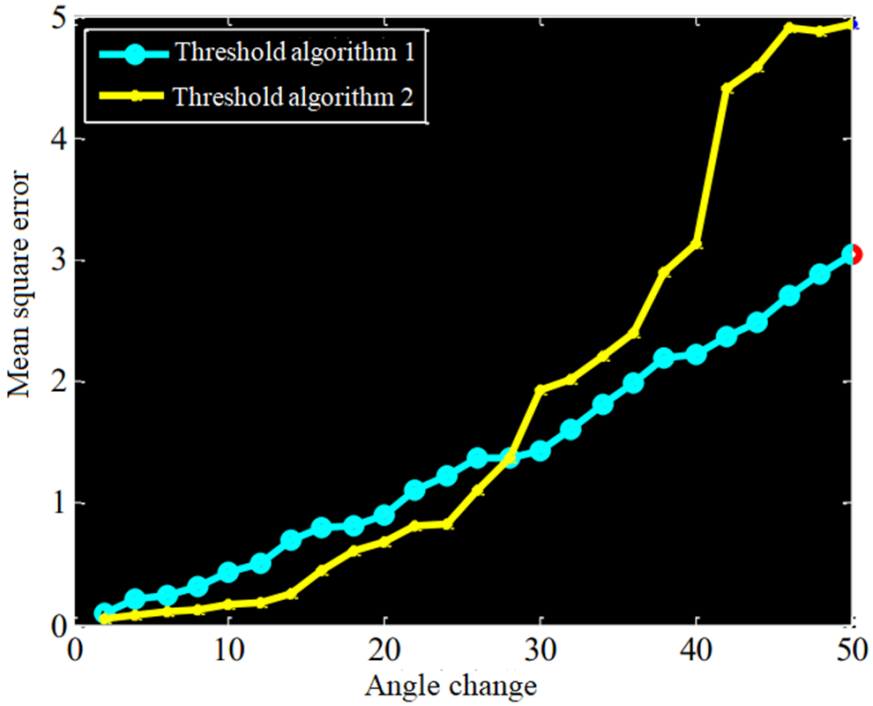


Figure 14. Comparison of errors of different field removal methods

Table 4. Statistical table of crime identification accuracy (%)

No.	Crime identification accuracy rate (%)	No.	Crime identification accuracy rate (%)	No.	Crime identification accuracy rate (%)
1	84	21	92	41	85
2	92	22	89	42	89
3	84	23	91	43	87
4	84	24	84	44	94
5	82	25	90	45	93
6	86	26	83	46	80
7	91	27	92	47	82
8	82	28	85	48	86
9	87	29	82	49	91
10	82	30	85	50	95
11	82	31	92	51	92
12	88	32	93	52	94
13	87	33	88	53	93
14	83	34	86	54	85
15	92	35	84	55	93
16	90	36	89	56	90
17	94	37	90	57	90
18	85	38	88	58	95
19	92	39	87	59	83
20	82	40	88	60	83

the occurrence of anomalies was detected, and the location algorithm of the escape center was studied. Firstly, the algorithm to detect the single escape center was implemented. To further detect the possible location of anomalies in the actual scene, based on the single escape center, further research was conducted to obtain multiple localization algorithms for the escape center. Many experiments were performed on the synthetic data and the UMN public dataset. The experiments show that the algorithm based on the multi-escape center is accurate in different scenarios.

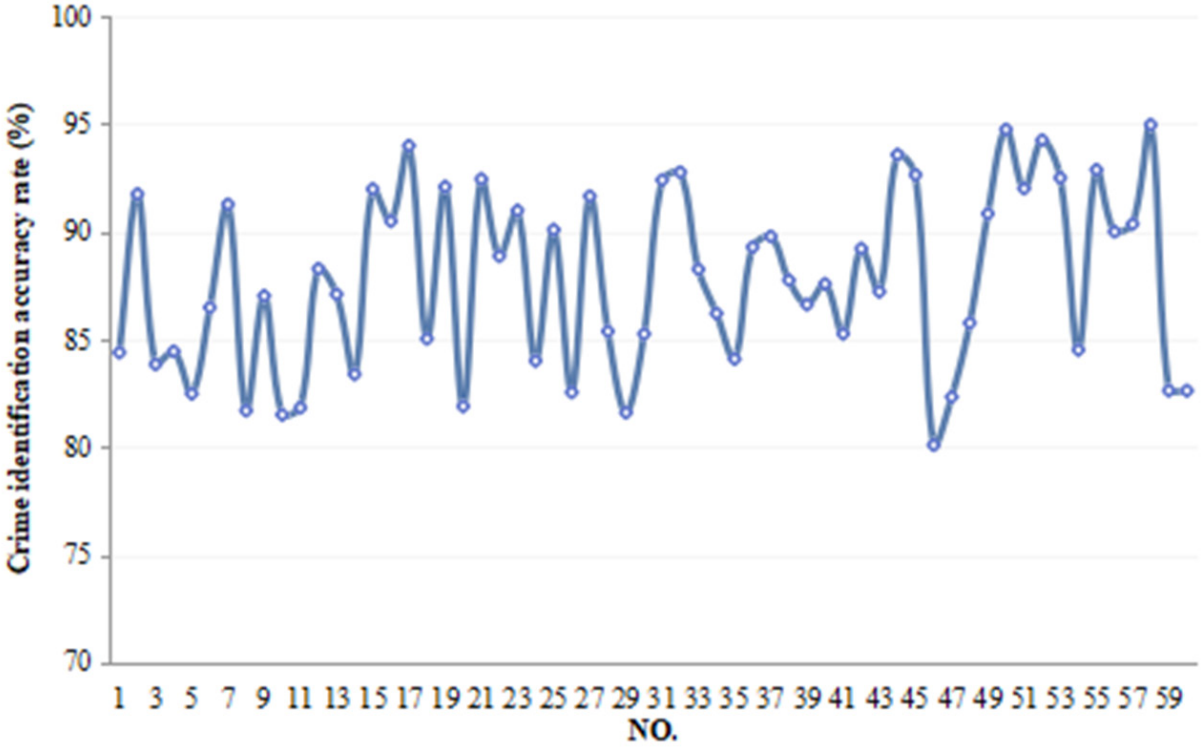


Figure 15. Statistics of crime recognition accuracy rate

In the actual public places, everyone will be affected by the movements of people around them and adjust their movement direction. The restriction of this group of people on individual movements must exist in a certain number of places. Therefore, the synthetic data used in the verification of the anomaly location method of the escape center should include the interference of the crowd limiting factor.

In view of the accuracy and real-time requirements of crime behavior identification, this study tested the accuracy and time consumption of the algorithm. Accuracy test calculated the system's false negative rate, false positive rate, and accuracy by comparing system analysis results with manual statistical results. The algorithm time consumption was evaluated by averaging the processing time per frame. The accuracy test process mainly included steps of selecting video, parameter configuration, manual statistical results, pixel-based traditional algorithm testing, algorithm testing, and comparative analysis.

As shown in Figure 7, when the algorithm detects the intrusion behavior of the area, it judges whether the left and right places of the pedestrian's external rectangular frame are in the monitoring area. If there is a certain location in the monitoring area, it is considered that there is a regional intrusion. Although the left and right positions of the rectangle are very close to the position of the two feet of the person, the position of the foot cannot be completely replaced correctly, which is also the reason for the false positive of the behavior. Although there are certain false positives in this method, the overall accuracy is high, and the method is simple and convenient. Therefore, it can meet the accuracy requirements of the security system.

It can be concluded from the results in Figure 8 that the accuracy of the proposed algorithm i is significantly improved compared with the traditional algorithm. The number of pixels corresponding to the actual distance of 1 m is affected by factors such as the angle at which the camera is mounted and the distance from the camera. Moreover, a uniform estimate is difficult to apply to any area of an image. When a pedestrian moves relative to the camera, the number of pixels separated by the two pedestrians' changes.

Therefore, when using traditional algorithms, it is easy to have false positives or false negatives. The algorithm uses the actual distance threshold and judges the distance between the actual trajectory points of the two pedestrians, which is not affected by any factors, thus the accuracy is higher.

It can be concluded from the results in Figure 10 that the accuracy of the proposed algorithm is significantly improved compared with the traditional algorithm. The traditional algorithm calculates the movement speed and movement angle of pedestrians based on the pixel coordinate trajectory. However, since the camera in the security scene is generally fixed on a vertical wall rather than directly above the surveillance scene, the angles of the pixels in the video tend to be different from the true angles. In addition, the threshold in the traditional algorithm is set by the number of pixels according to experience, which has no practical significance, thus the accuracy of the traditional algorithm is not high. In summary, the algorithm has high accuracy in identifying abnormal behaviors and has high practical value, which can meet the accuracy requirements of security systems.

Although the research presented in this paper has achieved good detection results in simulation experiments on actual datasets, there are still some problems in the whole algorithm that need to be improved. Firstly, the types of anomalies that can be identified in this article are not single, and they are not capable of identifying complex and diverse situations. Secondly, the algorithm is suitable only for the detection of abnormal behavior of low- and medium-density crowds, because when there are big crowds in the scene, severe occlusions will make the results of the number estimation algorithm inaccurate. Finally, the algorithm that detects the escape from the center does not detect more than three positions where anomalies may occur, and the algorithm is executed after detecting the occurrence of crowd abnormal events. How to optimize the calculation to achieve automatic and intelligent identification of the positions of crowd abnormalities that may occur is a key issue to be studied in the future.

Based on data mining, this study combined cloud computing image processing technology to identify real-time crime behavior. The abnormality detection object in this paper is a random group in ordinary public places, i.e., the people in the crowd are not unified and purposeful. The direction of movement of the crowd is irregular when there is no abnormality. At the same time, this study briefly illustrates that acceleration is an important motion feature of crowd anomaly detection. It also shows that the acceleration-based crowd anomaly detection algorithm is more reasonable than the traditional speed-based ones. Therefore, this paper uses the anomaly detection algorithm with improved acceleration characteristics to detect the abnormal escape behavior of the crowd. Firstly, the motion vector field is processed by block processing, and then the image is filtered to reduce the influence of noise. Next, the mean filtering is adopted, and then the algorithm is used to extract the foreground of the image sequence. This kind of operation not only facilitates the extraction of motion features, but also reduces the disadvantages of large computational complexity. The experimental research shows that the algorithm has high accuracy in identifying abnormal behavior and has high practical value, which can meet the accuracy and real-time requirements of the security system.

DECLARATIONS

Authors' contributions

Made substantial contributions to conception and design of the study and performed data analysis and interpretation: Xu Z

Performed data acquisition, as well as provided administrative, technical, and material support: Cheng C, Sugumaran V

Availability of data and materials

Not applicable.

Financial support and sponsorship

This work is supported by National Key R&D Program of China (No. 2018YFB1004605).

Conflicts of interest

All authors declared that there are no conflicts of interest.

Ethical approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

REFERENCES

1. Xie W. Research on big data processing model of multi objective genetic algorithm based on static bayesian game in cloud computing. J Comput Theor Nanosci 2016;13:9633-7.
2. Chen DY, Huang PC. Visual-based human crowds behavior analysis based on graphmodeling and matching. IEEE Sensors J 2013;13:2129-38.
3. Wu S, Wong HS, Yu Z. A Bayesian model for crowd escape behavior detection. IEEE T Circ Syst Vid 2014;24:85-98.
4. Biswas S, Babu RV. Anomaly detection in compressed H.264/AVC video. Multimed Tools Appl 2015;74:11099-115.
5. Cong Y, Yuan JS, Liu J. Abnormal event detection in crowded scenes using sparse representation. Pattern Recognit 2013;46:1851-64.
6. Li W, Mahadevan V, Vasconcelos N. Anomaly detection and localization in crowded scenes. IEEE T Pattern Anal 2013;36:18-32.
7. Song X, Wu M, Jermaine C. Conditional anomaly detection. IEEE Trans Knowl Data Eng 2007;19:631-45.
8. Kratz L, Nishino K. Tracking pedestrians using local spatio-temporal motion patterns in extremely crowded scenes. IEEE Trans Pattern Anal Mach Intell 2012;34:987-1002.
9. Liu W, Li J, Cho YB. A novel architecture for parallel multi-view HEVC decoder on mobile device. J Image Video Proc 2017;2017.
10. Benhajyoussef A, Ezzedine T, Bouallègue A. Gradient-based pre-processing for intra prediction in high efficiency video coding. J Image Video Proc 2017;2017.
11. Chu Y, Liu P. Some two-dimensional uncertain linguistic Heronian mean operators and their application in multiple-attribute decision making. Neural Comput Applic 2015;26:1461-80.
12. Mikaeil R, Haghshenas SS, Haghshenas SS, Ataei M. Performance prediction of circular saw machine using imperialist competitive algorithm and fuzzy clustering technique. Neural Comput Applic 2018;29:283-92.
13. Gupta BB, Badve OP. Taxonomy of DoS and DDoS attacks and desirable defense mechanism in a Cloud computing environment. Neural Comput Applic 2017;28:3655-82.
14. Chen L, Qu H, Zhao J, Chen B, Principe JC. Efficient and robust deep learning with Correntropy-induced loss function. Neural Comput Applic 2016;27:1019-31.
15. Tay KM, Jong CH, Lim CP. A clustering-based failure mode and effect analysis model and its application to the edible bird nest industry. Neural Comput Applic 2015;26:551-60.
16. Li G, Zhang L, Sun Y, Kong J. Towards the sEMG hand: internet of things sensors and haptic feedback application. Multimed Tools Appl 2019;78:29765-82.
17. Kowalczyk P, Sawicki D. Blink and wink detection as a control tool in multimodal interaction. Multimed Tools Appl 2019;78:13749-65.
18. Wang M, Chen W, Wang S, Liu J, Li X, et al. Answering why-not questions on semantic multimedia queries. Multimed Tools Appl 2018;77:3405-29.
19. Lei Y, Zhou X, Xie L. Emergency monitoring and disposal decision support system for sudden pollution accidents based on multimedia information system. Multimed Tools Appl 2019;78:11047-71.
20. Purificato E, Rinaldi AM. Multimedia and geographic data integration for cultural heritage information retrieval. Multimed Tools Appl 2018;77:27447-69.

Kim *et al*. *J Surveill Secur Saf* 2020;1:34-60
DOI: 10.20517/jsss.2020.14

Journal of Surveillance, Security and Safety

Original Article

Open Access

Check for updates

A survey of domain name system vulnerabilities and attacks

Tae Hyun Kim, Douglas Reeves

Department of Computer Science, North Carolina State University, Raleigh, NC 27695, USA.

Correspondence to: Prof. Douglas Reeves, Department of Computer Science, North Carolina State University, Raleigh, NC 27695, USA. E-mail: reeves@ncsu.edu

How to cite this article: Kim TH, Reeves D. A survey of domain name system vulnerabilities and attacks. *J Surveill Secur Saf* 2020;1:34-60. http://dx.doi.org/10.20517/jsss.2020.14

Received: 20 Apr 2020 **First Decision**: 9 Jun 2020 **Revised**: 13 Jul 2020 **Accepted**: 20 Jul 2020 **Available online**: 12 Sep 2020

Academic Editor: Fei Gao **Copy Editor**: Cai-Hong Wang **Production Editor**: Jing Yu

Abstract

Aim: The Domain Name System (DNS) plays an integral role in the functionality of the Internet. Clients receive Internet service by mapping domain names into internet protocol addresses, which are routable. DNS provides a scalable and flexible name resolution service to clients easily and quickly. However, DNS was initially developed without security, and the information is not secured. Although DNS security extensions was released in 1999 to protect against vulnerabilities, it is not widely deployed, and DNS continues to suffer from a variety of attacks. The purpose of this study is to provide a comprehensive survey of DNS security.

Methods: We describe an overview of DNS vulnerabilities, DNS attacks, and even mitigation systems. In detail, attacks are classified by purpose and methods for defending against these attacks are introduced and assessed. Finally, we conclude with a summary of the current state of DNS security.

Results: The main findings of this study is to introduce fundamental vulnerabilities of DNS and classify representative DNS attacks into four categories to efficiently analyze them. Moreover, we describe and assess mitigation systems to defense these attacks.

Conclusion: We conclude that DNS is an integral part of Internet operations but is still exposed to various attacks due to its vulnerabilities, low deployment of available mitigation techniques, and limitations of such techniques.

Keywords: Survey paper, Domain Name System, DNSSESC, network security, DNS attacks, DNS mitigation system

1. INTRODUCTION

Over the past 30 years, we have experienced more convenient Internet services through the human-friendly Domain Name System (DNS) functionality, which maps domain names to internet protocol (IP) addresses using globally distributed hierarchical name servers. Internet users with domain addresses can utilize various Internet services, such as web surfing, e-mail, and even mobile services without entering machine-recognized IP addresses. However, DNS was first developed without consideration of cybersecurity and caused many problems[1,2]. There is no doubt that there are many cyber attacks on DNS in the wild. In a recent attack, for instance, attackers redirected DNS lookup for MyEtherWallet.com to a malicious website that looked like an authentic website, for hijacking victims' account information[3].

To overcome such various DNS security problems (i.e., directory lookup) and reinforce cybersecurity, the DNS security extensions (DNSSEC) protocol was developed. DNSSEC implanted the digital signature mechanism of public-key cryptography into the DNS system[4-7]. DNSSEC extends DNS based on the hierarchical public key infrastructure (PKI) to protect data published in DNS. Certificates for the public keys are issued by trusted certificate authorities (CAs), which certify the ownership of the public keys. Thus, clients and resolvers can verify that DNS responses have not been forged or altered, using DNSSEC. However, DNSSEC still suffers from deployment issues in the current Internet. Chung *et al.*[8] found that 31% of domains supporting DNSSEC failed to publish all relevant records required for validation and 39% of domains used an insufficiently strong key-signing key. They also found that 82% of the resolvers requested DNSSEC records, but only 12% of them attempted to validate the DNSSEC records. Additionally, several studies have been performed to scrutinize the CA model for lack of transparency and choice of trusted CA sets[9,10]. If one of the CAs acting as a trust anchor is compromised, all information certified by the CA may be falsified.

The 2016 Dyn cyberattack was a significant event indicating serious DNS risk. Dyn, which is a popular DNS provider, was attacked by two large and complex distributed denial-of-service(DDoS) attacks against the DNS infrastructure[11]. Eventually, several major Internet services and banking systems were paralyzed. Figure 1[12] shows the map of the Internet disabling in North America by the Dyn cyberattack. An interesting issue with this attack is that a large part of the US was impacted by attacking Data Centers in only certain parts of the US. That is, the attack directly targeted only a locally distributed DNS with a local Botnet. Moreover, the Cyber Security Report[13], released in 2018, describes DNS as the largest (82%) Internet service target of application-layer attacks. Despite efforts to improve DNS's security problems, DNS is still a popular target for cyberattacks because of its essential role on the Internet, and its vulnerability.

This paper is a comprehensive survey of vulnerabilities of DNS (and DNSSEC), attacks exploiting those vulnerabilities, and mitigations proposed or deployed to address such attacks. There have been previous surveys on more restricted aspects of DNS security[14], a broader security context that includes DNS[15], or the use of DNS to combat specific types of attacks[16,17]. The contributions of this paper are: (1) first, the problems of DNS and DNSSEC security are described and classified as fundamental, structural, and systematic vulnerabilities. Also, the increasing seriousness of DNS attacks is discussed; second, various DNS attacks are discussed and classified by purpose, to understand and analyze them; finally, defenses against DNS attacks are described, and the effectiveness of current DNS attack mitigation is assessed.

The paper is organized as follows. Section 2 provides background on DNS and DNSSEC. Section 3 describes the security vulnerabilities of DNS and DNSSEC. Section 4 explains typical DNS attacks that currently threaten Internet users, assesses these attacks according to seriousness and classifies DNS attacks by purpose. Section 5 explores DNS attack mitigation methods and assesses their strengths and weaknesses. Section 6 concludes with the implications of this study and opportunities for research.

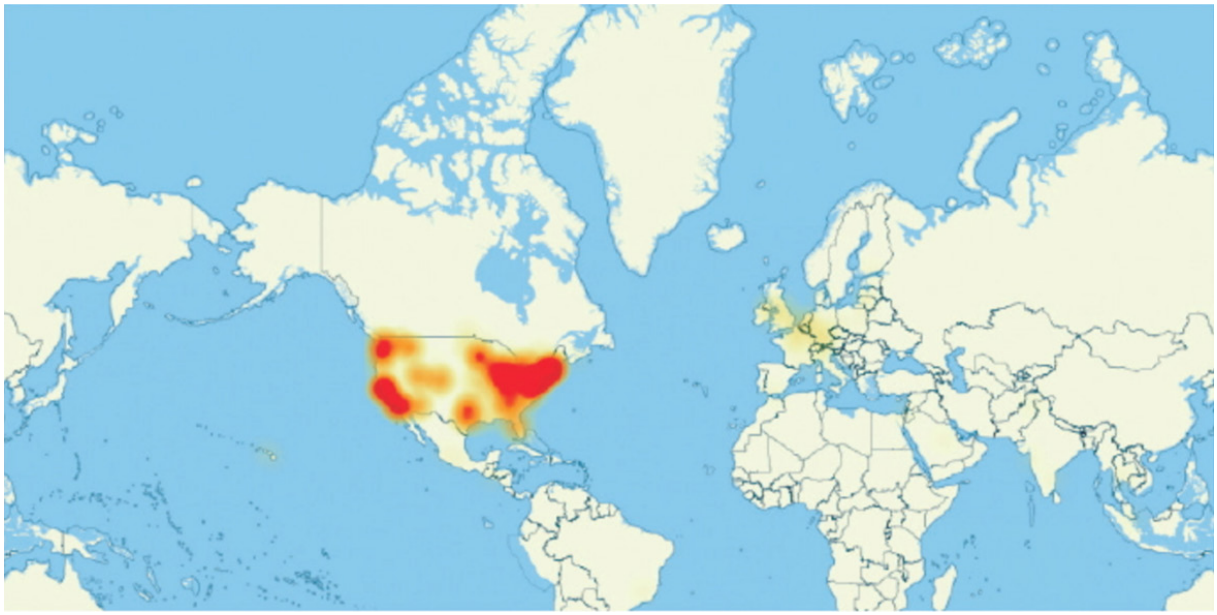


Figure 1. Map of Internet disabled in US by the Dyn Attack

2. BACKGROUND

2.1 DNS

DNS is an Internet system to map alphabetic domain names to numeric IP addresses[1,2,18]. In this paper, DNS is defined as the following:

Service: DNS is a name resolution service. The domain name can be matched to the IP address through DNS.

System: DNS is a distributed database system for the naming service as technical support. The DNS servers are located globally.

Server (Structure): DNS name servers are organized in a top-down tree structure to support an efficient naming service.

2.1.1 DNS history

In 1983, domain names were first translated to addresses through a local service, managed by the Operating System (OS). The host file in the OS stored these translations. Initially, only about 15 organizations used a single network, so keeping these files consistent and updated was straightforward, but not scalable. To address this inefficiency, the Stanford Research Institution Network Information Center (SRI-NIC) developed a new naming mechanism. The previous name service within the OS was transformed into a system that was managed and deployed collectively by SRI-NIC. The host file containing translation information (host name and numeric address) was hosted online by SRI-NIC and could be downloaded over FTP. However, as the Internet grew the difficulties of keeping the file updated, and the size of the file, became impractical. This resulted in poor search performance and traffic bottlenecks. To overcome these drawbacks, a new type of name system was introduced in 1987 as the IETF Request for Comments (RFC) 1034[2]. The DNS system was standardized and widely implemented and started to manage domain names on hierarchically-organized servers, growing into the current DNS system.

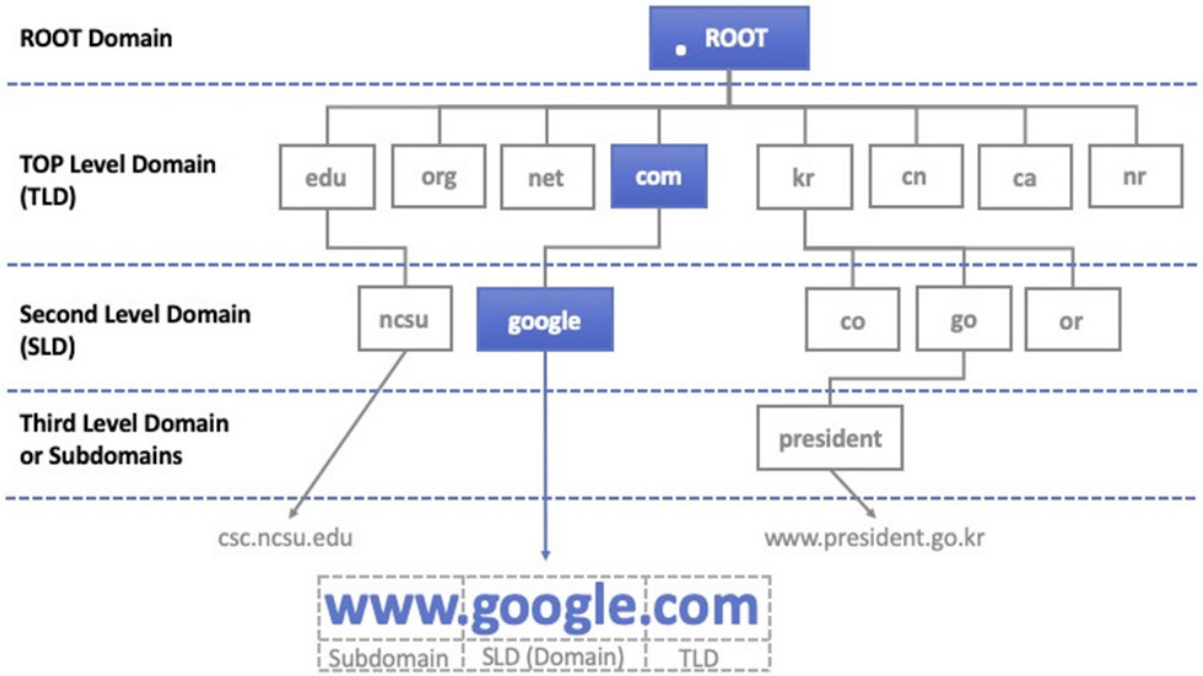


Figure 2. Domain Name System structure

2.1.2 DNS philosophy

Technically, DNS is a hierarchical name server system that uses a globally distributed database system that holds information about each domain. The DNS information is stored in distributed DNS servers, and the information can be searched at any time upon user request.

Figure 2 illustrates the hierarchical DNS structure via a common domain name. DNS begins with the .(Root) domain at the top. .com is a TLD (Top Level Domain) whose parent is the .(Root) domain. .google is an SLD (Second Level Domain) whose parent is the .com domain. Finally, .www (i.e., a web service) is a subdomain of .google.com.

As the top level of DNS, Root name servers are a global network with 13 redundant servers located in various countries, which manage all TLDs. The TLD comprises two types: the country code Top Level Domain (ccTLD) and the general Top Level Domain (gTLD). The ccTLD stands for the country domain name, such as .kr (South Korea) and the gTLD stands for the general domain type, such as .com (Company) or .org (Organization). As the number of domains increased, the number of available TLDs became insufficient, and ICANN announced a new set of TLDs in 2014. Currently, the number of TLD servers around the world is approximately 1,500 (maintained by IANA). Such vertical tree structure enables DNS not only to facilitate the management of each domain information but also to distribute numerous DNS requests efficiently.

The process of translating IP addresses to corresponding domain names through DNS is called name resolution or DNS resolution[1]. DNS resolution begins with a client's DNS request. Figure 3 illustrates how a client obtains the IP address for a web server via DNS resolution, allowing it to receive web services.

(1) A client requests an IP address www.google.com from a local recursive DNS resolver.
(2) The recursive DNS resolver first checks the address translation in its local cache.
(3) If there is no information in the cache, the recursive DNS resolver requests the IP address of the TLD

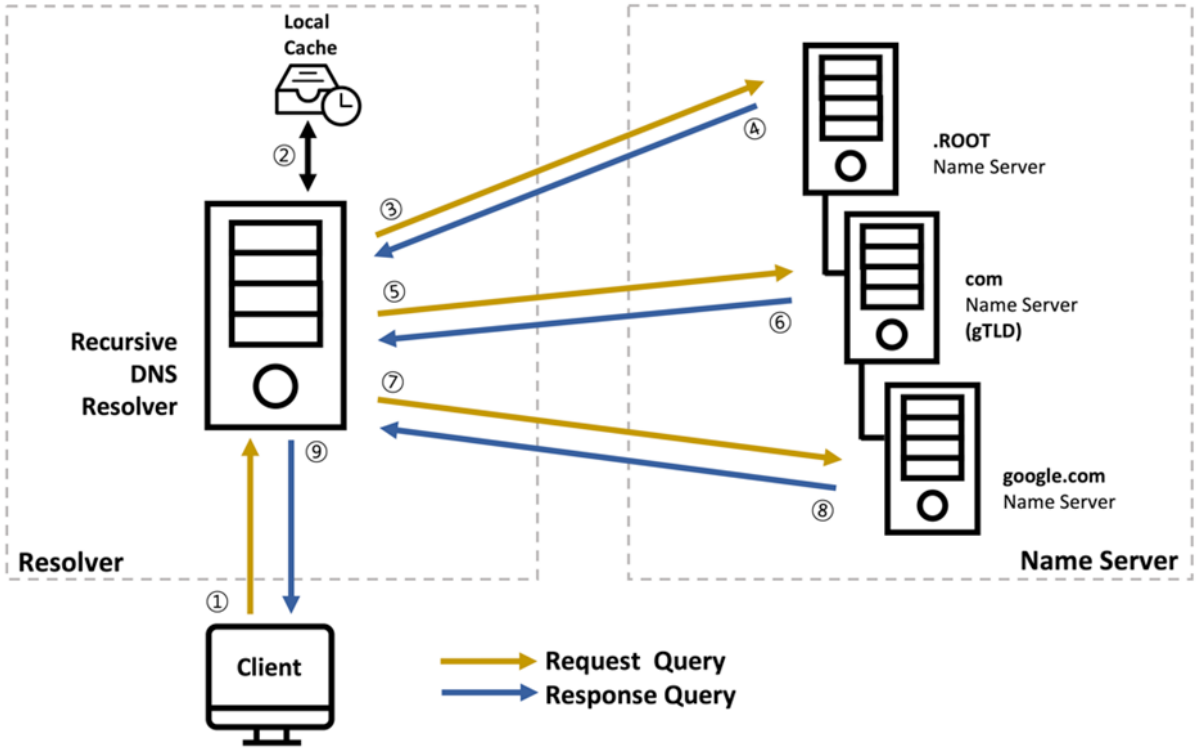


Figure 3. DNS architecture. DNS: Domain Name System; gTLD: general Top Level Domain

nameserver from the Root name server.
(4) The Root name server sends back the IP address of the .com name server as a response.
(5) Using this IP address, the recursive DNS Resolver requests the IP address of the SLD nameserver from the .com name server.
(6) The .com name server sends back the IP address of the .google.com name server as a response.
(7) With the IP address, the recursive DNS Resolver requests the IP address for www.google.com from the .google.com name server.
(8) The .google.com name server sends back the own IP address of www.google.com to the recursive DNS resolver.
(9) The recursive DNS resolver sends back the IP address of www.google.com to the client as a response. Finally, with the IP address (172.217.7.197 in this example), the client connects to the www.google.com server.

The DNS framework consists of the following three parts:
(1) Client: They request IP addresses with domain names through the stub resolver, a client of DNS, and transmits the request to the local DNS server address set on its device.
(2) Local DNS Server (Recursive DNS Resolver): They receive the DNS query from clients and obtains the IP address for the domain name from domain name servers. Also, the IP address once found is stored in memory for a certain period. So, it is called Caching Resolver.
(3) Domain Name Server (Authoritative Name Server): They have and manage IP addresses for the domain names as well as the information related to the IP addresses. The Authoritative Name Server is composed of more than 3-levels (Root, TLD, Lower-level Domain). Each domain server consists of a single master server and several slave servers.

In addition to the basic information regarding IP addresses for domain names, DNS databases provide additional information for a variety of services. DNS resource records (RR) have additional information

related to domain names as a DNS server database element, which is used to respond to DNS client queries. RRs are added to the DNS namespace generated by the DNS server and consist of various types, including the following:
(1) A and AAAA: A - IPv4 address or AAAA - IPv6 address.
(2) CNAME (Canonical Names): domain name aliases, used for mapping an alias to a domain name.
(3) NS (Name Server): indicates a specific authoritative name server or a name server address.
(4) Others: MX (Mail Exchange) - mapping the domain to an SMTP email server, PTR (Pointer) - Reversing IP address to Domain name resolution (reverse DNS lookup), and TXT - readable information.

2.1.3 DNS limitations
The major vulnerability in DNS is the lack of security. The original DNS protocol did not consider this issue in depth. Thus, DNS data could be forged to translate to a malicious IP address, so that Internet users would connect to a non-authorized site. This could, for example, be used to distribute false information or to surreptitiously collect personal information. DNS does not provide a way to verify that the received IP address translation is authentic. A corrupted or intercepted DNS response may provide false information to any requester. DNSSEC has been developed to overcome this fundamental security vulnerability of DNS[4,7].

2.2 DNSSEC

DNSSEC, which is an Internet standard technology, aims to eliminate this vulnerability of DNS. DNSSEC was originally standardized in 2005 as IETF RFCs 4033 through 4035[4-7]. Using two keys - the Zone Signing Key and Key Signing Key (KSK) - to create digital signatures with Public Key Cryptography, DNSSEC guarantees integrity and authentication for DNS data.

2.2.1 DNSSEC purpose
DNSSEC significantly enhances DNS security by adding Public Key Cryptography to the existing DNS. The DNS cache poisoning attack, for instance, configures an ISP's local DNS resolvers and their cache to map specific domain names to malicious IP addresses. As a solution to such DNS fundamental security problems, DNSSEC provides strong authentication using digital signatures, based on Public Key Cryptography[4,7].

2.2.2 DNSSEC philosophy
Figure 4 shows the basics of data authentication using public-key cryptography.
(1) Alice generates an asymmetric key pair, composed of a Public and a Private key.
(2) Alice distributes the Public key to the Internet.
(3) Alice creates "signature" by signing the plain text with her Private key.
(4) Alice transmits "signature" along with "original data" to Bob.
(5) Bob receives "original data" with "signature" from Alice
(6) Bob looks up the Public key of Alice
(7) Bob performs the signature validation of "original data" with "signature", using Alice's Public key.
(8) If the signature is successfully verified, then Bob is assured that the original data purportedly from Alice is correct.

DNSSEC applies the digital signature mechanism to resource records (RRs) to protect the data itself, which is set in each section of the response message. DNSSEC has added four new RR types to existing DNS records; these are Resource Record Signature (RRSIG), DNS Public Key (DNSKEY), NSEC/NSEC3, and DS. These record types support the digital signatures and the signature verification process[6,19].
(1) RRSIG: This RR has a signature for a DNSSEC-secured record set.
(2) DNSKEY: This RR contains the public key to verify the signature in RRSIG records.
(3) NSEC/NSEC3: This RR is for the explicit denial-of-existence of a DNS record.

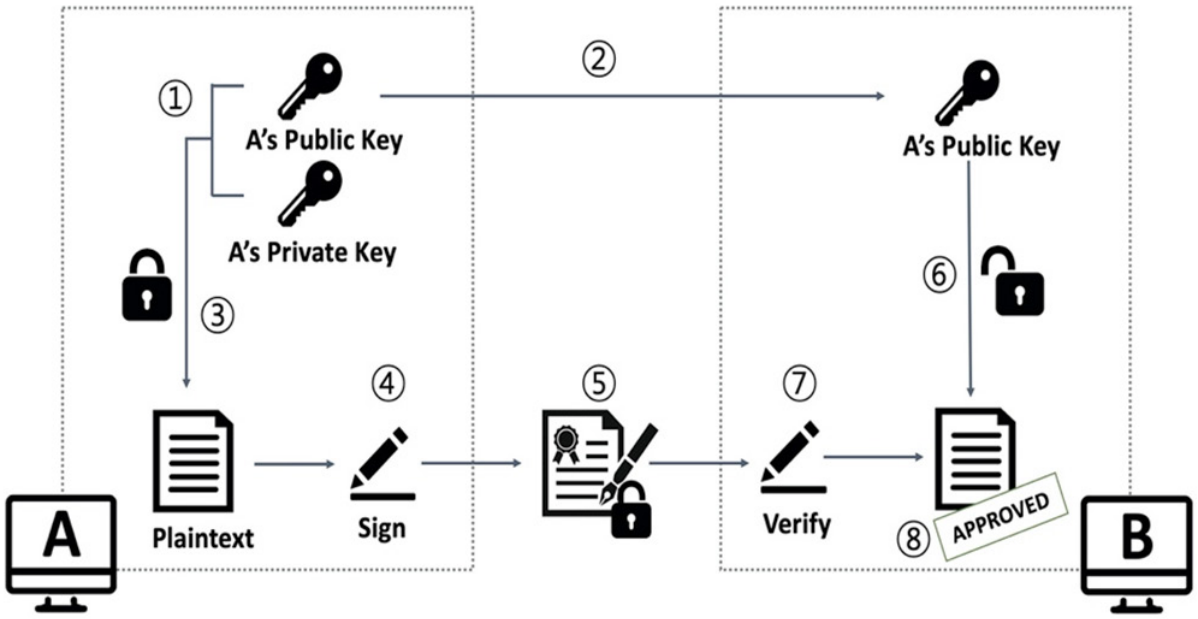


Figure 4. Public Key Cryptography Architecture

(4) DS (delegation signer): This RR holds the name of a delegated zone. The DS record is placed in the parent zone along with the delegating NS records for the authentication chain between the parent zone and child zone.

The DNSSEC protocol uses a Chain of Trust due to a strong, reliable connection between DNS servers. Figure 5 shows how DNSSEC works as the Chain of Trust. Compared with Figure 3, the IP address request of DNSSEC is the same as that of DNS. However, the verification process is added to the existing DNS. DNS servers verify each other with digital signatures from trusted CAs. Thus, DNS servers maintain a strong security chain between each other to guarantee the integrity and authentication of DNS data[7].
(1) A DNS resolver first sets a "Trust Anchor" that corresponds to the public key from a Root domain zone, as the KSK over DNSKEY record.
(2) The "Trust Anchor" is the starting point for verifying the signature in the signed DNS data, as the basis for ensuring "Trust" for Data Integrity.
(3) The DNS resolver performs signature verification from the Root domain zone to the A record data, which is the final node of verification, and then trusts the data.

DNSSEC adds strong security to authenticate DNS responses. Thus, DNSSEC assures users where the DNS data originated from, that is not forged in transit, and verifies whether a domain exists or not.

2.3 Multicast DNS

The multicast DNS (mDNS) protocol, described by RFC 6762[20], is a DNS service to resolve the hostname to IP address in small networks without a local name server. Unlike conventional unicast DNS, mDNS uses the IP multicast user datagram protocol (UDP) packet. Thus, every node on the network subscribing to that multicast address receives the request to resolve a hostname. The host owning that domain name responds, also using multicast, with its IP address. All nodes subscribing to the multicast address can update their DNS cache with the response. Figure 6 illustrates the basic mDNS protocol.

With the advent of IPv6 and the use of numerous embedded devices (e.g., IoT devices) greatly increasing, the normal, somewhat complex DNS infrastructure is inconvenient for local services configuration. To

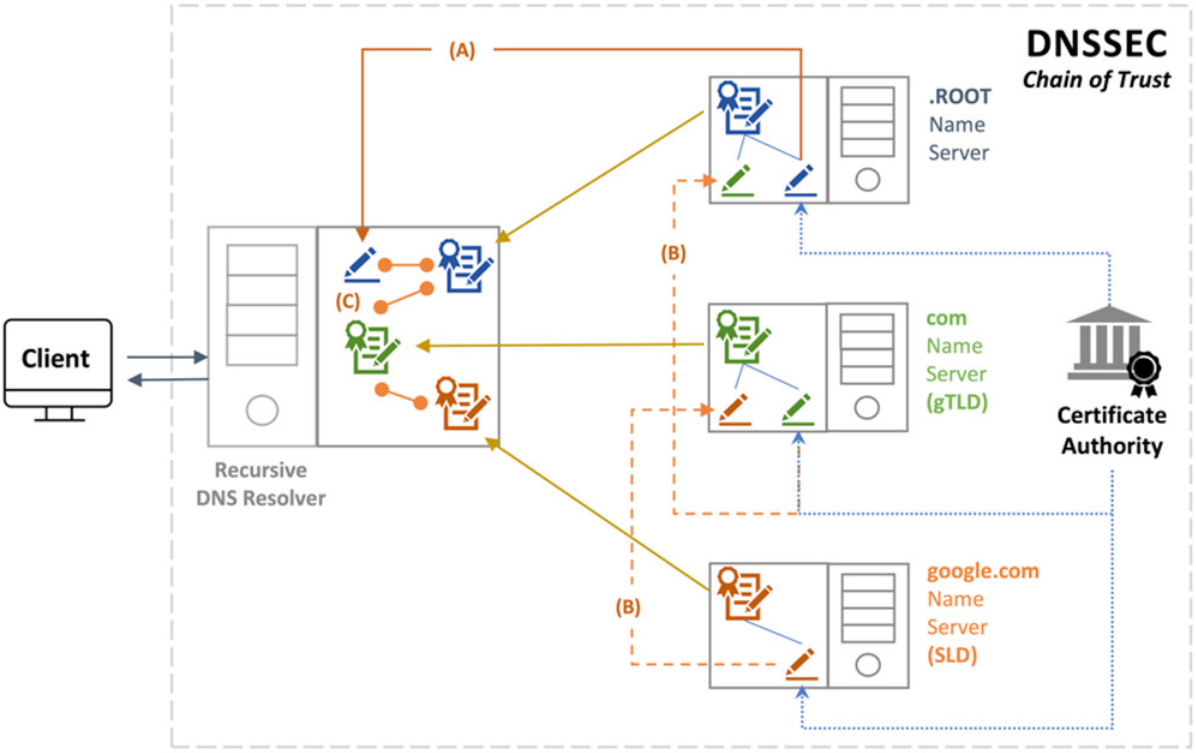


Figure 5. DNSSEC Architecture. DNS: Domain Name System; DNSSEC: DNS security extensions

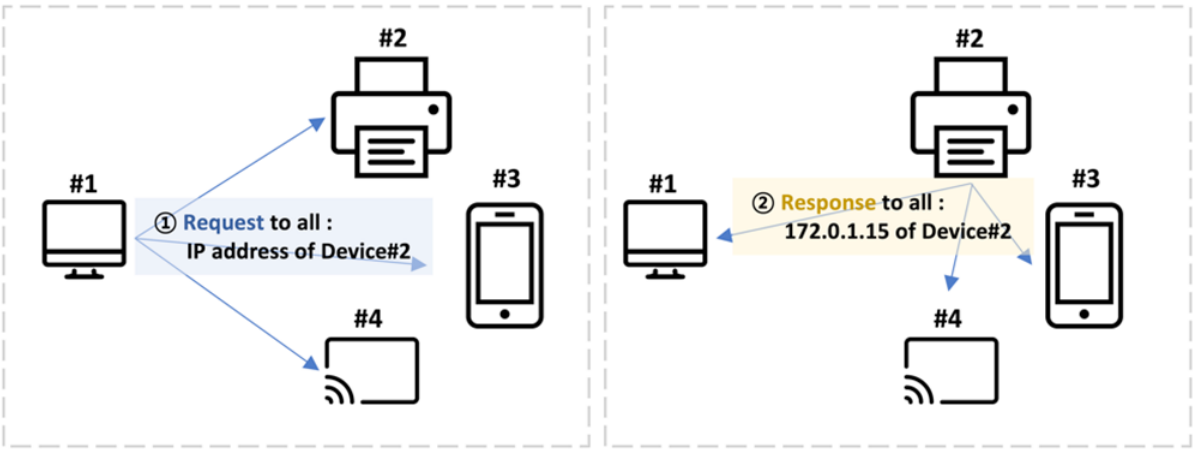


Figure 6. multicast DNS Architecture. DNS: Domain Name System

address this problem, mDNS was implemented by Apple Bonjour[20] and the Microsoft Link-local Multicast Name Resolution[21]. Initially, mDNS was intended to search for printer devices within a network but later expanded to the ability to resolve local hostnames.

The major benefits of mDNS are a zero-configuration and no infrastructure. It is available without conventional DNS settings and does not require a local name server. Also, users can connect and use devices in the network more conveniently because access to devices is intuitive.

mDNS has several weaknesses. First, if mDNS is exposed to the Internet, an attacker can easily collect information about devices and services on the network. Multicasting is inherently a powerful means

of mounting Denial of Service attacks. Since mDNS is a UDP-based protocol, it can be vulnerable to amplification attacks using mDNS queries, and spoofing attacks are trivial.

3. VULNERABILITIES

Cybersecurity is a defense mechanism to protect the system from various malicious attacks; cyberattacks disable or avoid these defenses. Vulnerabilities or weaknesses enable such attacks. This section looks specifically at DNS and DNSSEC vulnerabilities.

3.1 DNS vulnerabilities

DNS vulnerabilities can be viewed in 3 ways: by concept, by structure, and by communication.

3.1.1 Conceptual view

The CIA Triad is a conceptual model of information security, consisting of three factors: confidentiality, integrity, and availability[22]. The following is an assessment of DNS in terms of information security.

(1) Confidentially: DNS requests and responses are in most cases sent via the UDP protocol, which is light and fast, but normally unencrypted, allowing eavesdropping on all messages. Besides, the information stored by DNS servers is necessarily public, as name to address bindings must be served on demand.
(2) Integrity: DNS without modification does not have a mechanism sign data cryptographically, which is its single greatest weakness; anyone can tamper with or forge DNS data.
(3) Availability: the hierarchical structure of DNS, unless augmented with redundancy, is very much subject to attacks on DNS servers, or to failures of those servers.

3.1.2 Structural view

DNS servers have a hierarchical tree structure ranging from the Root to a specific domain name server. However, such a DNS feature includes structural problems, which can affect DNS vulnerabilities. The structural problems in DNS are as follows:
(1) Lack of redundant DNS[23]: The hierarchical DNS structure distributes and processes DNS queries more efficiently. Users can request an IP address of the desired domain step by step and obtain the response. Although DNS is designed to be distributed, traffics is concentrated because of the centralization. The centralized DNS structure makes it easier for an attacker to attack multiple Internet services used by many Internet users. For example, in 2016, a DYN attack exploiting such vulnerability made many users unable to receive normal DNS responses, as well as Internet services unavailable[11]. DNS above the SLD level, and major domain nameservers, have evolved over the years into a highly redundant system through numerous studies and cases. However, lower-level DNS servers remain exposed to threats due to a lack of redundancy. Resilient and reliable DNS support is possible if more domains adopt and support secondary DNS configurations[23].
(2) DNS server information exposure[24]: Because the fundamental security configuration of the DNS server is insufficient, the server information (e.g., server list, version) can be exposed through DNS servers of many companies. If such information is exploited, not only DNS operation but also server operation inside the companies can be exposed to the risk by attackers. The leakage of DNS server information allows malicious DNS data to be sent and the user to trust wrong DNS information. Additionally, attackers can collect information by reconnaissance attack and finally attack the server. Therefore, the security configuration of restricted server information transmission needs to be set up in each company's DNS servers.

3.1.3 Communication view

Responses to queries are only weakly protected in DNS. DNS uses the IP address, destination and source port numbers, and transaction ID in responses to match them with queries. It is relatively straightforward

for attackers to craft responses that pass these tests, as follows:
(1) No secured packet through UDP[25]: The basic query of DNS is delivered over the UDP protocol, which is unencrypted. An attacker could first capture a DNS query packet and forge a response from the name server in a malicious response before the resolver receives a valid response. This attack is made easier if routers are subverted as well.
(2) Transaction ID prediction[26]: The transaction ID is unique among several parameters that match DNS responses to requests. However, if the transaction ID is predictable, it makes it easier to forge a DNS response. The transaction ID is a 16-bit field in the DNS header and issued by the DNS algorithm. The ID value has a range of 32,768 values, but it is easier to predict if DNS randomization is poorly done (e.g., overload in cache). It is also predictable just by observing the request ID. Thus, attackers can easily guess the transaction ID and have their DNS response accepted as valid. For Berkeley Internet Name Domain (BIND) versions 4 and 8, a sequential transaction ID method is used, allowing the response ID simply to add 1 to the request ID. BIND version 9 and later adopts all randomized transaction ID and does not re-use the same ID for the same domain name. and predict the next transaction ID.
(3) Caching problems[27]: Caching is used for DNS efficiency. By storing the IP for the domain for a period of time, unnecessary IP address requests and access time to that domain can be reduced. Cache Poisoning, a typical DNS attack using such vulnerability, is one of the major threats to DNS. In cache poisoning, an attacker injects a malicious IP address into the DNS cache, causing users to receive false translation information for an extended period.
(4) Lack of protection against DDoS: About 93% of all cyberattacks on the Internet are reported as DDoS attacks[13]. DNS is also vulnerable to this attack. If DNS request floods occur, the DNS name server that handles the requests cannot respond to all requests making DNS service unavailable. As a consequence, all users using the DNS name server are unable to use the Internet. Due to the absence of a mechanism to block and prevent such attack patterns, DNS is currently suffering from many DDoS attacks.

3.2 DNSSEC vulnerabilities

As shown in Section II, DNSSEC has enhanced security for authentication and integrity by adding digital signatures using public and private keys to existing DNS to overcome known DNS vulnerabilities. However, DNSSEC is still suffering from various attacks through vulnerabilities and limitations.

3.2.1 Overhead

DNSSEC adds four record types to the DNS: RRSIG, DNSKEY, Delegation Signer (DS), and Next Secure (NSEC). Because of these extended records, DNSSEC requires more overhead than traditional DNS and increases processing time and packet size. The size of the DSSEC packet is up to 2000 bytes, while the UDP size specified by the RFC is 512 bytes. Therefore, the packets in DSSEC are fragmented, which may result in DNS fallback. For example, if the fragmented DNSSEC packets are not delivered properly and a public key that was previously verified during a key rollover is still stored in the local cache and a DNS data packet signed with a new key is received, verification of the new packet will eventually fail and be ignored. As a result, the user is provided neither with the DNS service nor authentication[28].

3.3.2 Complexity

The implementation of DNSSEC has been found to have problems in deployment. Misconfiguration may be increased because DNSSEC significantly increases the complexity of the existing DNS infrastructure[29]. The misconfiguration may result in incorrect DNSSEC RRs and authentication problems such that the data is regarded as fake, even though it is correct, causing name translation to fail[30].

3.2.3 Untrustworthy resolver

Assuming a reliable DNSSEC system is built on DNS, most of the DNS responses are trustworthy. However, if there are unreliable resolvers to deliver the final DNS response provided by the secure DNS

server, Internet users are exposed to DNS threats despite the robust DNSSEC[31]. Usually, most people do not consider how much they trust the local DNS resolver that is set up for them but simply use the default local DNS resolver provided by the network. For example, if a typical user connects to the Internet over public Wi-Fi, the DNS resolver is automatically configured as the default. Exploiting such a problem, an attacker may intercept the request and configure a malicious DNS resolver that delivers false DNS data to the victim. To counteract this, the chain of trust should be extended from the DNS resolver to the users. Dynamic Host Configuration Protocol (DHCP) with authorization tickets is one way to identify DNS resolvers that are trustworthy[32]. However, if the DHCP server is disabled, or untrustworthy itself, all users in the network could be affected.

3.2.4 Zone list exposure

The DNS database is broken into zones of records. Each zone contains not only a domain's records but may also contain its subdomains and related records. DNSSEC has a security function that can digitally prove a domain or resource record that does not exist, using the NSEC (Next Secure) record type. This, however, makes it possible for an outsider to find the names in an entire zone, a process known as zone enumeration. To address this issue, the standardization of the NSEC3 RR has been completed, but can still be seriously impacted by malicious NSEC3 and DNS servers that do not implement the standard[33].

Also, zone transfer is used to synchronize zone files between primary and secondary DNS servers. To synchronize zone files between DNS servers, it is often accomplished using NFS, or a specialized zone-transfer function. Although zone file transfers are necessary, misconfiguration of the transfer may pose a serious threat of leaking information.

3.2.5 Low deployment of DNSSEC

DNSSEC provides much stronger security for DNS, but it is currently plagued by the slow deployment of DNSSEC. According to an Internet Society Report in 2016[34], TLDs zones signed with DNSSEC were about 90%, while SLDs were only 65% of DNSSEC-enabled zones. In addition, considering that the usage of DNSSEC-validating resolvers is approximately 26%, the percentage of deployment might be lower. The report also points out that DANE's deployment, which enhances the DNSSEC's vulnerability, is also low.

3.2.6 Amplification and reflection DDoS threat

DNSSEC is still a possible vehicle for amplification and reflection attacks[35]. Due to the additional information caused by complex digital signatures, DNSSEC's record is significantly larger than a normal DNS response. On average, the size of an "ANY" response from DNSSEC is 28 times larger than a normal DNS "ANY" response[36], making amplification and reflection attacks even more damaging.

4 ATTACKS

This section presents the state-of-the-art for DNS attacks, classifies, and assesses them. Generally, the DNS attack is an attack that targets multiple DNS servers on the Internet, using the DNS and DNSSEC vulnerabilities described in the previous section. The goal of the DNS attack is to deplete the targeted system resource or to corrupt the data, make the DNS system unavailable, or exploit the system to achieve the final attack. As of now, the attacks are received considerable attention from researchers, governments and also industry, but they still cause a significant risk for Internet users.

DNS attacks may be separated into four categories: DNS data tampering, DNS data flooding, abuse of DNS, and DNS server structure. Figure 7 shows the list of 11 DNS attacks that are categorized.

4.1 DNS data tampering

DNS Data Tampering occurs when an attacker hijacks and/or compromises unencrypted DNS data in the middle between users and DNS servers, and then users receive false address translation information. The

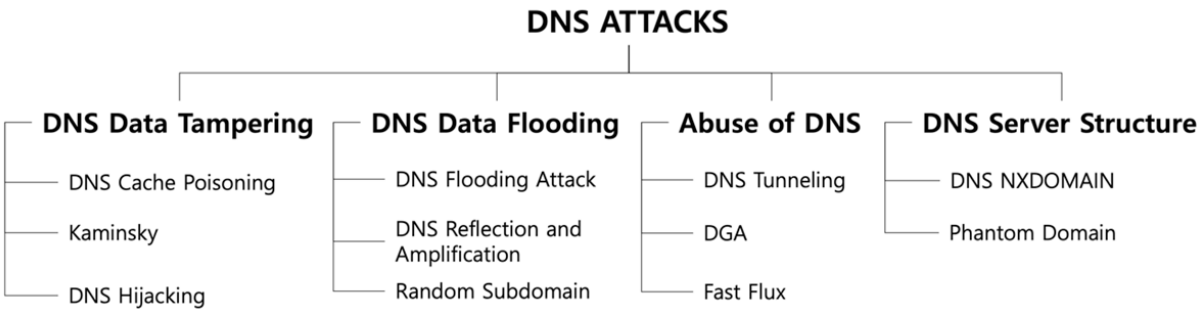


Figure 7. 11 DNS attacks. DNS: Domain Name System; DGA: domain generation algorithm

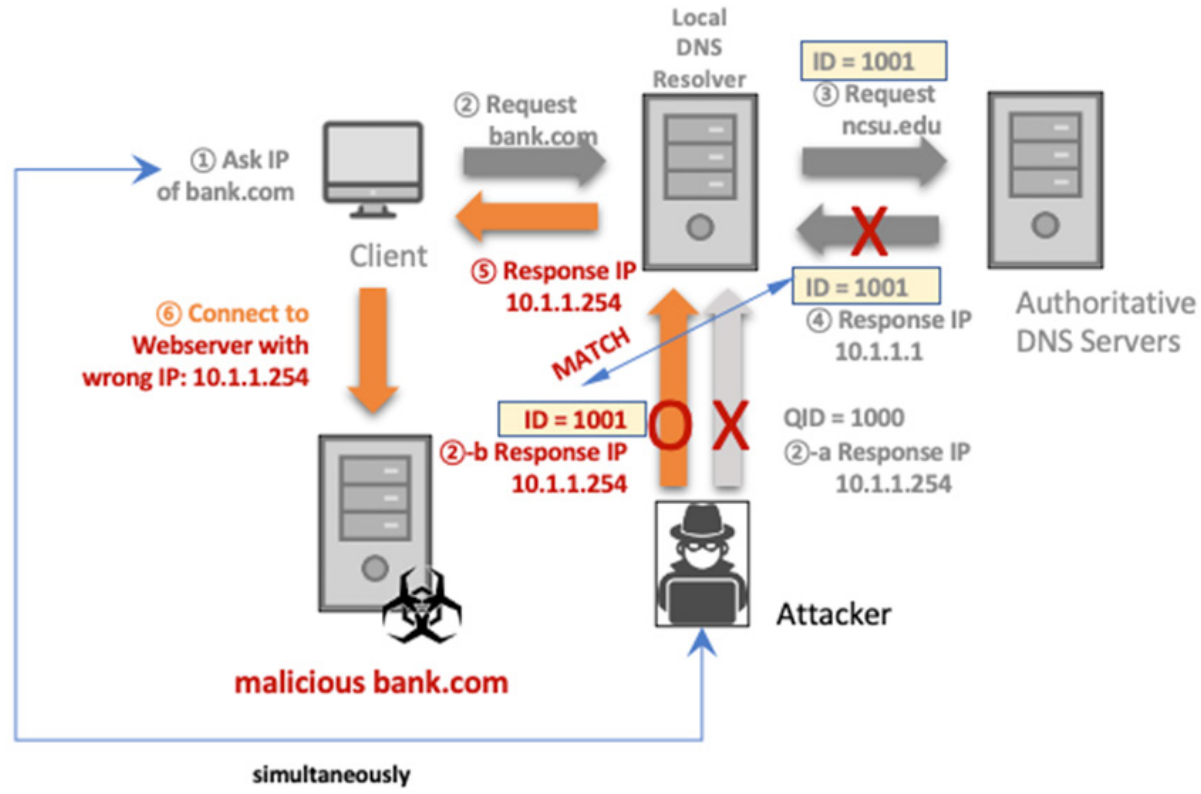


Figure 8. DNS attack: DNS data tampering. DNS: Domain Name System; QID: Query ID; "-a", "-b": the process order

attack is based on the vulnerability of insecure DNS data. Figure 8 shows how a typical DNS data tampering attack occurs. DNS attacks using data tampering are listed below.

4.1.1 DA01. DNS cache poisoning

DNS cache poisoning attack corrupts the data in the DNS cache. An attacker first queries a recursive DNS server for a domain. If the recursive DNS server (A) does not have an IP address corresponding to the requested domain in its cache, A sends queries to the authoritative name server (B). Before B can send an NXDOMAIN response, the attacker sends a large number of spoofed responses to A that appear to come from B. If the DNS response matches the DNS query, A will accept a spoofed response from the attacker and keeps the resource records (RRs) provided in that response in its cache. At a later time, a user asking for the translation of this same domain name contacts the A, which will provide the cached malicious IP address to the user[27].

Alharbi *et al*.[37] did a study on the risk of client-side DNS cache poisoning attack and discovered that a new type of DNS poisoning attack using vulnerabilities to caching within the end-user's operating system is feasible. Such vulnerability is still exposed because the client side is not considered as part of the DNS framework and, therefore, not considered in mitigations to the DNS cache poisoning attack.

4.1.2 DA02. Kaminsky

To protect against conventional cache poisoning attacks, DNS resolvers use a technique known as "bailiwick checking". To protect against malicious DNS additional records, the DNS resolver accepts only basic information and ignores additional information. To overcome this, attackers exploited the authoritative name server to poison resolver caches. Dating from Steven Bellovin's study in 1990, DNS hijacking and poisoning attacks developed into attacks based on the "birthday paradox", and eventually evolved into Kaminsky attacks in 2008[14,38].

Kaminsky attack hijacks the authoritative records instead of RRs. To succeed in the attack, the attacker should configure a domain name server that is authoritative for the malicious website zone, including all records, as a precondition. Kaminsky attack consists of two steps: Step 1: The attacker requests fake DNS queries about a random name within the target domain to local DNS servers. Since the local DNS server does not have the information in its cache, it will generate subsequent queries to authoritative name servers. Step 2: The attacker sends a barrage of forged answers to the local DNS server. Instead of fake RRs, it delegates to another name server, using the malicious authority record.

Finally, an attacker owns an authoritative name server for the specific website and provide users with malicious IP addresses for normal DNS requests of the domain through the DNS resolver. This attack is a higher level of attack than DNS Cache Poisoning Attack because it can affect not only the domain but also the subdomain.

4.1.3 DA03. DNS hijacking

DNS hijacking modifies DNS record settings (most often at the domain registrar) to point to a bogus DNS server or domain. Attackers hack the vulnerable DNS servers to change the IP address and the mapped domain address[39]. Cisco Talos discovered a new DNS hijacking attack called "DNSpionage"[40]. The main feature of this attack is to keep it as inconspicuous as possible during the attack. DNSpionage uses malicious Microsoft Office files with embedded malware, which provides HTTP and DNS communication with the attackers. Finally, malicious DNS redirection works when a user opens a forged document or malicious site. The main feature of this attack is to be as inconspicuous as possible during the attack.

4.2 DNS data flooding

In general, the goal of flooding attacks is to disable the user-server function by overwhelming the server, thereby hampering the DNS name resolution for its zone. Through the DNS data flooding attack, the attacker tries to exhaust server resources with an enormous amount of apparently valid queries, overwhelming server resources, and impeding the server's ability to respond to legitimate requests. Figure 9 describes the specific method of DNS data flooding.

4.2.1 DA04. DNS flooding attack

DNS flooding attack attempts to exhaust server-side resources through a flood of UDP requests from multiple machines contaminated by malware. DNS servers, which rely on UDP protocol for name resolution, may not be able to distinguish large UDP packets from normal requests. Attackers send a large volume of packets, mimicking legitimate DNS requests to a DNS server, causing the DNS server to run out of resources to handle legitimate requests[41].

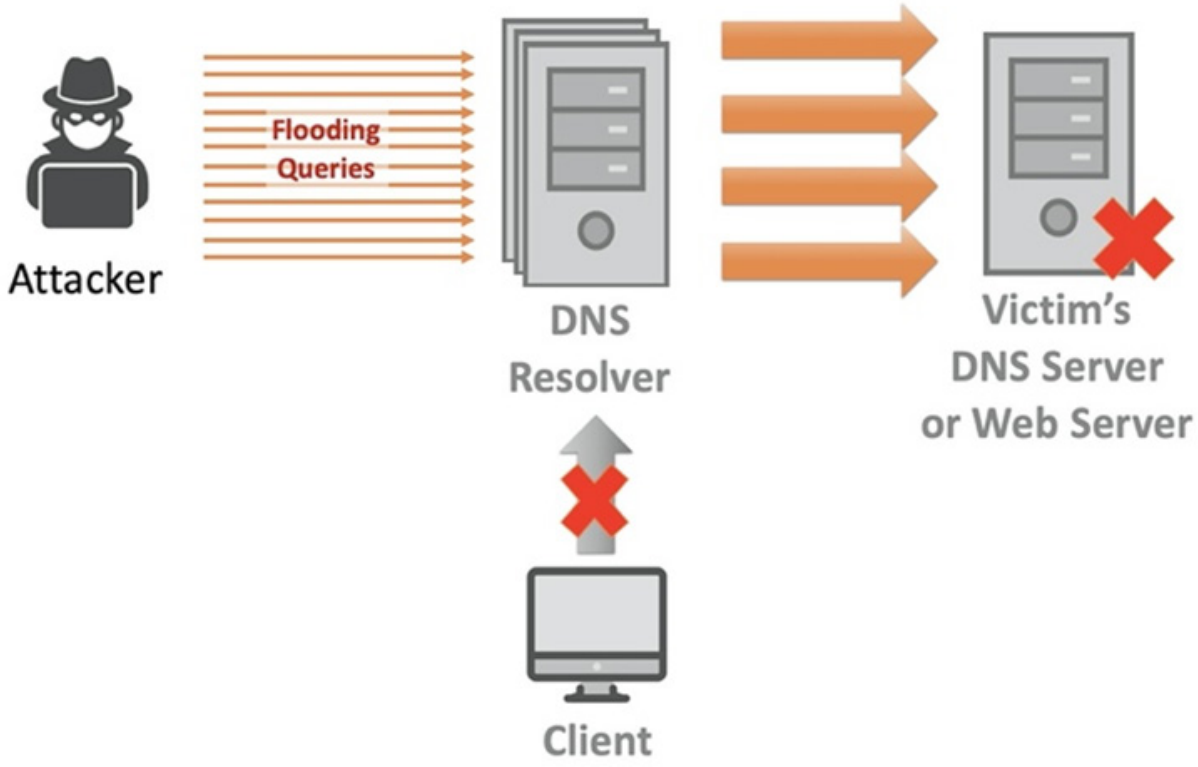


Figure 9. DNS attack: DNS Data Flooding. DNS: Domain Name System

4.2.2 DA05. DNS reflection and amplification DDoS attack

The obvious difference between DNS reflection/amplification DoS attack and DNS flooding attack is in the target of attacks[42]. While DNS flooding attack depletes DNS server's ability, DNS reflections and amplification attack attempts to saturate network capacity with heavy bandwidth traffic. This attack takes advantage of the vulnerability of third-party open resolvers in the network that combines reflection and amplification. An attacker sends out small request queries to multiple open recursive DNS servers, with a spoofed source IP address. The request is crafted to cause a large response packet. Through simultaneous reflection and amplification attack, the open recursive DNS servers generate a number of legitimate DNS responses, and finally, the victim server is attacked by DDoS. To mitigate such a DNS amplification attack, several security guidelines[43] have been issued, but still, amplification attacks have been widespread in recent years.

4.2.3 DA06. Random Subdomain

The random sub-domain attack is another type of DNS data flooding attack, sending a flood of randomized DNS requests for non-existent domains[44]. To succeed in the random subdomain attack, an attacker first infects numerous clients. Infected clients create request queries by adding randomly generated subdomain strings to the victim's target domain. Each client sends these numerous queries to a DNS recursive server, which attempts to resolve them with another server. Because this server continuously responds that the domain is nonexistent, the requests for random lookups eventually exhaust the limited resources, which delays or stops responses of legitimate lookups and all domains under the DNS server control. These attacks are used for DDoS attacks against domain name servers.

4.3 Abuse of DNS

The latest cyber attacks are active in botnets using Command Control (C&C) servers. A C&C server is a server that controls communication between attackers and zombie PCs (called Botnets) to attack a target.

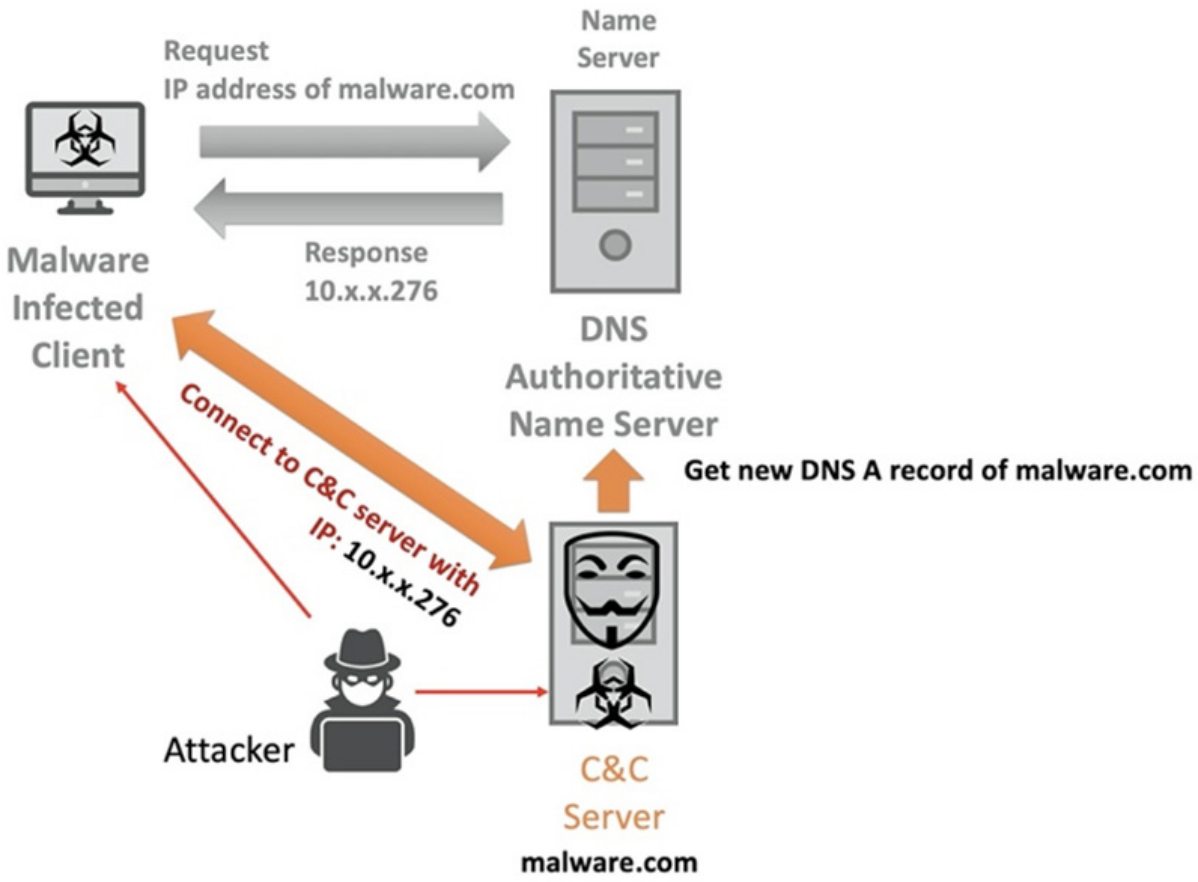


Figure 10. Demonstrating the DNS attack using abuse of DNS. DNS: Domain Name System

An attacker uses a C&C server to make it difficult to find the source of an attack and to scale to large numbers of bots. To counteract the development of methods for detecting C&C servers, an attacker exploits DNS to hide the location of C&C servers or to exfiltrate traffic to conceal the attack. To bypass firewalls, an attacker attempts to send malicious commands from inside a network to an external C&C server. In such a case, an attacker could conceal the information of the C&C server by using seemingly innocuous DNS (DNS TTL, NXDOMAIN) records, as shown in Figure 10.

4.3.1 DA07. DNS tunneling

DNS Tunneling is a type of bypass technology that allows an attacker to send attack commands and receive the results without blocking by the defense system. DNS requests may use up to 255 characters for a domain name, and subdomains separated by "." can be up to 63 characters. For example, if an attacker sends a DNS query of "ghAAAAATTTAAAACCCKKakdg.malware.com", the malware.com name server, as the C&C server, accepts the query as a malicious attack command. Conversely, the malware.com name server exploits records (A, CNAME, TXT) of the DNS response query to include the results for that attack command. Since an attacker and a C&C server communicate through DNS port 53, DNS tunneling may evade a defensive system[45,46].

4.3.2 DA08. domain generation algorithm

Domain generation algorithm (DGA) is an algorithm that randomly generates a large number of domains (from hundreds to tens of thousands)[47]. An attacker uses DGA to support malware attacks. First, an attacker attempts an attack by sending malicious commands to many botnets infected with malware

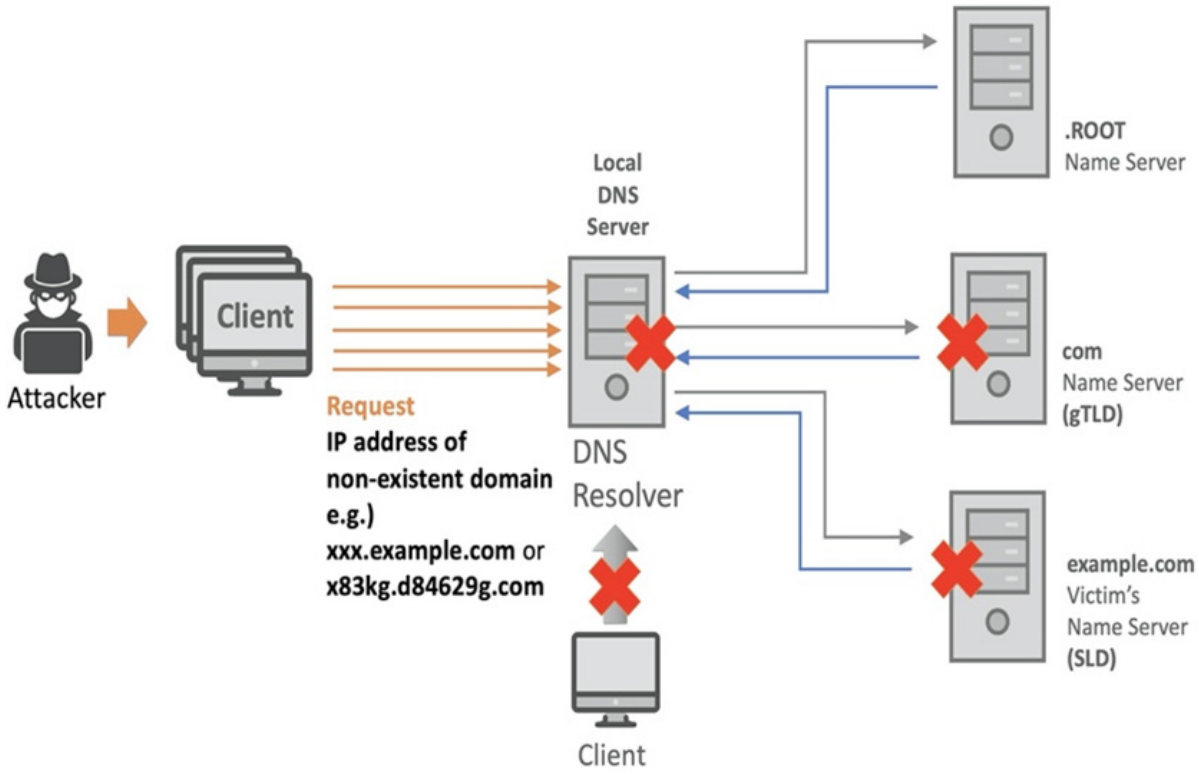


Figure 11. DNS Attack: DNS Server Structure. DNS: Domain Name System

through a C&C server. However, security devices or agencies may block the IP address of the C&C server to prevent communication. Some malware (such as the Necurs Botnet[48]) applies numerous domain names generated by DGA to continuously change the domain of the C&C server. This evades a domain reputation defense to hide the location of the C&C server.

4.3.3 DA09. fast flux

Fast Flux is a method of allocating multiple IP addresses to one domain. By setting the DNS response TTL (Time to Live) to a minimum value (typically within five minutes) and changing the DNS record on the DNS server periodically, the corresponding IP address of the C&C server may be changed repeatedly in a short time interval. This usually relies on a DNS server controlled by the attacker. If a security manager confirms access to a malicious domain and blocks the IP address of that C&C server on the firewall, Fast Flux attempts to bypass this defense[49].

4.4 DNS server structure

As we mentioned in the previous section, DNS has its structural problems. In the hierarchical structure, if a domain on the lowest level does not exist or has a problem, the DNS query processed from the top level may be contaminated. Due to the structural weakness, DNS can easily be attacked, resulting in a large number of victims connected to the DNS server. Figure 11 explains how the DNS attack with the DNS server structure vulnerability works.

4.4.1 DA10. DNS non-existent domain

Non-existent domain (NXDOMAIN) is one of the DNS response queries, which means that a domain does not exist. An attacker sends numerous queries to DNS servers for non-existent domains. The DNS servers try to process the queries to find non-existing domains, but they send back the NXDOMAIN

queries because the domains do not exist. Eventually, the cache in the recursive DNS server could be filled with NXDOMAIN results and users will experience slower DNS server response times for legitimate DNS requests. The authoritative DNS servers also spend valuable resources due to the multiple recursive queries to obtain resolution results[50].

4.4.2 DA11. phantom domain

The phantom domain attack is similar to the DNS NXDOMAIN attack. However, the major difference is that attackers use multiple phantom domains to interfere with normal DNS resolution. First, an attacker sets up several phantom domains which either respond very slowly or do not respond to DNS requests. Then, numerous bots send malicious DNS queries for the phantom domains to DNS resolvers. The DNS resolvers handle and deliver the queries to the authoritative servers. However, under the phantom domain attack, the DNS resolvers will continue to wait for responses and continue to query the unresponsive servers, which consumes their resources. As a result, the DNS resolvers' resources are used to process the queries for the phantom domain, and users could be delayed or unable to receive responses to normal DNS queries[51].

4.5 Assessment of DNS attacks

To classify DNS attacks, the types of attacks first are evaluated for each factor. Figure 12 shows the assessment of the 11 DNS attacks introduced in this paper. There are five criteria for evaluating DNS attacks. First is the Attack Method, as described above. The Effect factor classifies attacks according to their intended outcome. The Attack Mode factor refers to whether the attack is passive (i.e., takes place in response to a user-initiated query) or aggressive (launched by the attacker). The Attack Source/Target classifies the multiplicity of attack source(s) and target(s). The Location of Attack Target factor means the location where the attack is executed. If an attacker attempts to attack the DNS infrastructure directly, it is labeled "Internal". Otherwise, if an attacker attempts to attack a target using the DNS infrastructure, it is labeled "External".

The assessment for each factor is a filled circle, meaning fully or completely, half-filled circle, meaning partially, and empty circle, indicating does not apply or not at all. DNS attacks have a variety of purposes. Hijacking/poisoning-based attacks (DNS cache poisoning, Kaminsky, and DNS hijacking) mainly have attack targets to lead to specific malicious sites, while flooding-based attacks (DNS reflection and amplification, DNS flooding, Random sub-domain, DNS NXDOMAIN, and Phantom domain) have the purpose to exhaust DNS servers' resources through direct and aggressive attacks from malware-infected Botnets. van Rijswijk-Deij *et al*.[35] found that DNSSEC could be exploited as DNS reflection attacks. Thus, this attack can target specific servers as well as DNS servers. Finally, attacks that hide their attacks in normal DNS packets or procedures have the purpose of exploiting DNS.

Based on the assessment, Figure 13 shows the classification of DNS attacks by purpose.

(1) DNS Server Unable/Slow: These attacks target DNS servers. The attacker sends a flood of queries to a DNS server, and then the DNS server is forced to exhaust server resources to handle the enormous queries. Eventually, the DNS server will not function normally and not be able to provide the domain service to the user.

(2) Specific Target Server Unable: These attacks target a specific server. The attacker attempts to send heavy traffic to the target server through flooding from the DNS servers. Attackers exploit open DNS resolvers to amplify heavy traffic volume, as a third party[52]. The victim server receives a number of legitimate DNS responses and finally, is subjected to a denial of service attack.

(3) Malicious Website: These attacks provide malicious websites to victims despite requests with normal domains is a DNS Poisoning attack. By manipulating normal response queries, an attacker can illegally acquire and exploit user information by providing bogus IP addresses to the user.

		DNS Cache Poisoning	Kaminsky	DNS Hijacking	DNS Reflection and Amplification DoS Attack	DNS Flooding Attack	Random Sub-domain	DNS NXDOMAIN	Phantom Domain	Domain Generation Algorithms	DNS Tunneling	Fast Flux
Attack Method	Flooding	○	○	○	●	●	●	●	●	○	○	○
	Poisoning / Hijacking	●	●	●	●	○	○	○	○	○	○	○
	Malware	○	○	○	●	●	●	●	●	●	●	●
	DNS Server Attack	◐	◐	◐	●	●	●	●	●	◐	◐	◐
Effect	Target Server disable	○	○	○	●	○	○	○	○	◐	◐	◐
	Move to maricious site	●	●	●	○	○	○	○	○	○	○	○
	aim to hide attack	○	○	○	○	○	○	○	○	●	●	●
	DNS Server disable	○	○	○	●	●	●	●	●	○	○	○
Attack Mode	passive	●	●	●	○	○	○	○	○	●	●	●
	aggressive	○	○	○	●	●	●	●	●	○	○	○
Attack Source / Target	One to One	○	○	○	○	○	○	●	○	○	○	○
	Many to One	○	○	○	●	●	●	○	●	○	○	○
	One to Many	●	●	●	○	○	○	○	○	●	●	●
Location of Attack Target	DNS Internal	●	●	●	○	○	○	○	○	●	●	●
	DNS External	○	○	○	●	●	●	●	●	○	○	○

Figure 12. DNS attacks Assessment. DNS: Domain Name System

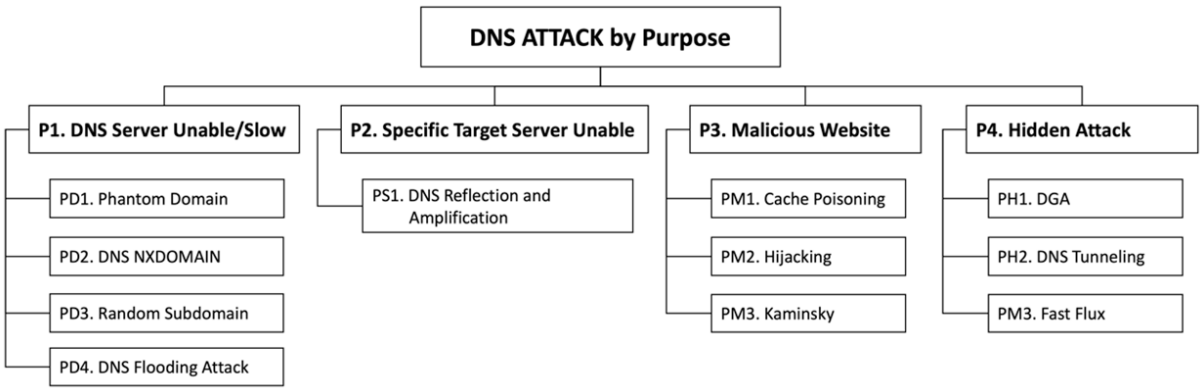


Figure 13. Classification: DNS Attacks by Purpose. DNS: Domain Name System

(4) Hidden Attack: These attacks abuse DNS servers to hide their attack location or attack message. The attacker tries to conceal the location of C&C servers or to exfiltrate the botnet command from C&C, using a vulnerability in internal DNS.

5. MITIGATION

Although DNS has suffered from many attacks, researchers' efforts to mitigate these attacks are ongoing. In particular, DNSSEC, which is the product of their efforts, has helped ensure the integrity of the unreliable DNS data as the main vulnerability of DNS. Additionally, various advanced methods have been introduced to overcome a number of limitations. This section briefly describes them.

5.1 DNSSEC and redundant DNS

Common DNS attacks, such as cache poisoning and spoofing attacks, occur easily by forging DNS data and disguising fake DNS queries. Designed to overcome these problems, DNSSEC uses digital signatures to authenticate the contents of DNS responses, preventing the use of forged DNS data and enhancing the reliability of DNS queries.

As discussed in Section III, DNSSEC suffers from technical complexity, overhead, and low deployment[8]. In 2018, NS1[53] has developed DNSSEC guidelines, so that DNSSEC can be configured correctly and used more easily. However, this does not solve all DNS security issues, including vulnerability to DDoS attacks. The additional length of DNSSEC responses exacerbates the problems of reflection and amplification (DDoS attacks). This dilemma is a major challenge for DSSEC to address in the future.

Redundant DNS servers are one solution to attacks on availability. The DNS standard specified that up to eight spare servers may be used for redundancy[54], so that if a server is unreliable or unavailable, another server can provide name lookup for the user[55]. However, these settings are rarely used in practice by enterprises and ISPs[56], although redundancy has been recommended for a long time.

Ansari *et al*.[57] introduced a new technique to overcome the limitation of DNSSEC and reinforce DNS security, based on using Cloud services for availability and reliability. The redundancy, flexibility, and managed nature of the cloud make it a promising solution for DNS security.

5.2 Existing DNS mitigation systems

A number of approaches for securing DNS have been proposed. We describe these systems by grouping them into three categories: Monitoring and Detection Systems, security extensions on DNS records, and Advanced DNS with additional security functions.

5.2.1 Monitoring and detection systems

DNS is vulnerable to the threat of counterfeited data. One approach is to detect and monitor forged data to distinguish reliable DNS data. The following systems are representative DNS defense systems that include these functions.

(1) Kopis System[58]: Independently detects malware-related domains at the higher levels of the DNS hierarchy (e.g., TLD level) by monitoring network traffic at a high level of the DNS hierarchy. In particular, the Kopis System analyzes the streams of DNS queries and responses at authoritative name servers. From the monitored DNS traffic, they extract the statistical features such as the diversity in the network locations and the reputation of the IP space into which the domain name resolves. Kopis can predict malware-related domains based on monitored traffic patterns with a statistical classification which is determined from higher DNS levels' information. This feature is different from existing detection systems such as Notos[59] (see below) or Exposure[60]. Even without current IP reputation information, Kopis can accurately detect

malware-related domains.
(2) Domain Watcher System[61]: A detection system that detects malicious domain names with local and global textual-based features based on machine learning. This system utilizes three textual features of domains - Lexical features, imitation features, and bi-gram features. First, they use the lexical features to combine the existing characteristic data provided by systems such as EXPOSURE[60] or Detection of Phishing Attacks[62] and new characteristics, such as the number of special characters and numeric characters in the domain name or the number of continuous numeric characters, to easily fetch and normalize the pattern. Imitation features and bi-gram features both utilize the domain information, but imitation looks at the distance between domain names, while bi-gram looks at the similarity of the distribution of letters in domain names.
(3) Anax[63]: A DNS protection system that detects the cache poisoning attack using a large set of open recursive DNS servers (ORDNSs), identifying poisoned DNS caches through DNS records. An infrastructure is added to intercept DNS responses (DNS Scanning Points) and collect and process the resulting data (DNS Data Collector). A Data Preparation Engine analyzes and labels this data, offline, in training mode. A Detection Engine then monitors in real-time DNS responses and flags suspicious responses as poisoning attempts.
(4) Notos-Dynamic Reputation System for DNS[59]: a dynamic reputation system to compute scores of domain names. The goal is to determine if a domain is legitimate or malicious using malicious domains' distinctive features or characteristics.

Other methods of DNS attack detection have been proposed. Zhang *et al.*[64] introduces a new detection method based on machine learning and hybrid methods, which obtains DNS data through active domain name data or passive domain name data. Palau *et al.*[65] proposes an approach to detect DNS tunneling, based on a Convolutional Neural Network (CNN) with a minimal architecture complexity. Also, they use their dataset that contains DNS Tunneling domains generated with five well-known DNS tools. The resulting CNN model correctly detected more than 92% of total Tunneling domains with a false positive rate close to 0.8%. Rajendran *et al.*[66] uses specific properties of DNS amplification and DNS tunneling attacks and presents a number of countermeasures and mitigation techniques to protect against these attacks on the DNS infrastructure.

Fast Flux generates a variety of domain names based on specific algorithms to avoid suppression. Normal DNS-based detection approaches and blacklist filtering are ineffective against the Fast Flux attack. Methods for analyzing new DNS traffic patterns using these Fast Flux characteristics have been developed. These methods recognize the overwhelmingly large or abnormal DNS traffic, filtering the suspicious DNS mapping, and detecting domains of pseudorandom strings generated by the algorithm compared with legitimate domain patterns[67-69]. In particular, DNSMap[67] can quickly identify excessive DNS traffic in real-time by analyzing the DNS mapping of abnormal domains and IP addresses through graphical analysis, unlike conventional methods of domain analysis based on machine learning.

5.2.2 Security extension of DNS records
DNS records provide information about domains that are needed by users. More information may be added to provide data integrity and improve/extend trust. Several methods attempt to do so with less overhead than DNSSEC.

(1) The Transaction SIGnature (TSIG) using CGA (Cryptographically Generated Addresses) Algorithm in IPv6[70]: DNS has a security problem between the client and the DNS resolver due to the untrustworthy resolver as discussed in the 'Vulnerabilities' section. To address this issue, TSIG is used. TSIG establishes a trust relationship between a client and a DNS server. This process provides not only end-to-end authentication but also data integrity between each other through a one-way hash algorithm and shared

keys. However, TSIG faces one problem that it requires the keys is exchanged manually. A solution to the key distribution problem is TSIG using CGA. TSIG-CGA provides an automated way for the negotiation of a shared secret key, with authentication of the host via IPv6's CGA algorithm.
(2) DNS-Based Authentication of Named Entities (DANE)[71-73]: DANE takes advantage of the source of trust provided by DNSSEC to authenticate transport layer security (TLS) certificates. Through TLSA records in the DNS hierarchy, DNSSEC can verify the integrity of DNS data. DANE was designed to provide a stronger trust anchor using DNS as the root. Especially, DANE uses the DNSSEC chain of trust to authenticate X.509 certificates used for transport layer security (TLS) and, as it relies on DNSSEC infrastructure, it can support authentication and data integrity. DANE allows domain owners to issue their certificates without CAs. Using the DNS hierarchy as a single trust anchor instead of many existing CAs, DANE greatly reduces the attack surface. DANE can be used to solve issues related to CAs' vulnerability through the use of a new DNS resource record type, TLSA, signed with DNSSEC. As a result, DANE allows TLS users to better control certificate validation.
(3) DNS-over-HTTPS (DoH)[74]: DoH is a standard web protocol to send DNS traffic over HTTPS. DoH is developed to prevent fundamental DNS privacy problem of unencrypted communication between users and DNS resolvers. As shown in the previous section, without a trusted DNS resolver, DNS queries cannot be guaranteed. In DoH, by using HTTPS's security platform, DNS queries and responses are protected. Moreover, DNS traffic and requests are not directly observable because DoH applies the same port 443 used by HTTPS traffic. Additionally, DoH can be provided by existing DNS servers using a built-in web server. Starting with Mozilla Firefox and Google Chrome in 2018, most major web browsers support or plan to support DoH. Despite this, there are some drawbacks to DoH. First, DNS traffic is encrypted, making it difficult to track/analyze. Mitigation systems that detect DNS attacks based on DNS data analysis will fail to function. Second, the prerequisite for DoH is the support of a trusted DNS resolver. Each web browser, such as Firefox-Cloudflare and Chrome-Google OpenDNS, provides a trusted open DNS resolver. However, traffic is centralized with a few DNS resolvers, with corresponding privacy and performance concerns. Finally, the policies of these enterprises will be difficult to ensure transparency in DNS operations.

5.2.3 Advanced DNS with additional secure functions

According to the DNSSEC deployment tracking system SecSpider[75], current DNSSEC-enabled zones number approximately 3.3 million. It seems that the full deployment of DNSSEC will take considerable time despite many efforts. Thus, additional security functions for DNS are required. The following are methods for improving DNS security.

(1) DNS Proxy Server (DPS) and BIND[76]: a new approach to detect cache poisoning attacks and then send an additional request for the same DNS Resource Record using a local proxy for the BIND caching server. This defensive system makes cache poisoning attacks more difficult.
(2) T-DNS[77]: DNS uses unconnected UDP as the standard protocol. However, because of the poorly secured UDP protocol, DNS is subject to attacks such as spoofing and flooding. T-DNS uses TCP and TLS to provide DNS security. T-DNS provides more secure DNS data through TCP encryption, reduces the impact of DoS attacks by establishing mutual connections, and overcomes the limitations of UDP's response size. DNS based on TLS can provide more secure privacy, support large payload, and mitigate spoofing and reflection DDoS attacks compared to the use of existing UDP protocols. However, the fundamental problems of TCP, latency, and resource needs, remain.
(3) S-DNS[78]: A security solution to prevent DNS cache poisoning and spoofing attacks. Based on the predictability measures and timing analysis, S-DNS mitigates man-in-the-middle attacks in the DNS hierarchy. This protocol has effects on decreasing the probability of the attack and also provides a simple security mechanism with light-weight computation and overheads.
(4) Response Rate Limiting[43]: A defense mechanism to reduce the impact of DNS amplification attacks

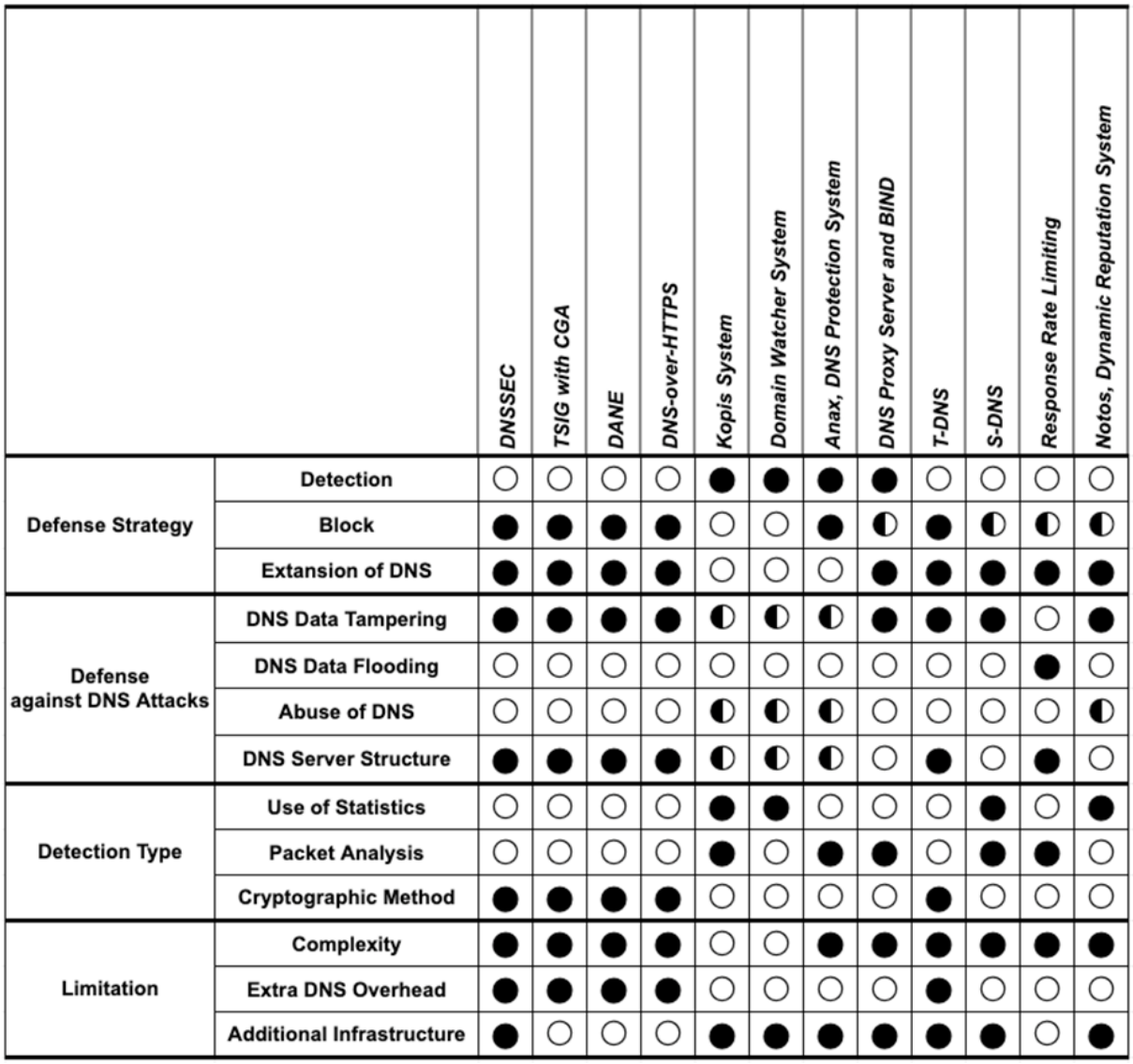


Figure 14. Assessment of DNS Mitigation. DNS: Domain Name System

and reflection attacks. The DNS server will respond a limited number of times to requests for a domain name resolution from a particular IP address, making it more difficult to flood the victim with traffic.

5.3 Overall assessment of DNS mitigation system

Figure 14 shows the assessment of whether the mitigation system can protect against DNS attacks.

A full circle denotes yes or fully, a half-circle denotes partially, and empty circles denote no or not at all. Each mitigation system was developed to solve specific vulnerabilities in DNS. Several key findings of our assessment are provided:

(1) DNSSEC is a major enhancement to DNS but can be exploited for DDoS attacks. According to the 2019 report released by Neustar[79], the number of DDoS attacks increased by 133% and the average DDoS attack size is 7.5 Gbps compared to 2018.

(2) Most monitoring and detection systems can observe the malicious DNS traffic, not protect against the attacks. But, using these mitigation systems, it is possible to filter or protect against the DNS data attacks.

	Google Public DNS	Microsoft Azure	Cloudflare	IBM Quad9	Cisco OpenDNS	Akamai	Oracle	Infoblox	NS1	Verisign	Neustar
DNSSEC	●	○	●	●	●	●	○	●	●	●	●
Certificate Transparency	●	●	●	●	●	●	●	●	●	●	●
CAA Record	●	○	○	○	●	●	●	●	●	●	●
TLS 1.0	○	○	●	○	●	●	●	●	○	○	○
TLS 1.1	○	○	●	○	●	●	●	●	●	○	●
TLS 1.2	●	●	●	●	●	●	●	●	●	●	●
TLS 1.3	●	○	●	●	○	○	○	○	○	○	○
DNS-over-HTTPS	●	●	●	●	●	●	○	●	○	○	○
DNS-over-TLS	●	●	●	●	○	●	○	●	●	○	○

Figure 15. List of the 10 Enterprise DNS providers. DNS: Domain Name System; TLS: transport layer security

(3) TSIG with CGA and DANE are solutions to overcome DNSSEC's limitations and are promising alternatives.
(4) Because most advanced DNS mitigation systems with additional security functions are focused on specific security problems in DNS, they do not cover all DNS attacks. On the other hand, T-DNS prevents most of the DNS attacks because they address the fundamental protocol problem in the DNS protocol. However, T-DNS, based on the TCP protocol, greatly helps improve DNS privacy, while its latency is the slower, and overall cost is significant compared to the UDP protocol.

5.4 Secure/enterprise DNS provider

Unlike these mitigation systems which provide additional security functions or monitor/analyze/detection techniques, an openDNS of major companies or organizations that ensure improved security, reliability and speed would be better option to defend against some of the DNS attacks. It is called Secure/Enterprise DNS, which is a fast and reliable DNS service from large organizations. Enterprise DNS centrally manages its security architecture that guarantees a more sophisticated and reliable DNS service.

To better understand the current Enterprise DNS situation, we provide and evaluate a list of 10 large Enterprise DNS providers, as shown in Figure 15. Each organization provides its open DNS and can be set up and used by anyone on their device. Except for Microsoft Azure and Oracle, most providers support DNSSEC. Azure and Oracle protect DNS through their systems.

Another factor is the support of the Certification Transparency and Certification Authority (CAA) records, which are techniques to compensate for weaknesses and defects in the PKI-certificate system. While all organizations provide Certification Transparency, some do not offer CAA records. Regardless of whether DoH or DoT is supported or not, it is judged as the support of a security solution for certificates.

Almost all providers support DoH and/or DoT, except for Oracle and Verisign. We expect that the support of the DoH/DoT would increase with time.

Finally, all providers offer TLS 1.2 for cipher transmission, especially Google, Cloudfare, and Quad9 that support DoH, up to the latest TLS 1.3. Therefore, these institutions are expected to provide more stable DoH based on TLS 1.3 in the future.

6. DISCUSSION

This paper presents a survey of DNS security. The background of basic DNS and DNSSEC was described, with an explanation for the motivation of DNSSEC. DNS is essential for proper operation of the Internet, but it is still subject to a variety of attacks, due to its vulnerabilities, lack of widespread adoption of available mitigation techniques, and limitations of those techniques. These vulnerabilities were described, and DNS attacks were classified based on those vulnerabilities. Also, several methods suggested in the literature for defending against such attacks were summarized.

This survey provides a novel and useful analysis to understand DNS and DNSSEC in terms of cybersecurity. Specifically, the classification of DNS attacks supports understanding and analysis of future DNS attacks. This paper provides the first DNS attack classification. The analysis of various mitigation systems also provides indicators for future DNS developments. Promising alternatives to DNSSEC include DANE/TLSA and DNS-over-HTTPS. Even lighter-weight approaches, suitable for deployment in the Internet of Things, are needed as well.

DECLARATIONS

Authors' contributions

Contributed to the design, survey, implementation, and analysis of the research and to the writing of the manuscript: Kim TH, Reeves D

Availability of data and materials

Not applicable.

Financial support and sponsorship

Not applicable.

Conflicts of interest

Both authors declared that there are no conflicts of interest.

Ethical approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

REFERENCES

1. Mockapetris P. Domain names-implementation and specification. RFC1035 1987. Available from: https://tools.ietf.org/html/rfc1035. [Last accessed on 17 Aug 2020]
2. Mockapetris P. Domain names-implementation and facilities. RFC1034 1987.
3. Engel S. My ether wallet DNS attack explained. Available from: https://cryptovoid.net/mew-dns-attack-explained. [Last accessed on 17 Aug 2020]
4. Arends R, Austein R, Larson M, Massey D, Rose S. DNS security introduction and requirements. RFC 4033 2005:1-21. Available from: https://tools.ietf.org/html/rfc4033. [Last accessed on 17 Aug 2020]
5. Arends R, Austein R, Larson M, Massey D, Rose S. Resource records for the DNS security extensions. RFC 4034 2005:1-30. Available from: https://tools.ietf.org/html/rfc4034. [Last accessed on 17 Aug 2020]
6. Arends R, Austein R, Larson M, Massey D, Rose S. Protocol modifications for the DNS security extensions. RFC 4035 2005:1-54. Available from: https://tools.ietf.org/html/rfc4035. [Last accessed on 17 Aug 2020]
7. Eastlake 3rd D. Domain name system security extensions. RFC 2535 1999. Available from: https://tools.ietf.org/html/rfc2535. [Last accessed on 17 Aug 2020]
8. Chung T, an Rijswijk-Deij R, Chandrasekaran B, Choffnes D, Levin D, et al. A longitudinal, end-to-end view of the DNSSEC ecosystem. Proceedings of the 26th USENIX Conference on Security Symposium; 2017 Aug; Vancouver, BC. USENIX Association, USA; 2017. pp. 1307-22.
9. NIST. Estimating USG IPv6 and DNSSEC external service deployment status. Available from: https://fedv6-deployment.antd.nist.gov/cgi-bin/generate-gov. [Last accessed on 17 Aug 2020]
10. Roosa SB, Schultze S. Trust darknet: control and compromise in the internet's certificate authority model. IEEE Internet Comput 2013;17:8-25.
11. Wikipedia. 2016 Dyn cyberattack. Avaliable from: https://en.wikipedia.org/wiki/2016_Dyn_cyberattack. [Last accessed on 17 Aug 2020]
12. Downdetector. Internet outage map. Avaliable form: https://downdetector.com/status/centurylink/map/. [Lasted accessed on 27 Jul 2020]
13. NETSCOUT. NETSCOUT's 14th Annual Worldwide Infrastructure Security Report. Avaliable from: https://www.netscout.com/report/. [Last accessed on 17 Aug 2020]
14. Zhauniarovich Y, Khalil I, Yu T, Dacier M. A survey on malicious domains detection through DNS data analysis. ACM Computing Surveys (CSUR) 2018;51:1-36.
15. Fernandes D, Soares LFB, Gomes JV, Freire M, Inácio PRM. Security issues in cloud environments: a survey. Int J Inf Secur 2014;13:113-70.
16. Alieyan K, ALmomani A, Manasrah A, Kadhum, MM. A survey of botnet detection based on DNS. Neural Computing and Applications 2017;28:1541-58.
17. Peng T, Leckie C, Ramamohanarao K. Survey of network-based defense mechanisms countering the DoS and DDoS problems. ACM Comput Surv 2007;39:31-42.
18. Casalicchio E, Caselli M, Coletta A. Measuring the global domain name system. IEEE network 2013;27:25-31.
19. Wikipedia. List of DNS resource records. Available from: https://en.wikipedia.org/wiki/List-of-DNS-record-types. [Last accessed on 17 Aug 2020]
20. Cheshire S, Krochmal M. Multicast DNS, RFC 6762 2013. Available from: https://tools.ietf.org/html/rfc6762. [Last accessed on 17 Aug 2020]
21. Aboba B, Thaler D, Esibov L. Link-local multicast name resolution (LLMNR), RFC 4795, January 2007. Available from: https://www.rfc-editor.org/info/rfc4795. [Last accessed on 17 Aug 2020]
22. Andress J. The basics of information security: understanding the fundamentals of InfoSec in theory and practice, 2nd ed. Syngress; 2014. p. 240.
23. Bates S, Bowers J, Greenstein S, Weinstock J, Xu Y, et al. Evidence of decreasing internet entropy: the lack of redundancy in DNS resolution by major websites and services. Available from: https://www.nber.org/papers/w24317. [Last accessed on 17 Aug 2020]
24. Schiffman M. Bound by tradition: a sampling of the security posture of the internet's DNS servers. LinuxSecurity 2003. Available from: http://packetfactory.openwall.net/papers/DNS-posture/DNS-posture-1.0.pdf. [Last accessed on 17 Aug 2020]
25. Migault D, Cédric G, Laurent M. A performance view on dnssec migration. 2010 International Conference on Network and Service Management (CNSM); 2010 Oct 25-29; Niagara Falls, Canada. IEEE; 2010. pp. 469-74.
26. Klein A. BIND 9 DNS cache poisoning. SecuriTeam 2007. Available from: https://securiteam.com/securitynews/5vp0l0um0a/. [Lasted accessed on 28 Jul 2020]
27. Yu X, Chen X, Xu F. Recovering and protecting against DNS cache poisoning attacks.2011 International Conference onInformation Technology, Computer Engineering and Management Sciences (ICM); 2011 Dec 26-28; Beijing, China. IEEE; 2011. pp. 120-3.
28. Ager B, Dreger H, Feldmann A. Predicting the DNSSEC overhead using DNS traces. In 2006 40th Annual Conference on Information Sciences and Systems. 2006, March 22-24, Princeton, NJ, USA. IEEE; 2006.
29. Van Adrichem NLM, Blenn N, Lua AR, Wang X, Wasif M, et al. A measurement study of DNSSEC misconfigurations. Secur Inform 2015;4:1-14.
30. Deccio C, Sedayao J, Kant K, Mohapatra P. Quantifying and improving dnssec availability. 2011 Proceedings of 20th International Conference on Computer Communications and Networks (ICCCN); 2011 Jul 31-Aug 4; Lahaina, HI, USA. IEEE; 2011. pp. 1-7.
31. Clark L. A cartoon intro to DNS over HTTPS. Avaliable from: https://hacks.mozilla.org/2018/05/a-cartoon-intro-to-dns-over-https/. [Last accessed on 17 Aug 2020]

32. Droms R, Arbaugh W. Authentication for DHCP messages. RFC 3118. Avaliable from: https://tools.ietf.org/html/rfc3118. [Last accessed on 17 Aug 2020]
33. Bau J, Mitchell JC. A security evaluation of DNSSEC with NSEC3. Proceedings of the Network and Distributed System Security Symposium, 2010 Feb 28-Mar 3; San Diego, California, USA. NDSS; 2010. pp. 18.
34. Internet society. State of DNSSEC deployment 2016. Avaliable from: https://www.internetsociety.org/resources/doc/2016/state-of-dnssec-deployment-2016. [Last accessed on 17 Aug 2020]
35. van Rijswijk-Deij R, Sperotto A, Pras A. DNSSEC and its potential for DDoS attacks: a comprehensive measurement study.Proceedings of the 2014 Conference on Internet Measurement Conference; 2014 Nov; Vancouver, BC, Canada. ACM; 2014. pp. 449-60.
36. Loveless J. DNSSEC: how Savvy DDoS attackers are using our defenses against us, Security Research Report by Neustar 2016. Avaliable from: http://www.circleid.com/posts/20160818_how_savvy_ddos_attackers_are_using_dnssec_against_us/. [Last accessed on 17 Aug 2020]
37. Alharbi F, Chang J, Zhou YC, Qian F, Qian ZY, et al. Collaborative client-side DNS cache poisoning attack. IEEE INFOCOM 2019-IEEE Conference on Computer Communications. 2019. Apr 29 - May 2; Paris, France. IEEE, 2019.
38. Kaminsky D. Black ops 2008: It's the end of the cache as we know it. Black Hat USA 2008; 2. Avaliable from: https://www.blackhat.com/presentations/bh-jp-08/bh-jp-08-Kaminsky/BlackHat-Japan-08-Kaminsky-DNS08-BlackOps.pdf. [Last accessed on 17 Aug 2020]
39. Vissers T, Barron T, van Goethem T, Joosen W, Nikiforakis N. The wolf of name street: hijacking domains through their nameservers. Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security; 2017 Oct; Dallas, Texas, USA. ACM; 2017. pp. 957-70.
40. Rascagneres P. Mercer W. DNSpionage campaign targets middle east. Available from: https://blogs.cisco.com/security/talos/dnspionage-campaign-targets-middle-east. [Last accessed on 17 Aug 2020]
41. Thornewell PM, Golden LM. DNS flood protection platform for a network. US Patent 2012;8,261,351. Available from: https://portal.unifiedpatents.com/patents/patent/US-8261351-B1. [Last accessed on 17 Aug 2020]
42. Rozekrans T, Mekking M, de Koning J. Defending against DNS reflection amplification attacks. University of Amsterdam System & Network Engineering RP1 2013. Available from: https://www.nlnetlabs.nl/downloads/publications/report-rrl-dekoning-rozekrans.pdf. [Last accessed on 17 Aug 2020]
43. Chandramouli R, Rose S. Secure domain name system (DNS) deployment guide. NIST Special Publication 2006;800:81-2.
44. Feibish SL, Afek Y, Bremler-Barr A, Cohen E, Shagam M. Mitigating DNS random subdomain DDoS attacks by distinct heavy hitters sketches. Proceedings of the fifth ACM/IEEE Workshop on Hot Topics in Web Systems and Technologies; 2017 Oct; San Jose, California. New York, NY, USA: Association for computing Machinery; 2017. pp. 1-6.
45. Farnham G, Atlasis A. Detecting DNS tunneling. SANS Institute InfoSec Reading Room 2013;9:1-32.
46. van Leijenhorst T, Chin KW, Lowe D. On the viability and performance of DNS tunneling. The 5th International Conference on Information Technology and Applications (ICITA); 2008. pp. 560-6.
47. Zhou Y, Li Q, Miao Q,Yim K. DGA-based botnet detection using DNS traffic. JInternet ServInfSecur 2013;3:116-23.
48. Kessem L. The Necurs Botnet: a pandora's box of malicious spam. Security Intelligence. Acaliable from: https://securityintelligence.com/the-necurs-botnet-a-pandoras-box-of-malicious-spam/. [Last accessed on 17 Aug 2020]
49. Metcalf LB, Ruef, Spring JM. Open-source measurement of fast-flux networks while considering domain-name parking. The LASER Workshop: Learning from Authoritative Security Experiment Results (LASER 2017); 2017 Oct 18-19; USENIX Association; 2017. pp. 13-24.
50. Dagon D, Lee C, Lee W, Provos N. Corrupted DNS resolution paths: The rise of a malicious resolution authority. Proceedings of the 15th Network and Distributed System Security Symposium (NDSS); 2008 Feb 10-13; San Diego, California, USA. NDSS; 2008.
51. Mergenhagen P, Domain DP. Mainstreethost. Available from: https://www.mainstreethost.com/blog/deindexing-phantom-domains. [Last accessed on 10 Aug 2020]
52. Krämer L, Krupp J, Makita D, Nishizoe T, Koide T, et al. Amppot: monitoring and defending against amplification ddos attacks. International Symposium on Recent Advances in Intrusion Detection; 2015 Nov 2-4; Kyoto, Japan. Springer; 2015. pp. 615-36.
53. NS1. Enabling DNSSEC. Available from: https://ns1.com/knowledgebase/dnssec. [Last accessed on 27 Jul 2020]
54. Elz R, Bush R, Bradner S, Patton M. Selection and Operation of Secondary DNS Servers. RFC 2182 1997. Available from: https://tools.ietf.org/html/rfc2182. [Last accessed on 27 Jul 2020]
55. Yu Y, Cai J, Osterweil E, Zhang L. Measuring the placement of DNS servers in top-level-domain. Verisign Technical Report 2011. Available from: https://www.semanticscholar.org/paper/Measuring-the-Placement-of-DNS-Servers-in-Yu/4afb5d97b5002edc7f14708a51d7abb322d28f9a. [Last accessed on 27 Jul 2020]
56. Bisiaux JY. DNS threats and mitigation strategies. Network Security 2014;7:5-9.
57. Ansari A, Khan N, Rais Z, Taware P. Reinforcing security of DNS using AWS cloud. Proceedings of the 3rd International Conference on Advances in Science & Technology (ICAST); 2020 Apr 8-9; Mumbai, India. SSRN; 2020.
58. Antonakakis M, Perdisci R, Lee W, Vasiloglou N, Dagon D. Detecting malware domains at the upper DNS hierarchy. Proceedings of the 20th USENIX Conference on Security; 2011 Aug; USENIX Association. USA; 2011. pp. 1-16.
59. Antonakakis M, Perdisci R, Dagon D, Lee W, Feamster N. Building a dynamic reputation system for DNS. Proceedings of the 19th USENIX Conference on Security; 2010 Aug; USENIX Association. USA; 2010. pp. 273-89.
60. Bilge L, Kirda E, Kruegel C, Balduzzi M. EXPOSURE: finding malicious domains using passive DNS analysis. Proceedings of the Network and Distributed System Security Symposium, 2011 Feb 6-9; San Diego, California, USA. NDSS; 2011.
61. Zhang P, Liu T, Zhang Y, Ya J, Shi J, et al. Domain watcher: detecting malicious domains based on local and global textual features. ProcComputSci 2017;108:2408-12.

62. Muhammet B, Ziya GZ. Detection of phishing attacks. 2018 6th International Symposium on Digital Forensic and Security (ISDFS); 2018 Mar 22-25; Antalya, Turkey. IEEE; 2018. pp. 1-5.
63. Antonakakis M, Dagon D, Luo X, Perdisci R, Lee W, et al. A centralized monitoring infrastructure for improving dns security. Proceedings of the 13th International Conference on Recent Advances in Intrusion Detection; 2010 Sep; International Symposium, Raid, Ottawa, Ontario, Canada. Berlin: Springer-Verlag; 2010. pp. 18-37.
64. Zhang K, Ji W, Li N, Wang Y, Liao S. Detection of malicious domain name based on DNS data analysis. JPhysConfSer 2020;1544:012169.
65. Palau F, Catania C, Guerra J, Garcia S, Rigaki M. DNS tunneling: a deep learning based lexicographical detection approach. Cryptography and Security 2020.
66. Rajendran, B. DNS amplification & DNS tunneling attacks simulation, detection and mitigation approaches. 2020 International Conference on Inventive Computation Technologies (ICICT); 2020 Feb 26-27; Coimbatore, India. IEEE; 2020. pp. 230-6.
67. Berger A, D'Alconzo A, Gansterer WN, Pescape A. Mining agile dns traffic using graph analysis for cybercrime detection. Comput Netw 2016;100:28-44.
68. Perdisci R, Corona I, Giacinto G. Early detection of malicious flux networks via large-scale passive DNS traffic analysis. IEEE T Depend Secure 2012;9:714-26.
69. Yadav S, Reddy AKK, Reddy AN, Ranjan S. Detecting algorithmically generated domain-flux attacks with DNS traffic analysis. IEEEACM TNetwork 2012;20:1663-77.
70. Vixie P, Gudmundsson O, Eastlake D, Wellington B. Secret key transaction authentication for DNS (TSIG). RFC28452000. Available from: https://www.bibsonomy.org/bibtex/fbdc74e947549d1d0939d567bd377f08. [Last accessed on 27 Jul 2020]
71. Barnes R. Use cases and requirements for DNS-based authentication of named entities (DANE). RFC 6394 2011. Available from: https://tools.ietf.org/html/rfc6394. [Last accessed on 27 Jul 2020]
72. Gudmundsson O. Adding acronyms to simplify conversations about DNS-based authentication of named entities (DANE). RFC 7218 2014. Available from: https://tools.ietf.org/html/rfc7218. [Last accessed on 27 Jul 2020]
73. Zhu L, Wessels D, Mankin A, Heidemann J. Measuring dane tlsa deployment. International Workshop on Traffic Monitoring and Analysis; 2015 Apr 21-24; Barcelona, Spain. Springer; 2015. pp. 219-32.
74. Hoffman P, McManus P. DNS queries over HTTPS (DoH). RFC 8484 2018. Available from: https://tools.ietf.org/html/rfc8484. [Last accessed on 27 Jul 2020]
75. SecSpider. Global DNSSEC deployment tracking. Available from: http://secspider.net/. [Last accessed on 17 Aug 2020]
76. Trostle J, van Besien B, Pujari A. Protecting against DNS cache poisoning attacks. 2010 6th IEEE Workshop on Secure Network Protocols; 2010 Oct 5-5; Kyoto Japan. IEEE; 2010. pp. 25-30.
77. Zhu L, Hu Z, Heidemann J, Wessels D, Mankin A, et al. T-DNS: connection-oriented DNS to improve privacy and security. ACM SIGCOMM CompCom 2014;44:379-80.
78. Bassil R, Hobeica R, Itani W, Ghali C, Kayssi A, et al. Security analysis and solution for thwarting cache poisoning attacks in the domain name system. 2012 19th International Conference on Telecommunications (ICT); 2012 Apr 23-25; Jounieh, Lebanon. IEEE; 2012. pp. 1-6.
79. Neustar. Q2, 2019 Cyber threats and trends report. Available from: https://www.home.neustar/resources/whitepapers/2019-cyberthreats-trends-report. [Lasted accessed on 17 Aug 2020]

Jiang *et al. J Surveill Secur Saf* 2020;1:61-78
DOI: 10.20517/jsss.2020.09

Journal of Surveillance, Security and Safety

Original Article

Open Access

Check for updates

Stereo storage structure assisted one-way anonymous auditing protocol in e-health system

Ling-Hong Jiang[1], Chen Wang[1], Jian Shen[1,2,3]

[1]School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing 210044, Jiangsu, China.
[2]Cyberspace Security Research Center Peng Cheng Laboratory, Shenzhen 518000, Guangdong, China.
[3]Henan Key Laboratory of Network Cryptography Technology, Zhengzhou 450000, Henan, China.

Correspondence to: Prof. Jian Shen, Jiangsu Engineering Center of Network Monitoring, School of Computer and Software, Nanjing University of Information Science and Technology, 219 ningliu Road, Nanjing 210044, Jiangsu, China.
E-mail: s_shenjian@126.com

How to cite this article: Jiang LH, Wang C, Shen J. Stereo storage structure assisted one-way anonymous auditing protocol in e-health system. *J Surveill Secur Saf* 2020;1:61-78. http://dx.doi.org/10.20517/jsss.2020.09

Received: 9 Apr 2020 **First Decision**: 29 Jun 2020 **Revised**: 30 Jul 2020 **Accepted**: 13 Aug 2020 **Available online**: 24 Sep 2020

Academic Editor: Stefanos Gritzalis **Copy Editor**: Cai-Hong Wang **Production Editor**: Jing Yu

Abstract

Aim: With the popularity of cloud storage, data integrity has become a hot research spot. As clients' data is outsourced to the cloud, how to prevent clients' privacy from being leaked has become an urgent problem to be solved. In addition, the design of the storage structure in the cloud is also a challenge. To solve the above problem, we focus on enabling data integrity verification in the medical environment with clients' privacy protection and a novel storage structure assisted.

Methods: By leveraging the one-way anonymous key agreement and the novel stereo storage structure, a novel stereo storage structure assisted one-way anonymous auditing protocol in e-health system is proposed. First, the one-way anonymous auditing protocol can realize the adaptive anonymity of clients in the e-health system. Second, the novel stereo storage structure can implement the storage and fast search of medical data.

Results: The theoretical analyses indicate that the proposed scheme is secure under the Computational Diffie-Hellman problem and Discrete algorithm problem and it has a decent performance in computational overhead. Besides, the simulation results demonstrate that the computational cost of the user is constant.

Conclusion: To protect the user's private information in e-health system, we propose a stereo storage structure assisted one-way anonymous auditing protocol in this paper. In the proposed scheme, fast searching of data,

the one-way anonymity and the data auditing with mutual supervision are supported, which is necessary for the patients and the medical personnel in a real e-health scenario.

Keywords: Data integrity, stereo storage structure, one-way anonymous

1 INTRODUCTION

The development of society is inseparable from the advancement of science and technology. As the huge development of the internet industry and the rise of data applications such as artificial intelligence and big data spread throughout people's lives, people begin to generally realize the importance of data. The world is gradually stepping into the era of data dominance, and all walks of life are generating data all the time. Due to the sheer amount of data that is ever increasing, traditional storage methods cannot meet the needs of the people anymore which leads to the emergence of cloud storage.

Cloud platform provides individuals and organizations with powerful storage services which brings great benefits as follows: (1) users only need to pay for the actual storage without worrying about insufficient storage resources; (2) with data hosted to cloud platform, local data will no longer be stored, which can reduce the purchase cost and energy consumption cost of storage equipment; (3) the maintenance of data storage can be left to cloud service providers (CSP) to save the cost of maintaining large amounts of data for users; (4) cloud data can form linkage with local data to make redundant backup to each other; and (5) users can easily access the data in the cloud through web interface or application.

While cloud storage has many advantages, there are also some security threats[1,2]. On the one hand, the cloud infrastructure may suffer some inevitable hardware or software failures or external attacks that lead to data corruption could occur, but cloud server providers could deliberately hide the fact of data corruption for the sake of their business reputation. On the other hand, the outsourced data stored in the cloud might suffer from illegal behaviors from CSP for commercial benefits. What is more, the outsourcing of data results in users no longer physically owning the data, so users cannot even verify whether their data is complete, available, or secure.

Therefore, how to guarantee the data integrity and the privacy of users on the cloud server has become a key issue for cloud storage services. More data security issues are increasingly prominent such as whether the user data is securely stored or whether the user privacy is leaked and so on[3-5]. When it comes to the electronic heath system, a physician records the information of patients' medical treatment electronically through electronic health records which involves the collection, quality control, transmission, storage, statistics, and utilization of patient information. Obviously, it is difficult for traditional storage methods to screen and retrieve typical health records for medical statistics and scientific research. An electronic health system can not only retrieve all kinds of medical records quickly, but also make the previously laborious process of obtaining medical statistics become very simple and fast, providing first-hand data for scientific research and teaching. Nevertheless, such information often contains confidential and sensitive information, and the disclosure or falsification of such information may damage the reputation and property of patients.

In order to address these issues, considerable efforts have been made. Among existing proposals, great amounts of cloud data integrity auditing schemes based on privacy protection have been proposed[6-9]. To verify whether the outsourced data remains intact, file owners or auditors can challenge the cloud server with low communication overheads and computation costs.

Motivation of this paper: Medical data include patients' information such as admissions, discharges, transfers, e-health system patient records, diagnoses, treatments, medical images, economic/financial data, and so on. The quality, confidentiality, and integrity of medical data will affect the real-time, short-term, and long-term performance of the application. First, it will directly affect the daily management and treatment of patients. Second, the application of software and systems for obtaining information and decision support may be affected. Third, there are unknown impacts data storage failure may cause on medical research which can lead to irreparable consequences. At present, researchers have designed many protection schemes for data in the cloud. However, there is no complete data protection scheme specially designed for medical data.

1.1 Our contributions

To solve the above security protection problem of cloud medical data, this paper designs a one-way anonymous auditing protocol in the e-health system. The contributions of this paper can be summarized as follows.

1.1 1 A novel stereo storage structure is proposed to assist the auditing protocol in the e-health system

As stated above, medical data consists of a variety of data types. Therefore, we propose a novel data storage structure to store medical data, which can achieve fast search of data. In addition, the design of this structure saves the storage overhead of index tables.

1.1.2 A one-way anonymous e-health system model is presented

In view of the current status of the medical environment, for better protection of the privacy of patients, we propose an e-health system model that supports one-way anonymity, which means patients in this system model can keep their identities anonymous. Simultaneously, medical personnel identity information is disclosed in the system, so that patients can find the responsible person when a medical accident occurs.

1.1.3 An auditing protocol aiming to support both physician and patient validation is provided

This scheme innovatively enables patients and attending physicians to independently verify the integrity of their commonly relevant medical data. In other words, both patients and their attending physicians can verify whether medical data file in the cloud is correct and complete. In addition, it can promote information exchange and mutual supervision between physicians and patients.

1.2 Related works

In the past few years, data integrity in the cloud has received much attention as a core security issue. Hereafter, abundant security models and data protection schemes have been proposed by researchers around the world to solve the integrity audit problem of outsourced data[10,11]. In 2003, Deswarte *et al.*[12] first put forward the theoretical model of remote verification of data integrity of untrusted servers based on the Diffie-Hellman key agreement protocol. The proposed model consists of only two entities, the user and the cloud server provider. The user can directly initiate data integrity verification to the cloud service provider, laying a foundation for the subsequent cloud data auditing protocol. At that time, cloud storage was not yet widespread, and only a few users outsourced a small amount of data on remote servers, so that the protocol did not take into account a situation where a large community of users are storing a great deal of data on cloud servers which we see today. Once the data stored by the user on the remote server is too large, the computing overhead on the user side cannot be borne by ordinary computers, and the protocol cannot work normally. Thus, to solve that problem, a third-party auditor entity is introduced to validate the integrity of the outsourced data in the cloud.

With a growing number of users using the storage service on the cloud, cloud data auditing protocols are rapidly being developed, and many scholars are proposing plentiful valuable solutions. In 2007, Ateniese *et al.*[13]

firstly put forward a notion of Provable Data Possession to confirm the outsourced data possession on the untrusted cloud, which is based on RSA homomorphic linear verification and supports third-party public auditing. However, the dynamic update of data is not supported in this scheme, and this scheme cannot protect users' privacy. In the same year, Juels *et al.*[14] proposed a model named Proof of Retrievability, as well as presented a practical scheme which supports the integrity verification of data and the recovery of damaged data. Nevertheless, this scheme has a limited number of times to verify data integrity and does not support dynamic auditing or batch auditing. Since then, to solve the aforementioned problems, many scholars have devoted themselves to making improvements based on these two schemes, and they have made great progress in supporting more performance such as batch auditing, operating efficiency, and dynamic data update. Nevertheless, few people paid attention to the problem that these schemes leak users' private data to third-party auditors in the process of auditing. In 2010, Wang *et al.*[15] first proposed an auditing scheme that can be publicly verified to support user privacy protection. This scheme is based on public key homomorphic label technology so that the auditor can perform auditing without obtaining all the data of the user which greatly increases the operating efficiency of the system. The scheme also uses a random masking technique which makes it impossible for third party auditors to obtain users' private information through the verification returned by cloud service providers. In addition, the auditing protocol supports dynamic update of data, batch auditing, and multiple auditing tasks that can be performed simultaneously. It was later confirmed that there were still security risks. Therefore, in 2011, Wang *et al.*[16] improved the system for the security but caused a huge computing burden on the cloud server, greatly reducing the efficiency of system operation. In terms of this problem, in 2015, Worku *et al.*[17] increased the efficiency of system operation while ensuring data security, but unfortunately, it did not support dynamic data operations.

Besides storage data, users would like to perform updates to outsourced data directly in the cloud. Based on this, Wang *et al.*[18] proposed a relatively complete protocol which can support data update, user privacy protection, and batch auditing, but it will lead to the problem of high computing cost on the client side. Then, Garg *et al.*[19] designed a protocol that can minimize the computational complexity for the client during the system setup phase, which is publicly verifiable and supports dynamic operations on data.

After that, many multi-user modification and user revocation schemes have been proposed[20-23]. However, the above scheme cannot solve the problem of data redundancy well. To solve that problem, Wu *et al.*[24], Daniel and Vasanthi[25] removed redundant data from the cloud server which saved the storage cost of cloud service providers and greatly improved the efficiency of data validation. However, none of the above schemes have been designed specifically for images stored on the cloud, thus Tang *et al.*[26] proposed an efficient real-time integrity auditing protocol specially designed for cloud images, which also supported fair arbitration. In 2019, based on a new primitive fuzzy identity, Zhao *et al.*[27] presented a dynamic auditing protocol for the integrity verification of big data. This scheme applies fuzzy identity to the integrity verification of big data for the first time.

However, the above existing solutions cannot be well applied to the e-health systems due to the special relationship between medical staff and patients, and the particularity of medical data. Therefore, we explore a novel storage structure for storing medical data for the e-health system and design a one-way anonymous auditing protocol in this paper.

1.3 Organization

The rest of this paper consists of the following parts: We first introduce the preliminaries in Section 2, mainly including some definitions and basic properties about bilinear pairing and one-way anonymous key agreement required for this paper. Then, we describe the system architecture that contains the proposed system model, system components, and stereo storage structure in Section 3. In Section 4, we formalize

the security model of the proposed one-way anonymous auditing protocol. In Section 5, a detailed description of the proposed scheme is demonstrated. After that, a security analysis is presented in Section 6. In addition, performance analysis of our stereo storage structure assisted one-way anonymous auditing protocol in e-health system is given in Section 7. Finally, Section 8 concludes the findings of the paper.

2 PRELIMINARIES

Necessary preliminaries mainly including some definitions and basic properties about bilinear pairing and one-way anonymous key agreement required for this paper are introduced in this section.

2.1 Bilinear pairing

Let G_1 and G_2 be two groups of the same prime order q. Let G_1 be an additive group, and let G_2 be a multiplicative group. A mapping e on (G_1, G_2): $G_1^2 \rightarrow G_2$ satisfying the following properties is named a cryptographic bilinear map[28].

2.1.1 Bilinearity

$e(aP,bQ) = e(P,Q)^{ab}$ for all $P,Q \in \mathsf{G}_1$ and $a,b \in Z_q^*$. This can be expressed in the following manner. For $P,Q,R \in \mathsf{G}_1$, $e(P+Q,R) = e(P,R)e(Q,R)$ and $e(P-Q,R) = e(P,R)e(Q,R)^{-1}$.

2.1.2 Non-degeneracy

If P is a generator of G_1, then $e(P, P)$ is a generator of G_2. That is to say, $e(P, P) \neq 1$.

2.1.3 Computability

e is efficiently computable.

2.2 One-way anonymous key agreement

One-way anonymous key agreement was proposed by Kate *et al.*[29]. Suppose Alice ID_A and Bob ID_B are clients of the same key generation center, whose master secret is s and $d_i = s \cdot H(ID_i)$ for clients with their identity ID. Then, clients can compute a shared key by using their own privacy key and the identity ID of the other participant. What is more, suppose Alice wants to remain anonymous with Bob. Hereafter, the key agreement protocol process can be roughly divided into the following two parts: (1) first, Alice computes $Q_A = H(ID_A)$ and $Q_B = H(ID_B)$. Finally, randomly chooses an integer $r_A \in_R Z_q^*$, computes $P_A = r_A \cdot Q_A$ as Alice's pseudonym and sends it to Bob; (2) after received Alice's pseudonym, Bob computes $K_{AB} = e(P_A, d_B)$. Then, Alice and Bob have the same shared key $K_{AB} = e(d_A, Q_B) = e(Q_A, Q_B)^{r_A \cdot s} = e(P_A, d_B)$.

3 SYSTEM MODEL AND DATA STRUCTURE

3.1 System model

Stereo storage structure assisted one-way anonymous auditing scheme in e-health system involves four entities: key generation center, users, the third-party auditor, and cloud server. Figure 1 illustrates the relationship between those four entities.

3.1.1 User

In our model, patients and physicians are considered as the two main electronic health system (EHS)-related personnel types. For instance, when a patient seeks a diagnosis through interview by a physician in EHS, the patient needs to inform the physician of his or her own information at first. To realize the privacy protection of the patient's identity, our scheme will set up a false name for the patient based on the patient's identity ID to interact with the physician. A physician needs to generate patients' electronic health records (EHRs), which contains basic information about the physician and the patient as well as the patient's medical data, and upload it to the cloud. Although physicians and patients are two different entities, their

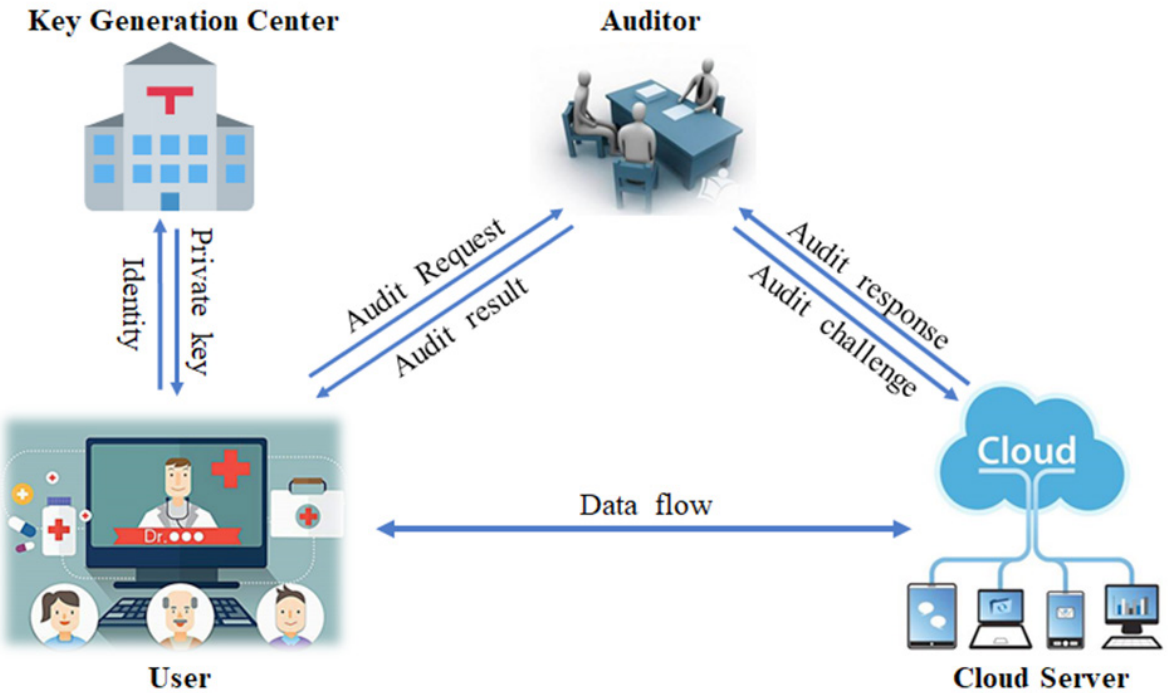


Figure 1. The proposed system model

functional needs for data in the EHS are similar. Therefore, we consider the physicians and the patient as one object in this system. As user, both physicians and patients can access the relevant EHRs and validate the integrity of their data by authorizing the TPA.

3.1.2 Key generation center

The key generation center is a trusted party in e-health system responsible for setting system parameters and generating the corresponding privacy key based on the client's identity and distributing it to the user.

3.1.3 Cloud server

It is supposed that the cloud server is a terminal that provides unlimited computing and storage capacity. Users can upload data through the cloud storage service and share it with other users. During the data integrity auditing process, Cloud server (CS) can respond to the challenges that users delegate to third-party auditor (TPA).

3.1.4 The TPA

TPA is a public verifier, which is assumed to be a terminal with unlimited computing and storage capability. TPA provides data auditing services and is entrusted by users to verify the integrity of cloud data.

3.2 System components

Stereo storage structure assisted one-way anonymous auditing protocol in e-health system consists of the following four algorithms: *Setup, KeyGen, Extract,* and *Audit*. Specifically, these algorithms are described as follows:

$Setup(1^{\kappa}) \rightarrow (para, msk)$: On input 1^{κ} where κ is a security parameter, the system setup algorithm, which is a probabilistic algorithm run by the Key generation center (KGC), generates the public parameter *PP* for the system and a master secret key *msk* for the KGC itself.

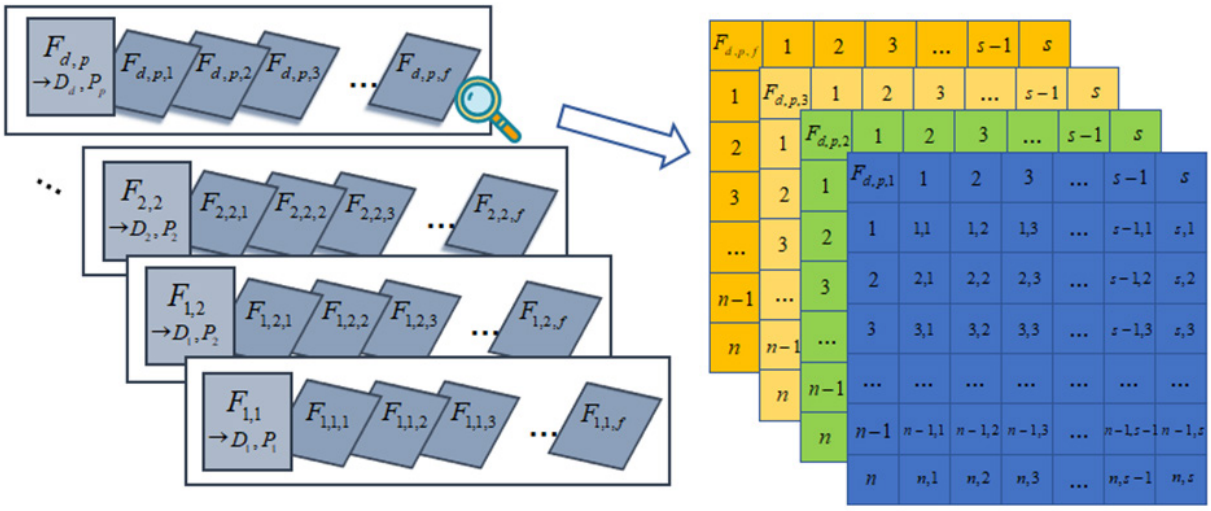


Figure 2. The presented data structure model

$KeyGen(PP,\ msk\ ,\ ID_A,\ ID_B) \rightarrow (d_A, d_B, K_{AB}, KAB)$: This algorithm is a probabilistic algorithm implemented by KGC. The public parameter *PP*, the master key *msk*, and patient's identity ID_A along with physician's identity ID_B are the inputs, and *KeyGen* generates a private key d_A for patient A and a private key d_B for physician B. This algorithm outputs a session key K_{AB} and secret key KAB for auditing.

$Extract(PP, F, KAB) \rightarrow (F^*, \tau, \{\sigma_i\}_{i\in[1,n]})$: This algorithm is a probabilistic algorithm run by a user. The user is given system parameters *PP*, key KAB, file *F* and its file name. It outputs a verifiable file tag τ, a set of block authenticators $\{\sigma_i\}_{i\in[1,n]}$ of the processed file blocks $\{\chi_i\}_{i\in[1,n]}$.

$Audit(PP, \tau) \rightarrow \{0,1\}$: This algorithm is a probabilistic algorithm jointly run by the auditor and cloud server. It outputs 1 to indicate all of the data block can be verified to be original and integrated by τ.

3.3 Stereo storage structure

The novel stereo storage structure proposed in this paper is aimed to realize fast retrieval and query of data and assist the auditing protocol in the e-health system. As is shown in Figure 2, a three-dimensional storage structure is designed to store mass amounts of medical data from the users. Specifically, each plane of the three-dimensional structure on the left part of the figure contains a header file and a series of *f* diagnosis and treatment files of a certain physician corresponding to a certain patient. The header file contains the identity information of the physician and the patient, which is convenient for quick search of the file. Here, $1 \le f \le \mathcal{N}$, and $\mathcal{N}$ is the upper limit of file number of each plane in the stereo storage structure. And those medical files contained in one plane can be generated, shared with, and verified for integrity by both of the specific physician and the patient. In other words, all diagnosis and treatment files of a physician D_d for one of his/her patients P_p are stored in the same plane. For example, $F_{1,1,f}$ represents the *f*-th files of the physician D_1 and the patient P_1, and $F_{1,2,f}$ represents the *f*-th files of the physician D_1 and the patient P_2. In the same way, the patient P_2 can also consult with the physician D_2, during which a series of files will be generated. In this e-health system, we suppose the user set contains a set of physician *D* and a set of patient *P*, and the index of the physician and patient is *d* and *p*, respectively. Here, the *f*-th files of the physician D_d and the patient P_p is denoted as $F_{d,p,f}$, and the header file of this series of files in the same plane is represented as $F_{d,p}$. In addition, the *f* files corresponding to one of the planes are shown on the right in the figure, which together form a smaller three-dimensional storage structure. Each plane in the right picture represents a file. In order to better process the file data, we uniformly divide each file into *n* blocks and each block

comprises *s* sectors. Each file and each plane of the stereo storage structure stores data as follows.

$$F_{d,p,f} = \{\chi_{x,y}\}_{1\le x\le n, 1\le y\le s}$$

$$F_{d,p} = \{F_{d,p,f}\}_{d\in D, p\in P, 1\le f\le N}$$

Furthermore, there are a warrant list of corresponding files in the header file of each plane in the structure for the auditing of the log information, which include the file origin, file type, and consistency of outsourced files. Based on this stereo storage structure, we can quickly search any user's file and the corresponding data block fragments to assist one-way anonymous auditing protocol. Additionally, dynamic data updates are an important part of the auditing schemes. However, due to the particularity of medical data, changes in the data may cause irreversible effects on the medical data. Therefore, dynamic data updates in this paper need both patients' and their attending physicians' authorization; however, those updates will not change the division of the original file.

4 SECURITY MODEL

The following security model of the stereo storage structure assisted one-way anonymous auditing scheme is proposed by designing a series of games between an adversary $\mathcal{A}$ and a challenger $\mathcal{C}$. Taking into account in our security model the fact that the cloud server may modify or remove the data in the cloud due to software and hardware failure or man-made destruction, we view the untrusted cloud server as an adversary $\mathcal{A}$ and the user as a challenger $\mathcal{C}$. The formalized security model of the game is as follows:

(1) Setup. Once security parameter κ is inputted in the system, the challenger $\mathcal{C}$ runs the system Setup algorithm, and generates the system public parameter *PP* and a master secret key *msk*. Then, the challenger $\mathcal{C}$ sends the system public parameters *PP* to $\mathcal{A}$.

(2) Query. In this process, $\mathcal{A}$ can spontaneously issue the following two queries to $\mathcal{C}$:
KeyGen Queries: At first, $\mathcal{A}$ queries the secret key for the patient P_A and physician Q_B. Then, $\mathcal{C}$ runs the KeyGen algorithm in the system to generate a secret key *KAB* and sends the secret key to $\mathcal{A}$.
Extract Queries: Then, in these queries, $\mathcal{A}$ adaptively make queries of the signatures for the file M. After $\mathcal{C}$ runs the KeyGen algorithm and gets the secret key, $\mathcal{C}$ runs the Extract algorithm to generate the signatures of the file M. Next, $\mathcal{C}$ sends the signatures of the file M to $\mathcal{A}$.

(3) Challenge. In this phase, $\mathcal{A}$ plays the role of a prover to yield a valid proof and $\mathcal{C}$ acts as a verifier to check out the correctness of the proof. The challenger $\mathcal{C}$ samples a series of random numbers and sends the challenge $chal = \{i, s_i\}_{i\in I}$ to $\mathcal{A}$.

(4) Output. Once receiving the challenge from the challenger $\mathcal{C}$, the adversary $\mathcal{A}$ generates corresponding proof *P* and feedback to $\mathcal{C}$. If this proof *P* can be verified by $\mathcal{C}$ with a non-negligible probability, that is to say, this game ends and $\mathcal{A}$ ultimately successful in the game above.

5 OUR PROPOSED SCHEME

Our proposed scheme is demonstrated in four phases in this section. Firstly, in the system setup phase, the KGC sets the system public parameters and a master secret key. Secondly, the KGC generates privacy keys for users and secret keys for auditing in the registration phase. Next, in the storage phase, users upload and update files to the cloud along with file warrants, authenticators, and tags. Finally, in the integrity verification phase, TPA is entrusted by the data owner to verify corresponding data integrity. Note that for simplicity, some primary notations used throughout the paper are summarized in Table 1. Moreover, the scheme is described in detail as follows:

Table 1. Main notations in the proposed scheme

Notation	Description
H_1, H_2, H_3, H_4	Four hash functions
msk	The master secret key
d_i	The secret key of user i
P_i	The pseudonym of user i
K_{AB}	The session key of user A and B
KAB	The auditing secret key of user A and B
$\tau, \{\sigma_i\}_{i\in[1,n]}$	The file tag and set of block authenticators
Λ, V_N, T_N	The warrant, version number, and time stamp of outsourced files
χ_{ij}	The i-th block j-th sector data of file

5.1 System setup: Setup

Once taking a security parameter κ as input, the KGC randomly selects two multiplicative cyclic groups G and G_T with prime order q, where g is a generator of G. e: $G \times G \longrightarrow G_T$ denotes a bilinear map. After that, the KGC picks an integer $a \in_R Z_q^*$ at random and computes $g_1 = g^a$ where $g \in G$.

Next, $v_0, v_1, \cdots, v_\ell, u_1, \cdots, u_s \in_R G$ are uniformly chosen at random. Four collision-resistant hash functions are chosen as follows: $H_1, H_2, H_4 : \{0,1\}^* \to G$ and $H_3 : \{0,1\}^* \to \{0,1\}^\ell$. So, the system public parameter is $PP = (g, g_1, g_2, v_0, v_1, \cdots, v_\ell, u_1, \cdots, u_s \in_R G, H_1, H_2, H_3, H_4)$. Finally, the master secret key msk is set as $msk = g_2^a$ with $g_2 \in G$ and keeps the msk in secret by the KGC.

5.2 Registration: *KeyGen*

The KGC runs the *KeyGen* algorithm to yield a shared secret key for users with the msk and public parameter PP. The registration procedure consists of two phases: *PrivacyKeyGen* and *SecretKeyGen*.

(1) *PrivacyKeyGen*: First, the KGC generates and distributes the corresponding private key for every user who may be a patient or a consultant in e-healthy system. In detail, the KGC computes Q_i based on user's identity as $Q_i = H_1(ID_i)$. Then, KGC calculates user privacy key as:

$$d_i = g_2^a \cdot H_1(ID_i) \quad (1)$$

For example, KGC independently yields a private key d_A for patient A, and a private key d_B for the attending physician B. Then, the KGC sends d_i to ID_i. After receiving the d_i, user validates ID_i by calculating:

$$e(d_i, g) \stackrel{?}{=} e(g_2, g_1) \cdot e(H_1(ID_i), g) \quad (2)$$

If the above equation is true, the user ID_i adopts the private key d_i; otherwise, the KGC fails to generate a valid privacy key.

(2) *SecretKeyGen*: To protect the identity of patient A, patient A randomly chooses a number $r_A \in_R Z_q^*$, creates a pseudonym $P_A = r_A \cdot Q_A$, and sends it instead of his or her actual identity to B. Then, A and B can calculate a session key K_{AB}, and this algorithm produces a secret key KAB for auditing. The specific algorithm is as follows:

$$\begin{aligned} K_{AB} &= e(d_A, Q_B) = e(P_A, d_B) \\ KAB &= g_2^a \cdot H_2(K_{AB}) \end{aligned} \quad (3)$$

5.3 Storage: *Extract*

The storage procedure contains the following three phases: *WarrantGen*, *AuthenticatorGen*, and *TagGen*.

(1) *WarrantGen*: When user uploads or updates a new medical data, the corresponding file information will be updated. For confirming some additional information about the source, type, and consistency of the files

outsourced to the cloud, the user generates a warrant Λ which includes the pseudonym of A, the identity hash value Q_i of attending physician B, and medical file information such as file type *filetype*, version number V_N, time stamp T_N, etc. For example, $\Lambda = P_A \,||Q_B\,||V_N\,||T_N\,||filetype$. Here, the N denotes the index of different medical files. Then, the following is calculated:

$$\vec{\Lambda} = (\zeta_1, \cdots, \zeta_\ell) \leftarrow H_3(\Lambda) \tag{4}$$

The patient A picks a random number $t_\Lambda \in_R Z_q^*$, and generates an authorization:

$$\delta_\Lambda = (KAB \cdot (v_0 \cdot \prod_{j=1}^{\ell} v_j^{\zeta_j})^{t_\Lambda}, g^{t_\Lambda}) \tag{5}$$

Finally, the patient A sends the warrant pair $(\Lambda, \delta_\Lambda) = (\Lambda, (\alpha, \beta))$ to attending physician B. Then, the attending physician B validates the warrant pair by calculating:

$$e(\alpha, g) \stackrel{?}{=} e(g_2, g_1) \cdot e(H_2(K_{AB}), g) \cdot e(v_0 \prod_{j=1}^{\ell} v_j^{\zeta_j}, \beta) \tag{6}$$

If the above equation is true, the attending physician B accepts the authorization δ_Λ; otherwise, the patient A fails to generate a valid warrant.

(2) *AuthenticatorGen*: Given a medical file F to be outsourced, the user first splits F into n blocks, and each contains s sectors: $F \rightarrow \{\chi_{i,j}\}_{n \times s}$, where $\chi_{i,j} \in_R Z_q^*$. For each file F, choose a random number $t_\vartheta \in_R Z_q^*$, and for the i-th block, yield a block authenticator as follows:

$$\sigma_i = KAB \cdot (H_4(\Lambda \,||FID\,||i) \cdot \prod_{j=1}^{s} u_j^{\chi_{i,j}})^{t_\vartheta} \tag{7}$$

(3) *TagGen*: A random name *FID* is chosen for a file from Z_q^*, and s random elements $u_1, \cdots, u_s \in G$. Set $\tau_0 = \Lambda \,||FID\,||n\,||u_1\,||\,...\,||u_s\,||g^{t_\Lambda}\,||g^{t_\vartheta}$. Then, the user generates file tag τ based on τ_0 and K_{AB} to guarantee the integrity of each distinct file information.

$$\tau = \tau_0 \,||S.Sign(\tau_0)_{K_{AB}} \tag{8}$$

Hereafter, the user sends the file tag τ to the TPA. Besides, $KP = e(H_2(K_{AB}), g)$ can be pre-computed and sent to TPA. In addition, the processed file F^* that comprises F, FID, Λ, δ_Λ, and σ_i is uploaded to the CS and can be stored in the proposed stereo storage structure and removed from the user's local side.

5.4 Integrity verification: *Audit*

The auditing procedure contains following three phases: *Challenge*, *Response,* and *Verification*. And the process of integrity verification is shown in Figure 3.

(1) *Challenge*: First, the TPA confirms whether the file tag τ of outsourced file can pass the verification by retrieving τ from the CS and performing $S.Vrf(\tau_0, K_{AB})$. If the file tag τ of outsourced file cannot pass the verification, then the auditing task will not be executed, and the protocol aborts; otherwise, the TPA will analyze τ_0 to acquire the total number n of outsourced file blocks. The TPA picks a random nonempty subset $I \subseteq [1, n]$ and a number of values $s_i \in_R Z_q^*$ at random, for each $i \in I$. Then, the TPA distributes the challenge set $C = \{(i, s_i)_{i \in I}\}$ and corresponding file identifier FID to the CS. After that, the TPA can compute $WP = e(H_4(\Lambda \,||\, FID\,||i), g^{t_\vartheta})^{\sum_{i \in I} s_i}$ in advance for the final verification.

(2) *Response*: CS locates to the corresponding file F^* in the stereo storage structure upon receiving a challenge C and its file identifier FID from the TPA. Then, the CS computes $\chi_j = \sum_{i \in I} s_i \cdot \chi_{i,j} \bmod q, j \in [1, n]$ and $\sigma = \prod_{i \in I} \sigma_i^{s_i}$. After that, the CS sends to the TPA a proof P that consists of $\chi_1, \cdots, \chi_s, \sigma$ and corresponding authorization δ_Λ.

(3) *Verification*: Once receiving the proof P, with public system parameter PP and file tag τ, the TPA first verifies the validity of δ_Λ by demonstrating the equation (6), and then, verifies aggregate block authenticator σ as follows:

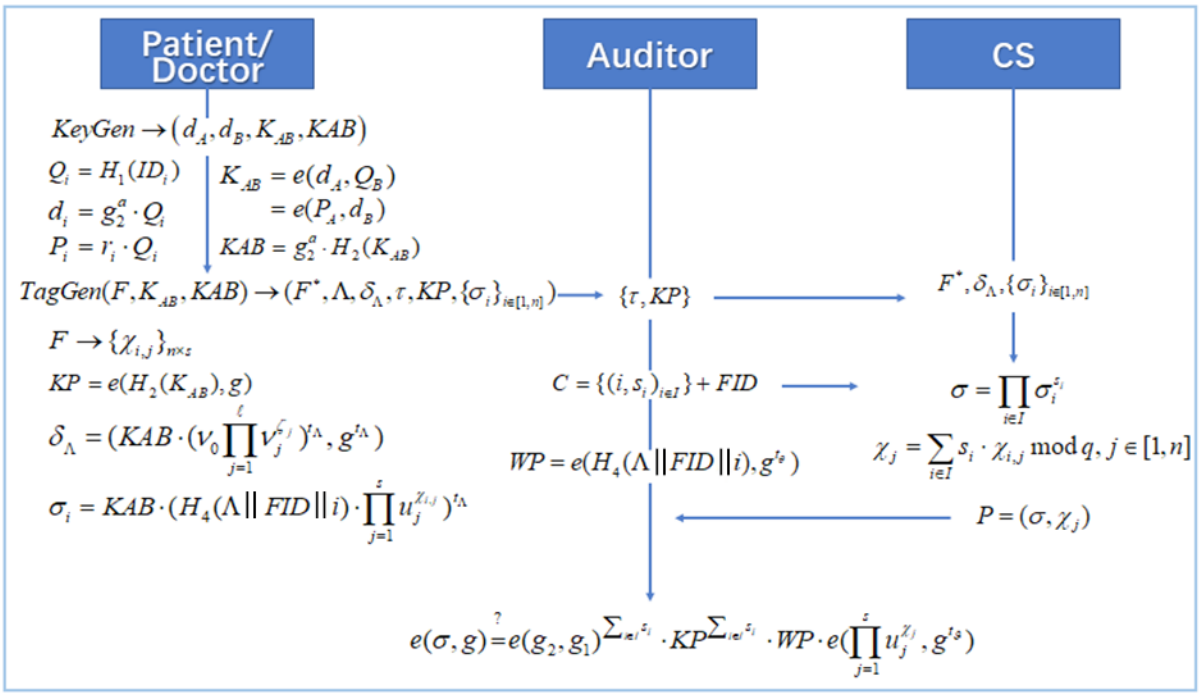


Figure 3. The process of integrity verification

$$e(\sigma, g) \stackrel{?}{=} e(g_2, g_1)^{\sum_{i\in I} s_i} \cdot KP^{\sum_{i\in I} s_i} \cdot WP \cdot e(\prod_{j=1}^{s} u_j^{\chi_j}, g^{t_\theta}) \quad (9)$$

If the equation (9) is true, the challenged outsourced file in the cloud is verified as intact; otherwise, the challenged file is corrupted. In the above auditing process, TPA can also audit the details of the challenged file warrant. That is, the proof *P*, which will be fed back by CS, should contain more file details.

6 SECURITY ANALYSIS

We analyzed the soundness of our scheme at first. That is, if all the entities are honest in this identity-based one-way anonymous e-health system, then the processed files and log warrants about medical data can be audited correctly. Then, we propose a simple security analysis for this scheme.

Theorem 1: In an appropriate registration process, the KGC is supposed to generate a correct privacy key for the user. In addition, the patient always produces a valid log warrant for his or her attending physician to render certain the authenticity of medical data. If the outsourced file in the cloud is not corrupted or tampered with, then the proof yielded by CS will be confirmed as valid.

Proof: As shown in Equation (2), we can confirm the correctness directly. Since patient A and the attending physician B have the shared auditing key, it follows that:

$$\begin{aligned} e(\alpha, g) &= e(KAB \cdot (v_0 \prod_{j=1}^{\ell} v_j^{\zeta_j})^{t_\Lambda}, g) \\ &= e(KAB, g) \cdot e((v_0 \prod_{j=1}^{\ell} v_j^{\zeta_j})^{t_\Lambda}, g) \\ &= e(g_2^a \cdot H_2(K_{AB}), g) \cdot e(v_0 \prod_{j=1}^{\ell} v_j^{\zeta_j}, g^{t_\Lambda}) \\ &= e(g_2, g_1) \cdot e(H_2(K_{AB}), g) \cdot e(v_0 \prod_{j=1}^{\ell} v_j^{\zeta_j}, g^{t_\Lambda}) \\ &= e(g_2, g_1) \cdot KP \cdot e(v_0 \prod_{j=1}^{\ell} v_j^{\zeta_j}, g^{t_\Lambda}) \end{aligned}$$

Therefore, Equation (6) holds.

Note that, $\chi_j = \sum_{i \in I} s_i \cdot \chi_{i,j} \bmod q$ for all $j \in [1, s]$ and

$$\begin{aligned}\sigma &= \prod_{i \in I} \sigma_i^{s_i} \\ &= \prod_{i \in I} KAB^{s_i} \cdot \prod_{i \in I} ((H_4(\Lambda \,||\, FID \,||i) \cdot \prod_{j=1}^{s} u_j^{\chi_{i,j}})^{t_\vartheta})^{s_i} \\ &= (g_2^a)^{\sum_{i \in I} s_i} \cdot H_2(K_{AB})^{\sum_{i \in I} s_i} \cdot (\prod_{i \in I} H_4(\Lambda \,||\, FID \,||i)^{s_i} \cdot \prod_{j=1}^{s} u_j^{\sum_{i \in I} s_i \chi_{i,j}})^{t_\vartheta}\end{aligned}$$

It follows that:

$$\begin{aligned}e(\sigma, g) &= e(g_2^a, g)^{\sum_{i \in I} s_i} \cdot e(H_2(K_{AB}), g)^{\sum_{i \in I} s_i} \cdot e((\prod_{i \in I} H_4(\Lambda \,||\, FID \,||i)^{s_i} \cdot \prod_{j=1}^{s} u_j^{\sum_{i \in I} s_i \chi_{i,j}}), g)^{t_\vartheta} \\ &= e(g_2, g_1)^{\sum_{i \in I} s_i} \cdot KP^{\sum_{i \in I} s_i} \cdot e(H_4(\Lambda \,||\, FID \,||i)^{s_i}, g^{t_\vartheta})^{\sum_{i \in I} s_i} \cdot e(\prod_{j=1}^{s} u_j^{\chi_j}, g^{t_\vartheta}) \\ &= e(g_2, g_1)^{\sum_{i \in I} s_i} \cdot KP^{\sum_{i \in I} s_i} \cdot WP \cdot e(\prod_{j=1}^{s} u_j^{\chi_j}, g^{t_\vartheta})\end{aligned}$$

Theorem 2: Here, we suppose that the signature algorithm is efficient and secure, and can generate file tags validly and correctly. And it is supposed that the Computational Diffie-Hellman (CDH) assumption holds in bilinear groups. The identity-based one-way anonymous scheme is secure against adaptive simulation. In detail, neither an untrusted cloud server nor the adversary $\mathcal{A}$ can forge a valid proof to get through the verification of the auditor successfully if the data in the cloud is tampered with or corrupted.

Proof: We utilize the theory of knowledge proof and a series of security games to prove this theorem which can acquire the challenged data blocks in the aforementioned game. When the adversary $\mathcal{A}$ interacts with the challenger $\mathcal{C}$ and generates a valid proof *P*, adversary $\mathcal{A}$ can successfully pass the verification for the challenged data blocks in the aforementioned game; there is a constructed knowledge extractor that can capture the challenged data blocks. It is assumed that the adversary $\mathcal{A}$ can get through the TPA's verification successfully without keeping the outsourced file integrity. Then, we can capture the whole challenged data blocks through the interaction between the constructed knowledge extractor and the proposed scheme.

Game 0: The challenger $\mathcal{C}$ and the adversary $\mathcal{A}$ behave in Game 0 in a manner similar to that described in Section 4. First, the challenger $\mathcal{C}$ executes the preprocessing Setup algorithm to obtain the public parameter *PP* and a master secret key *msk*, and then sends *PP* to the adversary $\mathcal{A}$. Next, $\mathcal{C}$ performs the *KeyGen* algorithm to obtain the secret key of user. Then, $\mathcal{A}$ picks a list of data blocks and queries the signatures of them. According to the queries, $\mathcal{C}$ executes the Extract algorithm to generate corresponding signatures for the data blocks and transmit these requested signatures to the $\mathcal{A}$. After that, $\mathcal{C}$ sends a challenge to $\mathcal{A}$, and $\mathcal{A}$ generates corresponding proof to $\mathcal{C}$. Finally, $\mathcal{A}$ succeeds and the game aborts if the proof can get through the verification of $\mathcal{C}$ successfully with non-negligible probability.

Game 1: This game is identical to Game 0 with one difference. The challenger $\mathcal{C}$ keeps a list of query records about the requested signature of $\mathcal{A}$. If the adversary $\mathcal{A}$ is able to yield a aggregate signature, which is valid under the verification of the challenger $\mathcal{C}$ and is not generate by $\mathcal{C}$, the game aborts and the adversary $\mathcal{A}$ succeeds.

Analysis: It is supposed that $\mathcal{A}$ wins in the Game 1 with non-negligible probability. With this in mind, we can construct a simulator in our scheme to solve the CDH problem in bilinear groups. Given a group *G* with prime order q, $g, g^\alpha, h \in G$ as input, the simulator is to generate h^α by interacting with $\mathcal{A}$. The simulator acts like the challenger and runs as follows:

(1) The simulator randomly chooses an element $x \in_R Z_q^*$, and yields the public parameters as $g_1 = g^x$, $g_2 = h$ and the master secret key $msk = g_2^a$. Next, it randomly picks integers $\omega_j, \varpi_j \in_R Z_q^*$, and sets $u_j = g_2^{\omega_j} g^{\varpi_j}$. There is a random oracle H_4. The simulator stores a list of queries in the game and responses to the challenger $\mathcal{C}$ in a consistent manner by controlling the random oracle.

(2) When processing a file *F*, the simulator first yields a secret key for user as *KAB* by executing KeyGen algorithm. Hereafter, the simulator picks a random unique identifier for file *F* and a random element $\tilde{x} \in Z_q^*$, and yields $g^{t_\vartheta} = (g^\alpha)^{\tilde{x}}$. For every data block i, the simulator picks random values $\chi_i \in_R Z_q^*$ and sets:

$$H_4(\Lambda \,||\, FID \,||i) = g^{\gamma_i} / (g_2^{\sum_{j=1}^{s} \omega_j \chi_{i,j}} g^{\sum_{j=1}^{s} \varpi_j \chi_{i,j}}) \quad (10)$$

Based on equation (10), we have:

$$(H_4(\Lambda \,||\, FID \,||i) \cdot \prod_{j=1}^{s} u_j^{\chi_{ij}})^{t_\vartheta} = (g^{\gamma_i})^{t_\vartheta} \quad (11)$$

In addition, the simulator computes the block authentication for file block x_i as $\sigma_i = KAB \cdot (H_4(\Lambda \,||\, FID \,||i) \cdot \prod_{j=1}^{s} u_j^{\chi_{i,j}})^{t_\vartheta} = KAB \cdot (g^{\gamma_i})^{t_\vartheta}$. From the perspective of $\mathcal{A}$, σ_i is computationally indistinguishable from the real value.

(3) With the constant interaction, the simulator sends the processed files F^* to the adversary $\mathcal{A}$, which contains $\{\{\sigma_i\}_{i\in[1,n]}, \delta_\Lambda, FID\}$. Then, $\mathcal{A}$ outputs a forgery $\tilde{\sigma}$ with a non-negligible probability. Finally, if the adversary $\mathcal{A}$ is succeed to pass the validation, but the aggregate authentication $\tilde{\sigma}$ is unequal to the excepted aggregate authentication σ calculated by the simulator, then the game aborts.

According to the correctness of the proposed protocol, it is obvious that a correct proof $\chi_1, \cdots, \chi_s, \sigma$ can get through the verification successfully of the equation as follow:

$$e(\sigma, g) = e(g_2, g_1)^{\sum_{i\in I} s_i} \cdot KP^{\sum_{i\in I} s_i} \cdot WP \cdot e(\prod_{j=1}^{s} u_j^{\chi_j}, g^{t_\vartheta}) \quad (12)$$

Suppose the adversary $\mathcal{A}$ forges a proof $\tilde{\chi}_1, \cdots, \tilde{\chi}_s, \tilde{\sigma}$ which is different from the correct proof. Next, compute the following equation:

$$e(\tilde{\sigma}, g) = e(g_2, g_1)^{\sum_{i\in I} s_i} \cdot KP^{\sum_{i\in I} s_i} \cdot WP \cdot e(\prod_{j=1}^{s} u_j^{\tilde{\chi}_j}, g^{t_\vartheta}) \quad (13)$$

It is obvious that $\tilde{\chi}_j \neq \chi_j$, otherwise $\tilde{\sigma} = \sigma$. Then, define a set $\{\Delta\chi_j = \tilde{\chi}_j - \chi_j\}_{j\in[1,s]}$, which means at least one element of $\Delta\chi_j$ is non-zero. After that, divide equation (13) by equation (12) and get the following equation:

$$e(\tilde{\sigma}/\sigma, g) = e(\prod_{j=1}^{s} u_j^{\Delta\chi_j}, g^{t_\vartheta}) = e(\prod_{j=1}^{s} (g_2^{\omega_j} g^{\varpi_j})^{\Delta\chi_j}, (g^\alpha)^{\tilde{x}}) \quad (14)$$

It further implies:

$$e(\tilde{\sigma} \cdot \sigma^{-1} \cdot (g^\alpha)^{-\tilde{x} \cdot \sum_{j=1}^{s} \varpi_j \cdot \Delta\chi_j}, g) = e(h, g^\alpha)^{\tilde{x} \cdot \sum_{j=1}^{s} \omega_j \cdot \Delta\chi_j} \quad (15)$$

Finally, we can get the value of h^α as follow:

$$h^\alpha = (\tilde{\sigma} \cdot \sigma^{-1} \cdot (g^\alpha)^{-\tilde{x} \cdot \sum_{j=1}^{s} \varpi_j \cdot \Delta\chi_j})^{1/(\tilde{x} \cdot \sum_{j=1}^{s} \omega_j \cdot \Delta\chi_j)} \quad (16)$$

As long as the $\tilde{x} \cdot \sum_{j=1}^{s} \omega_j \cdot \Delta\chi_j \neq 0 \bmod q$, the above equations are valid and can be structured to solve the CDH problem. The probability of solving the CDH problem is equal to the probability of $1 - \Pr[\tilde{x} \cdot \sum_{j=1}^{s} \omega_j \cdot \Delta\chi_j = 0 \bmod q] = 1 - 1/q$, which is contradictory with the assumptions of the CDH problem. It means that if the adversary $\mathcal{A}$ has a different probability of success in Game 0 versus Game 1, which is non-negligible, then the simulator can be constructed to solve the CDH problem.

Game 2: Game 2 is similar with Game 1, except the following difference. The challenger $\mathcal{C}$ keeps interaction with the adversary $\mathcal{A}$ and holds all the processed outsourced files that have been sent to $\mathcal{A}$. In the process of the proposed auditing protocol, if the aggregate authenticator $\tilde{\sigma}$ yielded by $\mathcal{A}$ is not equality to the aggregate authenticator σ of the challenged file blocks, then the game aborts and the adversary $\mathcal{A}$ succeeds.

Analysis: Suppose the adversary $\mathcal{A}$ wins in this Game with a non-negligible probability. Hereafter, a simulator is constructed to work out the Discrete algorithm (DL) problem if the adversary $\mathcal{A}$ can succeed in this game. Given a group G with prime order q, $g, h \in G$ as input, the target of the simulator is to yield α by interacting with $\mathcal{A}$, which satisfies $h = g^{\alpha}$. The simulator behaves like $\mathcal{C}$ in Game 2, but with the following differences:

(1) Before processing a file F, the simulator first performs the *KeyGen* algorithm and yields a secret key for user as *KAB*. Then, following the process of the presented scheme in this paper, the simulator uses $u_j = g_2^{\omega_j} g^{\varpi_j}$ for each $1 \le j \le s$, where $\omega_j, \varpi_j \in_R Z_q^*$.

(2) The simulator keeps interacting with $\mathcal{A}$ to execute the auditing protocol proposed in this paper. As described in Game 1, if the aggregate file sectors $\tilde{\chi}_j$ generated by the adversary $\mathcal{A}$ is not equal to the aggregate file sectors χ_j of the challenged sectors, then the game aborts and the adversary $\mathcal{A}$ succeeds. It is easy to know that $\tilde{\sigma} = \sigma$ for the reason that Game 1 is not aborted. Next, with this in mind, compared with equation (12) and equation (13), we can get the following equation:

$$e(\prod_{j=1}^{s} u_j^{\tilde{\chi}_j}, g^{t_g}) = e(\prod_{j=1}^{s} u_j^{\chi_j}, g^{t_g}) \tag{17}$$

It further indicates that:

$$\prod_{j=1}^{s} u_j^{\tilde{\chi}_j} = \prod_{j=1}^{s} u_j^{\chi_j} \tag{18}$$

In addition, set $\{\Delta\chi_j = \tilde{\chi}_j - \chi_j\}_{j\in[1,s]}$, which means at least one element of $\Delta\chi_j$ is non-zero. After that, compute:

$$\prod_{j=1}^{s} u_j^{\Delta\chi_j} = h^{\sum_{j=1}^{s}\omega_j\Delta\chi_j} g^{\sum_{j=1}^{s}\varpi_j\Delta\chi_j} = 1 \tag{19}$$

Finally, the value of a is as follow:

$$\alpha = -\frac{\sum_{j=1}^{s}\varpi_j\Delta\chi_j}{\sum_{j=1}^{s}\omega_j\Delta\chi_j} \bmod q \tag{20}$$

As long as $\sum_{j=1}^{s}\omega_j\Delta\chi_j \neq 0 \bmod q$, the above equations are valid and can be structured to work out the DL problem. The probability of solving the DL problem is the same as the probability of $1 - \Pr[\sum_{j=1}^{s}\omega_j\Delta\chi_j \neq 0 \bmod q] = 1 - 1/q$, which is contradictory with the assumption of the DL problem. It means that if the adversary $\mathcal{A}$ has a different probability of success in Game 1 and Game 2, which is non-negligible, then the simulator can be constructed to solve the DL problem. To summarize, the proposed one-way anonymous auditing protocol is secure and can be proven by uniting Game 0, Game 1, and Game 2.

7 PERFORMANCE ANALYSIS

In this section, we first compare our scheme with the related schemes in terms of various characteristics. In Table 2, we can clearly conclude that our solution can better satisfy all the major characteristics.

Then, we give the numerical analysis of the computation overhead of the proposed stereo storage structure assisted one-way anonymous auditing protocol and then evaluate the performance of our scheme. In Table 3, we analyze and present the computation overhead of each algorithm respectively in the proposed scheme. Primarily, the following notations are defined to represent the various operations in the specific algorithms of each phase. The symbols $\mathbb{M}$, $\mathbb{E}$, and $\mathbb{H}$ denote a multiplication operation, a exponentiation operation and a hashing operation in G, respectively. In this paper, H_1, H_2, and H_3 are not distinguished and all can be expressed as $\mathbb{H}$. Similarly, the symbols $\mathbb{M}_T$ and $\mathbb{E}_T$ are respectively expressed as a multiplication operation and a exponentiation operation in G_T. $\mathbb{A}_q$ and $\mathbb{M}_q$ are indicated as one addition operation and one multiplication operation in Z_q, respectively. And $\mathbb{P}$ represents a bilinear pairing evaluation operation $e: G \times G \to G_T$. Considering that both g_1 and g_2 are public system parameters in our protocol, then $e(g_2, g_1)$ can be calculated in advance and viewed as a public value.

Table 2. Characteristics comparison with related schemes

Schemes	Public verifiability	Certificate management simplification	Privacy protection	Dynamic operations
Worku *et al*.[17]	√	×	√	√
Garg *et al*.[19]	√	×	×	√
Daniel and Vasanthi[25]	√	×	√	×
Zhao *et al*.[27]	×	√	×	×
Jiang *et al*. (this study)	√	√	√	√

Table 3. Computational overhead of the proposed scheme

Phases	KGC	User (physician)	User (patient)	TPA	CS
Setup	$2\mathbb{E}$	/	/	/	/
KeyGen(a)	$\mathbb{M}+\mathbb{H}$	$2\mathbb{P}+\mathbb{H}+\mathbb{M}_T$	$2\mathbb{P}+\mathbb{H}+\mathbb{M}_T$	/	/
KeyGen(b)	/	$\mathbb{P}+\mathbb{H}+\mathbb{M}$	$\mathbb{P}+2\mathbb{H}+\mathbb{M}_q+\mathbb{M}$	/	/
Extract(a)	/	$3\mathbb{P}+2\mathbb{H}+2\mathbb{M}_T+\ell\mathbb{M}$	$2\mathbb{E}+\mathbb{H}+(\ell+1)\mathbb{M}$	/	/
Extract(b)	/	$\mathbb{E}+\mathbb{H}+(s+1)\mathbb{M}$	/	/	/
Audit(b)	/	/	/	/	$n\lvert I\rvert\mathbb{M}_q+n(\lvert I\rvert-1)\mathbb{A}_q+(\lvert I\rvert-1)\mathbb{M}+\lvert I\rvert\mathbb{E}$
Audit(c)	/	/	/	$(s+1)\mathbb{E}+\mathbb{H}+(\lvert I\rvert-1)\mathbb{A}_q+3\mathbb{P}+3\mathbb{E}_T+(s+1)\mathbb{M}+\mathbb{M}_T$	/

KGC: Key generation center; TPA: third-party auditor; CS: cloud server

Therefore, the computation overhead of $e(g_2,g_1)$ is not contained in Table 3. Furthermore, the symbols $S.Sign$ and $S.Vrf$ are used to denote the signature and verification file tag processes. Hereafter, as shown in Table 3, *Setup* is a system preprocessing phase, which is performed by KGC and needs $2\mathbb{E}$. In the algorithm of *KeyGen(a)*, KGC needs $\mathbb{M}+\mathbb{H}$ operations to generate a privacy key for user, and both the physician and the patient need $2\mathbb{P}+\mathbb{H}+\mathbb{M}_T$ operations to verify the validity of the private key distributed by KGC. In the algorithm of *KeyGen(b)*, the patient performs one $\mathbb{H}$ operation and one $\mathbb{M}_q$ operation more than the physician to generate a pseudonym. To process a medical file, patient firstly yields a warrant for the physician, which needs $2\mathbb{E}+\mathbb{H}+(\ell+1)\mathbb{M}$ operations. Then, the physician verifies the validity of the warrant, which needs $3\mathbb{P}+2\mathbb{H}+2\mathbb{M}_T+\ell\mathbb{M}$ operations. ℓ denotes the string length of warrant. The amount of file data blocks and sectors are expressed as n and s. After that, physician performs another $\mathbb{E}+\mathbb{H}+(s+1)\mathbb{M}$ operation to generate a block authenticator. After receiving a challenge from TPA, CS executes $n\lvert I\rvert\mathbb{M}_q+n(\lvert I\rvert-1)\mathbb{A}_q+(\lvert I\rvert-1)\mathbb{M}+\lvert I\rvert\mathbb{E}$ operations to yield a proof P, where the $\lvert I\rvert$ is indicated as a set of non-empty challenge file randomly selected by TPA for auditing. Finally, TPA performs $(s+1)\mathbb{E}+\mathbb{H}+(\lvert I\rvert-1)\mathbb{A}_q+3\mathbb{P}+3\mathbb{E}_T+(s+1)\mathbb{M}+\mathbb{M}_T$ operations to verify data integrity in the cloud.

Figure 4 shows the computational cost of each entity in the proposed scheme for auditing an outsourced medical file with various numbers of data blocks. In this scheme, the time costs of TPA to prepare a challenge $\lvert I\rvert$ is not taken into account, for TPA can sample a series of random elements by running offline. In the experiments, we set $\ell=160$ in this scheme and each file block consists of 160 sectors, which means that it has around 4 KB of size. Moreover, we compare the efficiency of processing a 1 MB file by set challenge data block as 20, 40, ... , 100, 200, respectively.

The simulation results of Figure 4 demonstrate that the computational cost of the user is independent of the number of data blocks in the file in carrying out the extraction algorithm. Specifically, this experiment of our scheme only considers the case that patients generate warrants for files, which can be verified by physicians and generate file tags for those files, so the calculation cost of physicians is slightly higher than that of patients, which is in line with the theoretical computational overhead analysis of the proposed scheme shown in Table 3. In addition, if it is necessary, the division of work between the physician and the patient is interchangeable during the file processing phase. After that, in the audit phase, TPA has transferred part of the calculate task to CS. Therefore, we can conclude that, as shown in Figure 4, with the increase of data blocks, the calculation cost of CS increases gradually.

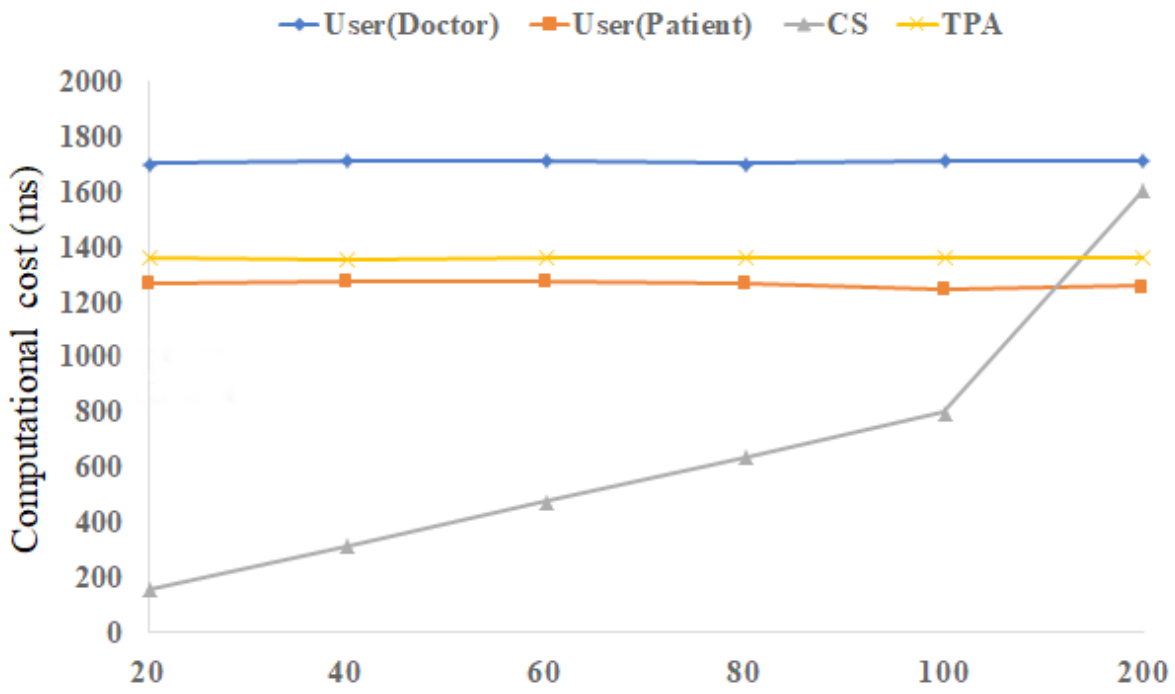


Figure 4. The computational cost of each entity in the proposed scheme

8 CONCLUSION

In this paper, we proposed a stereo storage structure assisted one-way anonymous auditing protocol aiming the e-health system for the particularity of medical data. In our scheme, medical data can be reviewed, used and verified for integrity by relevant medical personnel and relevant patients. Besides, both the file origin and the file integrity of medical data in EHS can be verified. In addition, the proposed stereo storage structure can effectively assist the storage and quick search of various types of medical data. Both the security analyses and experimental results demonstrate that the proposed scheme in this paper is efficient and secure in the cloud.

DECLARATIONS

Authors' contributions

Made substantial contributions to conception and design of the study and write the manuscript: Jiang LH
Provided administrative, technical, and material support: Wang C, Shen J

Availability of data and materials

The related data used to support the findings of this study are included within the article.

Financial support and sponsorship

This work is supported by the National Natural Science Foundation of China (No. U1836115, No. 61672295, No. 61922045, No. 61672290), the Natural Science Foundation of Jiangsu Province (No. BK20181408), Henan Key Laboratory of Network Cryptography Technology (No. LNCT2019-A01), the Peng Cheng Laboratory Project of Guangdong Province (No. PCL2018KP004), the 2020 Research Innovation Program for Postgraduates of Jiangsu Province (No. KYCX20-0936), the CICAEET fund, and the PAPD fund.

Conflicts of interest

All authors declared that there are no conflicts of interest.

Ethical approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Copyright

REFERENCES

1. Qian L, Luo ZG, Du YJ, Guo LT. Cloud computing: an overview. Proceedings of the IEEE International Conference on Cloud Computing; 2009 Sep 21-25; Bangalore, India. Springer; 2009. pp. 626-31.
2. Ion I, Sachdeva N, Kumaraguru P, Čapkun S. Home is safer than the cloud!: privacy concerns for consumer cloud storage. Proceedings of the Seventh Symposium on Usable Privacy and Security; 2011 Jul 14-16; Pittsburgh, PA, USA. ACM; 2011. pp. 1-20.
3. Yu Y, Au MH, Ateniese G, Huang XY, Susilo W, et al. Identity-based remote data integrity checking with perfect data privacy preserving for cloud storage. IEEE T Inf Foren Sec 2016;12:767-78.
4. Kang B, Wang J, Shao D. Attack on privacy-preserving public auditing schemes for cloud storage. Math Probl Eng 2017;2017:8062182.
5. Li Y, Yu Y, Yang B, Min G, Wu H. Privacy preserving cloud data auditing with efficient key update. Future Gener Comp Sy 2018;78:789-98.
6. Mehmood A, Natgunanathan I, Xiang Y, Hua G, Guo S. Protection of big data privacy. IEEE access 2016;4:1821-34.
7. More S, Chaudhari S. Third party public auditing scheme for cloud storage. Procedia Computer Science 2016;79:69-76.
8. Shen W, Yu J, Xia H, Zhang H, Lu X, et al. Light-weight and privacy-preserving secure cloud auditing scheme for group users via the third party medium. J Netw Comp Appl 2017;82:56-64.
9. Wang H, He D, Yu J, Wang ZW. Incentive and unconditionally anonymous identity-based public provable data possession. IEEE T Serv Comput 2019;12:824-35.
10. Balasubramanian V, Mala T. Cloud data integrity checking using bilinear pairing and network coding. Cluster Comput 2019;22:6927-35.
11. Yang G, Yu J, Shen W, Su Q, Fu Z, et al. Enabling public auditing for shared data in cloud storage supporting identity privacy and traceability. J Syst Software 2016;113:130-9.
12. Deswarte Y, Quisquater JJ, Sadane A. Remote integrity checking. In: Working Conference on Integrity and Internal Control in Information Systems. Springer; 2003. pp. 1-11.
13. Ateniese G, Burns R, Curtmola R, Herring J, Kissner L, et al. Provable data possession at untrusted stores. Proceedings of the 14th ACM conference on Computer and communications security; 2007 Oct 29- Nov 2; Alexandria, Virginia, USA. ACM; 2007. pp. 598-609.
14. Juels A, Kaliski BS. PORs: proofs of retrievability for large files. Proceedings of the 14th ACM conference on Computer and communications security; 2007 Oct 29- Nov 2; Alexandria, Virginia, USA. ACM; 2007. pp. 84-97.
15. Wang C, Wang Q, Ren K, Lou WJ. Privacy-preserving public auditing for data storage security in cloud computing. Proceedings of the 29th IEEE Conference on Computer Communications; 2010 Mar 15-19; San Diego, CA, USA. IEEE ComSoc; 2010. pp. 1-9.
16. Wang Q, Wang C, Li J, Ren K, Lou W. Enabling public verifiability and data dynamics for storage security in cloud computing. Proceedings of the 14th European Symposium on Research in Computer Security; 2009 Sep 21-23; Saint-Malo, France. Springer; 2009. pp. 355-70.
17. Worku SG, Xu C, Zhao J, He X. Secure and efficient privacy-preserving public auditing scheme for cloud storage. Comput Electr Eng 2014;40:1703-13.
18. Wang C, Chow SSM, Wang Q, Ren K, Lou W. Privacy-preserving public auditing for secure cloud storage. IEEE T Comput 2013;62:362-75.
19. Garg N, Bawa S, Kumar N. An efficient data integrity auditing protocol for cloud computing. Future Gener Comput Syst 2020;109:306-16.
20. Yuan J, Yu S. Public integrity auditing for dynamic data sharing with multiuser modification. IEEE Trans Inform Forensic Secur 2015;10:1717-26.
21. Suguna M, Mercy Shalinie S, Sivaranjani R. Integrity verification for shared data in group with user revocation. In: Zungeru AM, Subashini S, Vetrivelan P, editors. Wireless Communication Networks and Internet of Things. Singapore: Springer; 2019. pp. 41-9.
22. Wang X, Weng J, Ma J, Yang X. Cryptanalysis of a public authentication protocol for outsourced databases with multi-user modification. Inform Sciences 2019;488:13-8.
23. Zhang Y, Yu J, Hao R, Wang C, Kui R. Enabling efficient user revocation in identity-based cloud storage auditing for shared big data. IEEE T Depend Secure 2020;17:608-19.
24. Wu Y, Jiang ZL, Wang X, Yiu SM, Zhang P. Dynamic data operations with deduplication in privacy-preserving public auditing for secure cloud storage. Proceedings of the IEEE International Conference on Computational Science and Engineering and IEEE International Conference on Embedded and Ubiquitous Computing; 2017 Jul 21-24; Guangzhou, Guangdong, China. IEEE; 2017. pp. 562-7.
25. Daniel E, Vasanthi NA. LDAP: a lightweight deduplication and auditing protocol for secure data storage in cloud environment. Cluster Comput 2019;22:1247-58.
26. Tang X, Huang Y, Chang C, Zhou L. Efficient real-time integrity auditing With privacy-preserving arbitration for images in cloud storage

system. IEEE Access 2019;7:33009-23.
27. Zhao C, Xu L, Li J, Wang F, Fang H. Fuzzy identity-based dynamic auditing of big data on cloud storage. IEEE Access 2019;7:160459-71.
28. Barua R, Dutta R, Sarkar P. Extending Joux's protocol to multi party key agreement. In: Johansson T, Maitra S, editors. Progress in Cryptology - INDOCRYPT 2003. Berlin: Springer Berlin Heidelberg; 2003. pp. 205-17.
29. Kate A, Zaverucha G, Goldberg I. Pairing-based onion routing. In: Borisov N, Golle P, editors. Privacy Enhancing Technologies. Berlin: Springer Berlin Heidelberg; 2007. pp. 95-112.

Salmani *et al. J Surveill Secur Saf* 2020;1:79–101
DOI: 10.20517/jsss.2020.16

Journal of Surveillance, Security and Safety

Original Article

Open Access

Leakless privacy-preserving multi-keyword ranked search over encrypted cloud data

Khosro Salmani[1], Ken Barker[2]

[1]Department of Mathematics and Computing, Mount Royal University, Calgary, AB T3E 6K6, Canada.
[2]Department of Computer Science, University of Calgary, Calgary, AB T2N 1N4, Canada.

Correspondence to: Prof. Ken Barker, Department of Computer Science, University of Calgary, Calgary, AB T2N 1N4, Canada. E-mail: kbarker@ucalgary.ca

How to cite this article: Salmani K, Barker K. Leakless privacy-preserving multi-keyword ranked search over encrypted cloud data. *J Surveill Secur Saf* 2020;1:79–101. http://dx.doi.org/10.20517/jsss.2020.16

Received: 4 May 2020 **First Decision:** **Revised:** 28 Jue 2020 **Accepted:** 11 August 2020 **Available online:** 27 Sep 2020

Academic Editor: Xiaofeng Chen **Copy Editor:** Stella Zhang **Production Editor:** Jing Yu

Abstract

Aim: During the last decade, various type of cloud services have encouraged individuals and enterprises to store personal data in the cloud. Despite its flexibility, cost efficiency, and convenient service, protecting security and privacy of the outsourced data has always been a primary challenge. Although data encryption retains the outsourced data's security and privacy to some extent, it does not permit traditional plaintext keyword search mechanisms, and it comes at the cost of efficiency. Hence, proposing an efficient encrypted cloud data search service would be an important step forward. Related work focuses on single keyword search and even those which support multi-keyword search suffer from private information leakage.

Methods: Our proposed method, employs the secure inner product similarity and our chaining encryption notion. The former helps to provide sufficient search accuracy and the latter yields the privacy requirements.

Results: In this paper, we address the problem of leakless privacy-preserving multi-keyword ranked search over encrypted cloud data (LRSE), and our new contributions address challenging problems of search pattern, and co-occurrence information leakage in the cloud.

Conclusion: Our security and performance analysis shows that the proposed scheme guarantees a high level of privacy/security and efficiency.

Keywords: Data privacy, cloud security, multi-keyword ranked search, privacy-preserving data search

1 INTRODUCTION

With the advent of cloud computing, data owners are motivated to outsource their personal data from local repositories to the cloud to increase flexibility, convenience and to reduce the burden of the local data maintenance costs. However, the privacy of the data must be preserved against unwanted, unauthorized, and malicious accesses from outside attackers and unapproved insiders, including systems such as cloud servers. To access the data, unlike external attackers that must develop smart and subtle ways to circumvent security firewalls and access-control mechanisms, the cloud accesses data directly, which is the reason it is considered a primary privacy threat.

Encrypting the data before it is transferred to the cloud protects the data to some extent, but at the cost of efficiency. Posing queries on the encrypted data so it is searchable is one the current open challenges. Even though Searchable Symmetric Encryption (SSE) enables search on ciphertext, these schemes support only Boolean keyword search, i.e., whether a keyword exists in a document or not[1–7]. Moreover, these schemes must strike a balance between security and efficiency[8]. Consequently, the documents retrieved may be incorrect or incomplete which consumes more time, computation power, and network bandwidth. Conversely, disclosing more information to the cloud to increase accuracy and query efficiency, leads to further privacy exposure.

Multi-keyword ranked search over encrypted cloud data (MRSE) schemes have been proposed[9–11] to (1) acquire result relevance ranking, instead of returning undifferentiated results; and (2) improve search result accuracy by supporting multiple keywords search instead of single keyword search which often yields in unacceptably coarse results. Consequently, each keyword refines the results further. Moreover, the cloud is able to rank the results based on their relevance and returns the top-k most relevant data items which causes less network bandwidth consumption, increased data user's satisfaction, and is highly desirable in the "pay-as-you-use" cloud paradigm[9].

Although MRSE schemes[9–11] are helpful and allow a user to search and retrieve the documents of interest, they suffer from private information leakage. Query algorithms for existing MRSE schemes are mostly deterministic, which means the same keyword can be used for the same type of queries. Thus, the attacker is able to determine whether the keywords retrieved by two queries are the same[12] (search pattern attack). Moreover, some words often co-occur with other words so the attacker can determine keywords with similar term of distribution (co-occurrence attack)[11]. For instance, the bigram "of the" occurs much more frequently than any other bigrams in English language[13]; and possessing some auxiliary knowledge can disclose more information about the co-occurring terms. For example, the term "united" is very likely to co-occur with "states" in White House official paperwork. Thus, identifying a term is not difficult when the attacker knows the corresponding co-occurring term in the ciphertext. Further, documents in the same category share a considerable overlap of terms and keywords so information leakage in one document can lead to privacy violation in other documents. In addition, during each search, tracking the keywords and the corresponding retrieved documents can leak more information (access pattern attack). Thus, to preserve the data privacy, this information must be hidden from untrusted, unapproved, and unauthorized parties.

This paper tackles the problem of leakless privacy preserving multi-keyword ranked search over encrypted cloud data (LRSE), and we solve the problem of search pattern, and co-occurrence (for the first time among multi-keywords SSE schemes) private information leakage. To capture the relevance between documents and the search query, we employ secure inner-product similarity that provides sufficient search accuracy[14]. In our approach, the documents and the search queries are described as binary vectors where each bit represents the existence of the corresponding keyword in the document/search query. Thus, the similarity can be measured by the inner-product of the query and document vector[14]. The distribution of the keywords in the documents is not uniform and decreases uncertainty (entropy) of the accessed documents. To address this problem, the

key idea in our scheme is to exploit our chaining encryption notion to generate a variety of ciphertexts for high frequency keywords which leads to more uncertainty and a uniform probability model for the keywords distribution.

The contributions of this work are:

1. We explore the problem of leakless privacy preserving multi-keyword ranked search over encrypted cloud data. We build on a private model to prevent (without compromising efficiency):
   (a) Search pattern attack (tracking the keywords searched by two or more queries)
   (b) Co-occurrence attack (determining keywords with similar frequency)
2. Our methodological contribution is a novel chaining encryption notation which prevent the aforementioned attacks.
3. We demonstrate using privacy and security analysis the correctness of our proposed method.

To achieve our goals (see Section 2.1) there are two approaches. One possibility is to create an index which decreases the searches elapsed time. In the related literature two types of index are considered[4]: (1) building an index for each document D_i [15]; (2) design an index which encompasses the entire corpus[10,16]. The alternative approach is to perform a sequential scan without an index. When the documents are large, an index will likely be faster than sequential search, but on the flip side, storing and updating the index increases overhead considerably. Either approach would be appropriate here depending on the corpus's characteristics such as file length and file modification frequency.

We first describe the sequential search scheme using our novel chaining notion (see Section 4.1). Next we express our second scheme which benefits from an index for the whole corpus (see Section 4.2). Note that, in the second scheme we exploit the chaining notion idea in generating the index vectors even though it (chaining notion) is not employed to encrypt the documents.

1.1 System model

Our system model as illustrated in Figure 1, involves three different entities: data owner, data users, and cloud server. The data owner has a collection of documents $\mathcal{D}$ (files) to be outsourced to the cloud server. Since files may contain sensitive information and the cloud server is not fully trusted, data must be encrypted ($\mathcal{C}$); and any kind of information leakage that jeopardizes the data privacy is inadmissible. Moreover, for the sake of effective data utilization and to ensure precise results, the cloud server must apply the search requests (queries) on the encrypted data. Hence, before outsourcing the data onto the cloud, the data owner extracts a set of keywords Δ_d to build an encrypted searchable index $\mathcal{SI}$. We extract the keywords before encrypting the data, so the keywords with a high frequency get encrypted into a number of ciphertexts. As a result, it becomes significantly more difficult for the cloud to track specific keywords in documents as we will explain shortly. Both the encrypted index and encrypted data are then transferred to the cloud server.

To search for files of interest, an authorized user first acquires a key K from the data owner through a search control mechanism such as broadcast encryption[4]. Upon receiving the encrypted search request q from a data user, the cloud server applies the request on the corresponding index $\mathcal{SI}$ and returns the results $\mathcal{R}(q)$. To increase result precision, the results are ranked based on their relevance to the request by the cloud server. Furthermore, to reduce the communication cost, the data user may send an optional number k along with q, so the cloud server only sends back the top-k documents that are most relevant to the search request[14].

The rest of this paper is organized as follows: Section 2 presents our threat model, design goals, and the pre-

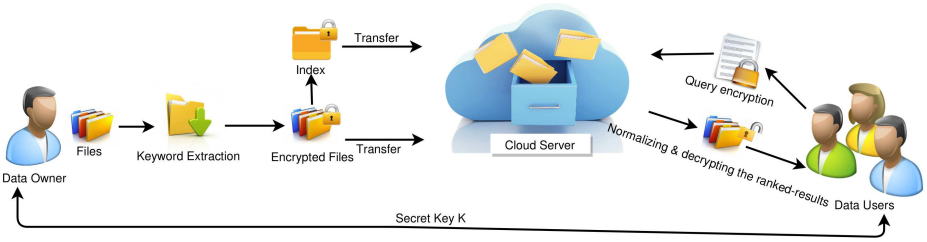


Figure 1. Architecture of the search over encrypted cloud data.

liminary. In Section 3, we describe the LRSE privacy requirements. Section 4 shows the proposed schemes in detail, followed by Section 5 which presents the privacy and security analysis. We summarize related works on privacy-preserving multi-keyword ranked search over encrypted cloud data in Section 6, and Section 7 summarizes our conclusions.

2 PROBLEM FORMULATION

2.1 Design goals

To address the aforementioned privacy issues (see Section 1), our design system should achieve privacy, security, and a high level of performance simultaneously with the following three goals:

- **Leakless ranked search:** For the sake of effective data retrieval and preserving privacy, data users should be able to generate a leakless search query which reveals nothing more than the encrypted query.
- **Privacy-preserving:** Preventing the cloud server from learning additional information rather than seeing encrypted files, queries, and indexes is our highest goals. We describe the privacy requirements in Section 3.
- **Efficiency:** All of the above goals should be realized with a reasonable (or low) computation and communication overhead.

2.2 Preliminaries

Let $\mathcal{D} = \{D_1, \ldots, D_n\}$ be a corpus of n documents, and $id(D_i)$ be the unique identifier of the of document D_i. Let Δ be a dictionary of keywords with size m. Let $\Delta_d = \{w_1, \ldots, w_d\}$ be the dictionary of the d words for the corpus $\mathcal{D}$ such that $\Delta_d \subseteq \Delta$.

Definition 1. (Searchable Encryption). A multi-keyword Searchable Encryption (SE) scheme consists of 6 algorithms, $SE = (KeyGen, BuildIndex, Encryption, Query, Search, Decryption)$ such that:

1. KeyGen (1^λ): Taking a security parameter λ as an input and outputs a secret key K.
2. BuildIndex ($\mathcal{D}$): This algorithm takes in a corpus of documents $\mathcal{D} = \{D_1, \ldots, D_n\}$ and generates an index $\mathcal{I}$.
3. Encryption ($\mathcal{D}, \mathcal{I}, K$): The encryption algorithm takes a document corpus $\mathcal{D}$, an index $\mathcal{I}$ and a secret key K as input and outputs an encrypted document corpus $\mathcal{C} = \{C_1, \ldots, C_n\}$, and a secure index $\mathcal{SI}$.
4. Query (Δ_q, K): This algorithm takes a set of keywords $\Delta_q \subseteq \Delta_d$, and a secret key K as input, and generates an encrypted query q.
5. Search ($q, \mathcal{SI}$): The search algorithm takes an encrypted query q and the secure index $\mathcal{SI}$ as input, it

outputs $\mathcal{R}(q)$ a set of document identifiers whose corresponding documents are the most relevant files to the query q.

6. Decryption (C_i, K): The decryption algorithm takes in an encrypted data file $C_i \in \mathcal{C}$ and a secret key K as input, and outputs D_i.

Definition 2. (History). Let $\mathcal{D}$ be a document corpus. Let Δ_{q_i} be a set of queried keywords of query q_i. A history over $\mathcal{D}$ is a tuple $\mathcal{H}^t = (\mathcal{D}, \Delta_{q_1}, \ldots, \Delta_{q_t})$ over t queries.

The history is information that we are trying to hide from an adversary (cloud).

Definition 3. (Search Pattern). The search pattern over a history $\mathcal{H}^t$ is a tuple, $\Psi = (\hat{\Delta}_{q_1}, \ldots, \hat{\Delta}_{q_t})$, over t queries where $\hat{\Delta}_{q_i}, 1 \leq i \leq t$ is a set of encrypted keywords in the i-th query.

Definition 4. (Access Pattern). The access pattern over a history $\mathcal{H}^t$ is a set, $\Omega = (\mathcal{R}(q_1), \ldots, \mathcal{R}(q_t))$ over t queries.

2.3 Threat model

We consider the cloud server an "honest-but-curious" entity in our model [4,9–11,14]. This means the cloud server complies with the designated protocol ("honest"), but it is eager to collect more information by analyzing the encrypted data, message flows, and the index ("curious"). In our scheme, we assume that the cloud server knows the employed encryption and decryption methods, in addition to the encrypted documents $\mathcal{C}$ and index $\mathcal{SI}$. However, it does not know the key K. We are willing to leak document identifiers $id(D_i), 1 \leq i \leq n$, encrypted queries and the access pattern defined in Definition 4. We can assume that the document sizes will also be leaked, but it can be trivially preserved by a "padding" method [4]. Thus, we classify the attack model to Known Ciphertext Attack in which the adversary only observes the ciphertext, i.e., encrypted documents $\mathcal{C}$, encrypted index $\mathcal{SI}$, and queries.

3 PRIVACY REQUIREMENTS

To address security concerns and preventing a "honest-but-curious" server (see Section 2.3) from collecting users' personal information, the data owner applies a symmetric key cryptography before outsourcing data to the cloud. Although cryptography impedes the cloud prying into the data owner's private data, it cannot address all privacy concerns. Ideally, a cloud should learn nothing but the (encrypted) search results; and it jeopardizes data privacy and even security if the cloud deduces any information from the index, accessed files, queried keywords, *etc.* For example, by analyzing this information, the cloud server may infer the major subject of a document, or even the file's content [17]. Therefore, methods must be designed to prevent the cloud from performing these kind of association attacks. Data privacy and index privacy are default requirements in the literature, and in the following, we enumerate more challenging and more complex privacy requirements.

1. **Search Pattern Privacy:** Uncovering the relation between two or more search requests can lead to more information leakage and data/user privacy violation. Also, the resultant documents, which are ranked based on the query q, provide a good opportunity for the cloud to identify the keywords and their corresponding outsourced documents. Disguising the search pattern from unauthorized parties is among the most complex challenges in this field, so related literature [10,11,18] has not yet addressed this completely issue.

2. **Co-occurrence Keyword Privacy:** Keywords with the same distribution pattern expose more privacy violation risks. In the other words, the privacy of the keywords that co-occur often are tied to each other, and compromising the privacy of one term can lead to privacy violation of the co-occurring term. As a result, the privacy level of co-occurring terms is lower than the regular terms in the same condition. Therefore, we should protect and hide this term dependency to protect the co-occurring terms or at least put them at the same level of privacy protection with singularly occurring terms.

4 LRSE SCHEME

To meet these requirements, we propose the LRSE scheme. LRSE is a privacy preserving multi-keyword ranked search mechanism over encrypted files. We adopt the secure inner product proposed by Wong *et al.*[14]. The index $\mathcal{SI}$ and the query q are both protected using this encryption strategy.

The key idea in our approach is to generate a variety of ciphertexts for high frequency keywords from the corpus $\mathcal{D}$. Hence, multiple encrypted versions of a keyword will be used to search for the same keyword. This directly affects the search pattern ψ that the adversary collects to discover the keyword plaintext. We propose two schemes: Scheme I introduces a solution for searching with sequential scan using our novel chaining notation; and Scheme II, expresses how the chaining notion can be employed to handle controlled searching with an outsourced encrypted index.

4.1 Scheme I - sequential scan

Recall that in Definition 1 a searchable encryption scheme contains a suite of 6 algorithms. Here, we explain how each algorithm works in Scheme I. In the setup phase, beside calling the $KeyGen$ to generate the secret key K, we generate a document-term matrix and a required ciphertext vector. Scheme I's details are presented here:

- **Setup.** Let γ be a $n \times d$ document-term matrix(DTM) which an element e_{ij} represents the frequency of keyword w_j in document D_i. Let $\varphi = (l_1, \ldots, l_d)$ be required-ciphertext vector which $l_i, 1 \leq i \leq d$ shows the number of required ciphertext for keyword w_i. We first extracts the corpus keyword dictionary Δ_d from the entire corpus $\mathcal{D}$ and then generates the document-term matrix γ. Afterwards, we generate the the required-ciphertext vector φ (see below for more details).Then we call $KeyGen$ algorithm to generate the secret key K that will be used to encrypt the documents.

- **RequiredCiphertext.** The probability of querying high frequency keywords is high, so these keywords are more prone to privacy leakages. In a strong attack model[12] where the cloud server is equipped with more knowledge such as the term frequency statistics, the attacker(cloud) can extract invaluable information form the encrypted files[16]. Moreover, by issuing each query to the cloud, the data user is revealing more private information such as her interests and hobbies. Thus, there are two main challenges: 1) hiding the keyword frequencies in the outsourced encrypted documents and 2) obscuring the frequency of the queried keywords which may lead to exposing the underlying keywords[12].

  In an ideal world the keyword frequencies are equal, and the data user uniformly queries all of the available keywords. Thus, the cloud observation from the encrypted files and the queries does not leak any information. Although this is impossible in the real world, our goal is to break down the keyword frequencies to get close to the ideal scenario. For this reason we calculate the average of each column (average of each keyword in the document collection $\mathcal{D}$). Note that because we are employing uniform distribution the average and the median are effectively the same. Let $A = (a_1, \ldots, a_d)$ be the average vector in which $a_i, 1 \leq i \leq d$ shows the average of keywords w_i in corpus $\mathcal{D}$; thus,$\forall\, w_i \in \Delta_d, a_i = \frac{\sum_{j=1}^{j \leq n} e_{ji}}{n}$.

  To determine the number of required ciphertexts for each keyword (l_i) we define a new measure as threshold τ which is the floor of the minimum element in the average vector A. Finally, we determine the number of required-ciphertext vector φ by calculating the ceiling of the maximum frequency for each keyword in DTM-matrix γ divided by the threshold τ:

$$\forall\, w_i \in \mathcal{D},\ k_i = \left\lceil \frac{\max_{ei}^{1 \leq i \leq n}}{\tau} \right\rceil$$

  where $\max_{ei}$ is the maximum value in the w_i column.

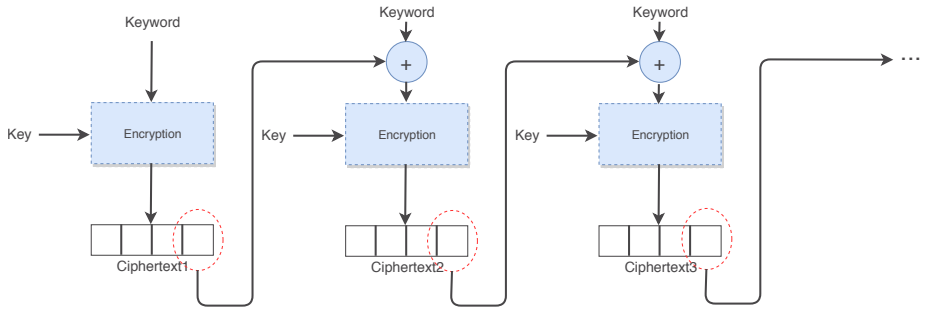


Figure 2. Chaining notion - generating multiple ciphertexts for each keyword.

We divide the maximum frequency of each keyword by the minimum average of the keywords (threshold factor τ) to ensure each encrypted version will occur with the same frequency as the minimum value in the average vector A. With this strategy, the cloud (or an adversary) sees all of the encrypted keywords in the same frequency range. Thus, we gain more uncertainty and a higher level of entropy so identifying the keywords becomes more difficult for the cloud.

- **GenerateCiphertext.** Before we explain how we encrypt the documents, we define the *BuildChain* algorithm which will be used by the *Encryption* algorithm.

  – $BuildChain(K, \Delta_d, \varphi)$. This algorithm takes a secret key K, a document keyword dictionary Δ_d, and a required-ciphertext vector φ as input. It outputs an encrypted keyword dictionary $\hat{\Delta}_{d'} = (\hat{w}_{11}, \ldots, \hat{w}_{ij})$ where $\hat{w}_{ij}$ is the j-th ciphertext of $w_i^{1 \le i \le d}$ and d' is the number of encrypted keywords.

For each keyword $w_i \in \Delta_d$, *BuildChain* generates the first ciphertext $\hat{w}_{i1}$ by encrypting keyword w_i using the secret key K. To generate the second ciphertext $\hat{w}_{i2}$, *BuildChain* encrypts XOR of previous ciphertext $\hat{w}_{i1}$ with keyword w_i using the same secret key K (i.e., $\hat{w}_{i1} \oplus w_i$). Hence, each ciphertext is chained to the previous one. The algorithm continues generating new ciphertext for keyword w_i until we have the expected number of ciphertexts according to the φ vector. Figure 2 shows the chaining process.

Note that compared to other approaches that adopt multiple keys[1], in our chaining methodology, we employ only one key for all of the required ciphertexts. This characteristic mitigates the key management challenges and lessen system costs during the key generation. Moreover, it enables the data users to generate the whole chain on their own side which reduces the communication costs and increases security and privacy. From the privacy viewpoint, keywords with a high frequency get encrypted multiple times (using chaining notion). As a result, it becomes significantly more difficult for the cloud to track down a specific keyword in a single document or across multiple documents.

We explain how *Encryption* algorithm (see Definition 1) works in LRSE. Note that, Scheme I is not an index-based method, so it does not need to take in an index $\mathcal{I}$ (we will employ an index in Scheme II). After taking the inputs, *Encryption* algorithm call the *BuildChain* to generate the required-ciphertext vector φ. The *BuildChain* algorithm returns the encrypted keyword dictionary $\hat{\Delta}_{d'}$ to the *Encryption* algorithm. Then *Encryption* algorithm starts to encrypt each document by fetching each word from the file and if the word is one of the keywords in Δ_d, we randomly choose one of its encrypted versions from $\hat{\Delta}_{d'}$, otherwise we encrypt the word with the secret key[1]. The subroutine "Add" in this algorithm adds the encrypted document (here C_i) to the encrypted file collection ($\mathcal{C}$). Algorithm 1 demonstrates how we encrypt each file.

[1]We assume the random number generator is fair.

Algorithm 1 *Encryption*

```
1:  procedure Encryption(𝒟, K, Δ_d, φ)
2:      Δ̂_d' = BuildChain(K, Δ_d, φ)
3:      for all D_i in 𝒟 do
4:          while !eof do
5:              w = readNextWord(D_i)
6:              if isKeyword(w) then
7:                  ŵ = select randomly an encrypted ciphertext for w from Δ̂_d'
8:              else
9:                  ŵ = encrypt w with secret key K
10:             end if
11:             C_i + = ŵ
12:         end while
13:         Add(𝒞, C_i)
14:     end for
15:     return 𝒞
16: end procedure
```

Since the cloud sees the encrypted document collection $\mathcal{C}$, it generates the DTM matrix (γ') based on the encrypted keywords in $\mathcal{C}$. Thus, the number of columns in γ' is d' rather than d ($d \leq d'$) and the high frequency keywords are eliminated in the whole matrix.

- **GenerateQuery.** In the initialization phase, the data owner and the data user exchange φ vector and the secret key K that enable the data user to make an encrypted query q. In addition, because we have multiple encrypted versions (ciphertexts) of each keyword, the data user can use a portion of the available ciphertexts for each keyword, but the data user must use the same portion for all of the keywords in the same query to not effect the results. For example, the data user may decide to use sixty percent of available ciphertext for each keyword, but he cannot employ sixty percent for the first keyword and forty percent for the second one, because it makes the results imprecise. Finally, encrypted query q is sent to the cloud. The data user may send an optional parameter k to the cloud to retrieve only the top-k resultant documents.

  This is one of the characteristics that distinguishes our approach from other schemes. With each query the data user is able to randomly choose some of the ciphertext for each keyword which delivers more uncertainty and consequently more entropy. Thus, even if consecutive queries share some of their keywords, the cloud is not able to find a pattern between the queries due to using different versions of ciphertext in each query. Moreover, co-occurring terms appear with different ciphertext in the encrypted files, so, finding the co-occurring terms becomes significantly more difficult for the cloud.

  The details of *Query* is shown in Algorithm 2. The data user declares the "portion" manually or it can be determined randomly by the algorithm (like we did in the Algorithm 2). This feature determines the percentage of each keyword ciphertext that will be employed in the query encryption process. For example, if w_i possesses five different ciphertext and the portion is set to sixty percent, the algorithm employs three versions of the ciphertexts randomly for the current query. Moreover, the data user is able to generate the ciphertexts as all encrypted versions are chained together. Note that the plain query can be indicated by today's web search engine such as Bing® and Google®, in which the data users tend to provide a sentence in natural languages or a set of keywords to express their intentions. In this case, we first extract the keywords Δ_q from the plain query.

- **Search.** Before explaining the LRSE search algorithm we define the document and query vector and the

Algorithm 2 *Query*

```
1:  procedure Query(Δ_q, K, φ)
2:      por = randomly choose a portion
3:      for all w_i in Δ_q do
4:          neededVersions=⌈φ_i×por⌉
5:          j = 1
6:          while j ≤ neededVersions do
7:              ŵ = choose randomly one encrypted versions of w_i
8:              q + = ŵ
9:          end while
10:     end for
11:     return q
12: end procedure
```

relevance score:

Definition 5. (Document Vector). Let Δ_d be the dictionary of keywords from corpus $\mathcal{D}$. A document vector $T_i = (f_{w_1}, \ldots, f_{w_d})$ is a normalized vector that demonstrates the normalized frequency of each keyword $f_{w_j}, 1 \leq j \leq d$ in document D_i. Let $T = \{T_1, \ldots, T_n\}$ be a set of all document vectors.

Definition 6. (Query Vector). A query vector Q is a d-element boolean vector such that $\forall\ 1 \leq i \leq d, Q[i] =$ 1 if $w_i \in \Delta_q$, and 0 otherwise.

Definition 7. (Relevance Score). Let T_i denote the normalized frequency vector of the document D_i and Q be the query vector from Δ_q. Their relevance score s_i is inner product of document D_i to query vector Q.

Recall that upon receiving the encrypted document collection $\mathcal{C}$ the cloud builds its own DTM matrix γ' and based on that it generates a set of encrypted document vectors $\hat{T} = \{\hat{T}_1, \ldots, \hat{T}_n\}$. However, the cloud cannot make the real γ matrix and T (see Section 4.1). Upon receiving the encrypted query q, the cloud server executes *Search* algorithm. The LRSE $Search(q, \hat{T}, k)$ algorithm takes the encrypted query q, encrypted document vectors $\hat{T}$, and an optional k, and returns the top-k encrypted resultant documents corresponding to $\mathcal{R}(q)$ to the data user. Consider that the resultant documents are ranked according to their relevance score s.

- **Decryption.** The data user executes $Decryption(C_i, K, \hat{\Delta}_{d'})$ for each C_i in the resultant documents. This algorithm inputs an encrypted document C_i, a secret key K, and an encrypted keyword dictionary $\hat{\Delta}_{d'}$, and outputs D_i. Keywords in the resultant documents are encrypted randomly with a different encrypted piece of the chain. Thus, before decrypting the results, *Decryption* algorithm normalizes the results by changing each keyword's ciphertext with the first ciphertext in the chain. It then decrypts the normalized encrypted document using the secret key K.

4.2 Scheme II - Index

In Scheme II we employ an index structure to increase the search speed. In Scheme I we encrypt each document keyword by keyword, i.e., each block cipher contains one keyword (we consider each block large enough to store an encrypted keyword). In contrast, in Scheme II each block cipher contains 128 bits of a file (which may vary depend on the protocol and employed encryption scheme). Hence, the cloud learns nothing from the encrypted documents $\mathcal{D}$ and is not able to generate its own γ' matrix, so must to rely on the index (provided by the data owner) to find and rank the results.

Furthermore, the cloud is not able to learn the keyword positions in the documents. We also employ the secure

asymmetric inner-product mechanism[14] which prevents the cloud from learning relevance score between two encrypted documents, and is only able to calculate the relevance score between the encrypted query vector $\hat{Q}$ and encrypted document vectors $\hat{T}$. The following describes scheme II in detail:

- **Setup.**This part of the scheme consists of the same steps as Scheme I. We first extracts the corpus keyword dictionary Δ_d from the entire corpus $\mathcal{D}$ and then generates the document-term matrix γ. Afterwards, we generate the required-ciphertext vector φ. In contrast with Scheme I we do not need to actually apply Algorithm 1 and generate multiple ciphertexts for each keyword w_i. Instead, we apply the concept of the chaining notion on the document vectors and represent high frequency keywords with more than one element in the document vectors. For example, assume $\varphi_i = 5$, so keyword w_i is represented with 5 positions in the document vectors. This property has less system cost and improves the system efficiency. Then we call $KeyGen$ algorithm to generate a secret key K that will be used to encrypt the documents.

- **KeyGen.** The key generation algorithm is slightly different from the previous scheme. Beside a secret key K, the $KeyGen$ should create a $(d' \times d')$ invertible random matrix m where d' is the sum of all elements in φ (i.e. all of needed encrypted versions). The data owner runs $KeyGen(1^{\lambda}, d')$ algorithm. The LRSE $KeyGen$ algorithm takes in a security parameter λ and a number d' and it returns a secret key K and an invertible $(d' \times d')$ matrix m. This matrix will be used to encrypt the query and document vectors, so like the secret key, it should be protected.

- **BuildIndex.** The LRSE $BuildIndex(\mathcal{D}, \varphi)$ takes in a document corpus and a required-ciphertext vector φ and outputs the plain index $\mathcal{I}$. This algorithm first generates a normalized document vector T_i for each document D_i based on $tf\text{-}idf$ mechanism[19]. Note that each keyword w_i is presented with multiple positions in the document vector based on value of φ_i. Then using the document vector set T, the algorithm builds a plain index. We adopt a tree-based index structure called keyword balanced binary (KBB) tree proposed by Xia *et al.*[16]. Their secure index structure uses a "Greedy Depth-First Search(GDFS)" algorithm to find the most related nodes (documents) to the query.

  In the KBB tree, each node u consists of 5 elements: node $\mathcal{ID}$, two pointers to the left and right child of node u, document $\mathcal{ID}$ (set to *null* in case u is an internal node), and the last element is a vector T_i that denotes the normalized "$tf \times idf$" values of the keywords in the document D_i. If u is an internal node, each element in vector T_i gets the maximum value of the corresponding keyword among u's child nodes.

  In the index construction phase, we generate a node for each document in the corpus. These nodes are the index tree's leaves. The internal nodes are generated by generating a parent node for each two nodes (same strategy we apply to build a balanced binary tree). The parent node gets the highest value from its children for each element in its T_i vector.

  Figure 3 provides an example of the index tree that is applied in our approach. The corpus in this example consists of 6 files(documents) $f_1, \ldots, f_6$ and 4 keywords(cardinality of the dictionary $d = 4$). Assuming the query vector Q is equal to (0,0.83,0,0.24) and parameter k is set to 3 (*i.e.* number of documents returned to the data user). The figure shows the search process. The search process starts from the root node n and calculates the relevance score of the n_{11} (1.07) and n_{12} (0.428) to the query vector and moves to the n_{11} due to its higher relevance score. The search continues and reaches leaf node f_4 with relevance score of 1.07 and then reaches f_3 and f_2 with 1.127 and 0.498 relevance scores, respectively. Afterwards, the search algorithm gets to the f_1 with 0.524 relevance score and replace f_3 in the result list (because we should return the top-3 relevant files and f_1 relevance score is higher than f_3 score). Finally, the search algorithm goes back to n_{12} node (based on Depth First Search algorithm) and stops there because the relevance score of the n_{12} (0.428) is less than the minimum relevance score (0.524) in the result list. Note that in practice T and Q vectors are encrypted using secret key K and matrix m. We refer the reader to Xia *et al.*[16] for more information about the index structure.

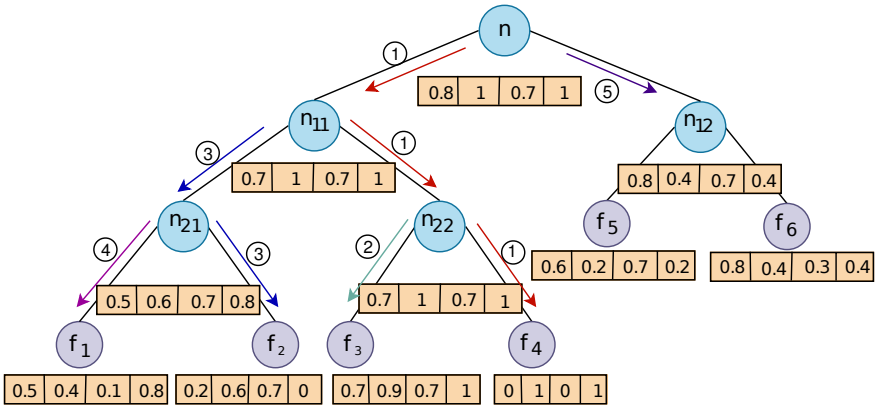


Figure 3. An example of the index tree that is employed in our approach.

- **GenerateCiphertext.** The LRSE $Encryption(\mathcal{D}, \mathcal{I}, m, K)$ algorithm takes a document collection D, a plain index $\mathcal{I}$, a secret matrix m, and a secret key K. The algorithm encrypts all of the documents in $\mathcal{D}$ using the secret key K. To encrypt the index $\mathcal{I}$, we adopt the secure asymmetric inner product by Wong *et al.*[14]. The algorithm can make the cloud server compute the inner product of two encrypted vectors $\hat{T}_i$ and $\hat{Q}$ without revealing any information about the actual values of them. Accordingly, the encrypted subindex is built as $\{m^T T_i\}$ where m is the random matrix that the data owner generates through running $KeyGen$ algorithm. After encrypting the index $\mathcal{I}$ and document corpus $\mathcal{D}$, the data owner performs a random shuffle on the encrypted document vectors, ensuring that the order of the encrypted entries do not reveal any information about the underlying data. The data owner then transfers the encrypted index and documents to the cloud.

- **GenerateQuery.** The $Query(\Delta_q, K, \varphi)$ algorithm is similar to $Query$ algorithm in Scheme I (see Section 4.1), except in the last step, instead of sending the encrypted query q, it builds the corresponding query vector and generates the encrypted query vector $\hat{Q}$ by performing $\{m^{-1}Q\}$. It shuffles the encrypted query vector in the same way that the document vectors are shuffled and sent it to the cloud. The data user may send an optional parameter k to the cloud to retrieve only the top-k resultant documents.

- **Search.** Upon receiving the encrypted query $\hat{Q}$, the cloud server runs $Search(\hat{Q}, \mathcal{SI}, k)$ algorithm which takes in an encrypted query vector $\hat{Q}$, an encrypted index $\mathcal{SI}$ and an optional parameter k. The $Search$ algorithm finds the top-k resultant documents using the tree-based index $\mathcal{SI}$. To gain the relevance score we calculates the inner product of the encrypted document vector in the encrypted query vector.

$$\hat{T}_i.\hat{Q} = (m^{\mathrm{T}} T_i)^{\mathrm{T}} \times (m^{-1} Q) = T_i{}^{\mathrm{T}} m \times m^{-1} Q = T_i{}^{\mathrm{T}} \times Q = T_i.Q$$

  Note that the cloud server only learns the relevance score (similarity) of the documents to the query, while deducing the similarity between encrypted documents is not possible[14]. Finally, the cloud server returns the top-k resultant files to the data user.

- **ResultDecryptor.** In contrast with Scheme I, in this scheme we do not need to normalize the encrypted resultant documents so we achieve lower system cost and faster decryption. The LRSE $Decryption(C_i, K)$ algorithm inputs an encrypted document and the secrets key k and outputs the document D_i. The data user simply call the $Decryption$ algorithm for each encrypted document in the resultant documents.

Table 1. Comparison of related works

Properties	Curtmola *et al* [4] (2011)	Cao *et al* [9] (2014)	Liu *et al* [12] (2014)	Xia *et al* [16] (2016)	Guo *et al* [18] (2018)	LRSE
Preserving access pattern	No	No	No	No	No	No
Server computation	$\mathcal{O}(1)$	$\mathcal{O}(n)$	$\mathcal{O}(n)$	$\mathcal{O}(n)$	$\mathcal{O}(n)$	$\mathcal{O}(n)$
Server storage	$\mathcal{O}(n)$	$\mathcal{O}(n)$	$\mathcal{O}(n)$	$\mathcal{O}(n)$	$\mathcal{O}(n)$	$\mathcal{O}(n)$
Communication	$\mathcal{O}(1)$	$\mathcal{O}(1)$	$\mathcal{O}(n)$	$\mathcal{O}(1)$	$\mathcal{O}(1)$	$\mathcal{O}(1)$
Preserving search pattern	No	Yes	Yes	No	No	Yes
Preserving co-occurrence terms	No	No	No	No	No	Yes
Boolean/multi-keyword search	Boolean	Multi	Boolean	Multi	Multi	Multi

5 PRIVACY AND PERFORMANCE ANALYSIS

Our main goal in this section is to prove the proposed schemes in Section 4 provide privacy and security, as defined in Section 2.1. We also show that in comparison with related works, LRSE has an acceptable complexity in various criteria among previous SSE schemes (see Table 1). This property along with preserving the search pattern and co-occurring terms demonstrate the efficiency of our scheme.

In Section 4.1 we explained that to preserve the user privacy, our goal is to make the document and query vectors as uniform as possible (without compromising the efficiency). Hence, the cloud server is not able to distinguish the high frequency keywords in the encrypted documents. Entropy measure can evaluate the uniformity of document vectors and is employed in many approaches [20–22] to evaluate the privacy. By comparing the entropy of the LRSE document vectors with original ones we demonstrate higher entropy and consequently higher privacy of the document vectors generated by LRSE.

5.1 Entropy of LRSE Document Vectors

We prove that by *expanding* the document vectors using our approach, privacy and security of the outsourced data increases. Note that, adding dummy keywords [4,9] to *extend* the length of the data vectors does not necessarily ensure an increase of the security, and in some case it may even decrease the privacy and security of the outsourced data (see Example 1).

The main idea behind expanding/extending the length of the document vectors is to add more uncertainty to document and query vectors, which results in higher entropy. Although, adding to the length of the document vector can lead to higher entropy, in Example 1 we demonstrate that just extending the document vector's length does not guarantee having a more uniform vector and higher entropy.

Example 1. Consider a document D_1 with 3 keywords. Assume the frequency of each keyword in D_1 is $(2, 3, 4)$, so term-frequency(tf) vector is $(\frac{2}{9}, \frac{3}{9}, \frac{4}{9})$. The entropy of this vector is equal to 1.06.

Now, to increase privacy and security to D_1, we add a new dummy keyword with the frequency of 15. The modified frequency vector is $(2, 3, 4, \mathbf{15})$ and the new term-frequency(tf) vector is $(\frac{2}{24}, \frac{3}{24}, \frac{4}{24}, \mathbf{\frac{15}{24}})$ The first impression is because of adding a dummy keyword, the entropy increases; however the entropy of the new vector is 1.059 which is less than the entropy of the original vector.

In Example 1 we showed that adding dummy keywords to the document/query vectors does not necessarily provide more security/privacy. Defining the property of the new dummy keywords that ensures higher entropy, are not considered in the related literature and we leave it as a future work. However, in Theorem 1 we prove that LRSE scheme provides more security/privacy.

Theorem 1. Given any document vector T_i for document D_i, valid in the LRSE scheme,

$$H(T'_i) \geq H(T_i)$$

where H is the entropy measure and $LRSE(T_i) = T'_i$.

Proof. Consider:

$$LRSE : [0,1]^d \mapsto [0,1]^{d'} \text{ such that } \quad d' \geq d \tag{1}$$

and

$$T_i \in [0,1]^d \quad and \quad T'_i \in [0,1]^{d'}$$

.

The entropy of the document vector $T_i = (f_{w_1}, \ldots, f_{w_d})$ is:

$$H(T_i) = -\sum_{j=1}^{b} f_{w_j} \times \log(f_{w_j})$$

and the entropy of the $T'_i = (f'_{w_1}, \ldots, f'_{w_{d'}})$ is:

$$H(T'_i) = -\sum_{j=1}^{d'} d'_{w_j} \times \log(d'_{w_j})$$

Recall $\varphi = (l_1, \ldots, l_d)$, with $\sum_{\ell=1}^{d} l_\ell = d'$, and for any $f_{w_j} \in T_i$ we have:

$$f_{w_j} = \sum_{k=1}^{l_j} f'_{w_{(k+\alpha_{w_j})}}, \text{ where }, \alpha_{w_j} = \sum_{\ell=1}^{j-1} l_\ell.$$

Hence:

$$\begin{aligned} &- f_{w_j} \log(f_{w_j}) = -(f'_{w_1} + \cdots + f'_{w_m}) \log(f'_{w_1} + \cdots + f'_{w_m}) \\ &where, \ f_{w_j} = \sum_{k=1}^{m} f'_{w_k}. \end{aligned} \tag{2}$$

Moreover, note that $\log(x)$ is a **monotonically increasing** function and T'_i possesses **positive values** (based on (1)), thus we have:

$$\begin{aligned} &- (f'_{w_1}) \times \log(f'_{w_1} + \cdots + f'_{w_m}) \leq -(f'_{w_1}) \times \log(f'_{w_1}) \\ &where, \ f_{w_j} = \sum_{k=1}^{m} f'_{w_k} \end{aligned} \tag{3}$$

By extending the above inequality for all of the keyword frequencies in T_i and T'_i:

$$-\sum_{j=1}^{d} f_{w_j} \times \log(f_{w_j}) \leq -\sum_{m=1}^{d'} f'_{w_m} \times \log(f'_{w_m})$$

Thus we have:

$$H(T'_i) \ \geq \ H(T_i)$$

□

5.2 Privacy attacks

In Section 3 we identified two private information leakage (privacy attacks). In this section we explain how LRSE prevents search pattern and co-occurrence attack. Our schemes are not designed to preserve the user privacy against access pattern attack and we leave it as future work.

In LRSE with each query the data user is able to randomly choose a portion of the ciphertexts for each keyword. Thus, even if consecutive queries share some of their keywords, the cloud is not able to find a pattern between the queries due to using multiple versions of ciphertexts in each query. Further, co-occurring terms appear with different ciphertexts, so, finding the co-occurring terms becomes significantly more difficult for the cloud (see Section 4).

Assume, φ_i is the number of available ciphertexts for keyword w_i, and in each query the query generator uses β percent of the φ_i. The number of possible permutations Γ that the query generator can employ is: $\Gamma = \binom{\varphi_i}{\beta \times \varphi_i}$.

For example, if we have $\varphi_i = 10$ available ciphertexts for keyword w_i, and the query generator employs 40 percent of the ciphertexts in each query ($\beta = 40\%$) the number of possible permutation is: $\Gamma = \binom{10}{4} = 210$. This means, there are 210 distinct possible choices for the $query$ algorithm to ask for the same keyword. In other words, the probability of having 2 queries with same permutation of the ciphertexts for w_i is 0.047% ($\frac{1}{210}$). Note that the main idea is cloud sees all of the keywords are queried with the same probability even if the user starts requesting for the same keyword multiple times.

5.3 Result completeness

To apply the LRSE scheme we first extract keywords from documents in corpus. Technically, keywords with high frequency get extracted considering some linguistic knowledge to avoid stop words such as "the", "for", "if", and *etc.* [23]. We then generate the φ vector based on the minimum value among the average of each keyword in the entire corpus. Further, we assume the random number generator is fair.

Considering all of the aforementioned, there is still a rare chance of incomplete results on the paper. Assume the term frequency of the keyword w_i in document D is f_i, and the number of the available ciphertexts for the corresponding keyword is φ_i. As long as $f_i \geq \varphi_i$ there is no problem (case 1), because the corresponding C (encrypted D) contains all of the available ciphertexts and no matter which ciphertext versions are employed in the user query, C (and consequently D) will be placed among the possible results.

If $f_i \leq \varphi_i$ the user query may contain encrypted versions of w_i that do not exist in D (case 2). Recall that along with the query, the user also sends the parameter k to retrieve only the top-k documents. In case 2 if k is relativity less than the number of documents that includes w_i there is still no problem because D's relevance score to the query is very low and even if D had all of the encrypted versions of w_i, it would not be placed among the top-k files (case 3). Note that number of documents that $f_i \leq \varphi_i$ should be very low, otherwise the keyword extraction algorithm would not choose w_i as a keyword in the first place.

The problem occurs when the user asks for all of the documents that includes keyword w_i (case 4). In this condition, the results may be incomplete, because the user employs a portion of the available ciphertexts which may not be used in D. For example, consider we have a corpus with four documents (d_1, d_2, d_3, d_4), and the term frequency of keyword w_i in each document is $(25, 32, 6, 29)$. Also, consider that the number of available ciphertexts $\varphi_i = 10$ and the portion is set to forty percent for query q (e.g., forty percent of the available ciphertexts are employed to generate q by the query generator). Thus, the query generator randomly selects four versions of available ciphertexts (forty percent of φ_i). No matter what versions of the ciphertexts are employed in the query q, d_1, d_2, and d_4 will be placed among the possible results because the term frequency of keyword w_i in those documents are greater than φ_i, so those documents possess all of the available ciphertexts

for w_i (case 1). However, if $f_i \leq \varphi_i$ (case 2), D_3 may get excluded from the results due to not possessing the same versions that are employed in the query q. If k is set to 1,2, or 3, this issue does not effect the result accuracy because the user is only interested in the first top-k documents and d_3 will not be placed in the top-k list. The problem occurs when users demands all of the documents containing w_i (case 4).

To tackle this challenge (case 4), in Scheme I, we can inject the missed ciphertexts in the corresponding documents. In Scheme II, we do not even need to touch the encrypted documents; the only action is to adjust the corresponding document vectors regarding the missed ciphertexts. In order not to effect the relevance score, we employ the same strategy the related literature employs by adding dummy keywords while not affecting the relevance score[9,14].

5.4 Security

The secure asymmetric inner product[14] is widely used in many existing secure search schemes[9,10,16,18,24] and it has been proved to be secure in known ciphertext attacks. In this section we show that chaining the ciphertexts are not weakening the underlying secure symmetric encryption scheme; and we give the security proofs for the schemes presented in Section 4.

Definition 8. (Symmetric encryption scheme)[4]. A symmetric encryption scheme is a set of three polynomial-time algorithms $(\mathcal{G}, \xi, \mathcal{D})$ such that $\mathcal{G}$ takes an unary security parameter k and returns a secret key K; ξ takes a key K and n-bit message m and returns ciphertext c; D takes a key K and a ciphertext c and returns m if K was the key under which c was produced. We refer to Curtmola *et al.*[4] for formal definition of security for symmetric encryption schemes.

Definition 9. (Indistinguishability). Let G and E be two random variables distributed on $\{0,1\}^n$. A SSE scheme is indistinguishable secure if for all non-uniform probabilistic polynomial-time adversaries $\mathcal{A} : \{0,1\}^n \rightarrow \{0,1\}$, for all polynomial "$poly$" and all sufficiently large k the distinguishable probability (also called advantage of adversary) is:

$$Adv\mathcal{A} = |\Pr[\mathcal{A}(G)] - \Pr[\mathcal{A}(E)]| \leq \frac{1}{poly(k)} \tag{4}$$

Definition 10. (Novel chaining notion). Let $(\mathcal{G}, \xi, \mathcal{D})$ be an indistinguishable secure symmetric encryption scheme with $\xi : \{0,1\}^k \times \{0,1\}^n \rightarrow \{0,1\}^n$. Let $w_i \in \{0,1\}^n$ be a keyword in the dictionary Δ_d. Given a natural number of ℓ The novel chaining notion of w_i ($NCN(w_i)$) is defined as:

$$\begin{aligned} &NCN(w_i) = (NCN^{(1)}(w_i),\ NCN^{(2)}(w_i),\ \ldots,\ NCN^{(\ell)}(w_i)) \\ &\text{where}\ ,\ \ NCN^{(1)}(w_i) = \xi(w_i), \\ &\text{and}\ \ \ NCN^{(j)}(w_i) = \xi(w_i \oplus NCN^{(j-1)}(w_i))\ \forall j \in \{2,\ldots,\ell\}. \end{aligned}$$

Theorem 2. The novel chaining notion is indistinguishable secure against known plaintext attack.

Proof. Let $(\mathcal{G}, \xi, \mathcal{D})$ be the an indistinguishable symmetric encryption scheme. Moreover, recall $\xi : \{0,1\}^k \times \{0,1\}^n \rightarrow \{0,1\}^n$, so an encrypted vector $\hat{V}$ is a valid vector as an input for ξ (because both encrypted ($\hat{V}$) and plain (V) vectors are in $\{0,1\}^n$). Hence for any w_i in the dictionary Δ_d and for any j, $NCN^j(w_i)$ is in $\{0,1\}^n$. Since $(\mathcal{G}, \xi, \mathcal{D})$ is indistinguishable secure scheme, the attacker is not able to find any relation between input and output except for a negligible amount[4], so the outputs is independent of the inputs, otherwise the encryption function (ξ) is not indistinguishable. Hence, keyword w_i is independent from $NCN^{(1)}(w_i)$ and $NCN^{(1)}(w_i)$ is independent from $NCN^{(2)}(w_i)$. Now we have to prove that every pair of the chains such as $NCN^{(j)}(w_i)$ and $NCN^{(j+i)}(w_i)$ is indistinguishable. By contradiction, let's assume $NCN^{(j)}(w_i)$ and $NCN^{(j+i)}(w_i)$ are distinguishable, then even the pairwise $NCN^{(j)}(w_i)$ and $NCN^{(j+1)}(w_i)$ is distinguishable too, which is a contradiction. Therefore, $NCN(.)$ is secure. □

Theorem 3. LRSE is an indistinguishable secure scheme against known ciphertext attack.

Proof. Let $(\mathcal{G},\xi,\mathcal{D})$ be an indistinguishable symmetric encryption scheme. Let the $\mathcal{C} = \{\xi(D_i)\ ,1 \leq i \leq n\}$ be the encrypted document collection. Let $\mathcal{SI}$ be the encrypted searchable index, and let the $NCN(.)$ be our secure chaining notion. Since $(\mathcal{G},\xi,\mathcal{D})$ is indistinguishable secure the encrypted documents $\xi(D_i)$ $,1 \leq i \leq n$ are indistinguishable secure from a random string $\{0,1\}^*$. Hence, the encrypted documents are indistinguishable secure. Further, the encrypted index $\mathcal{SI}$ and encrypted query vectors are secured using asymmetric inner product[14]. In Theorem 2 we proved that the NCN is indistinguishable secure, so chaining the ciphertexts does not weaken the symmetric encryption. Recall that in Scheme I we encrypt each file word by word using our novel chaining notion. In Scheme II we take advantage of the same idea in generating the document vectors and encrypt the documents block by block (instead of word by word). Thus, the encrypted documents $\xi(D_i)$, $1 \leq i \leq n$, the index $\mathcal{SI}$ and the encrypted query vectors, and the novel chaining notion (NCN) are indistinguishable secure, so LRSE is indistinguishable secure. □

5.5 Efficiency and System Costs

Although two(multi)-party computation can address our designed goals, they suffer from low efficiency. They usually employ a n-path (in best case two-path) algorithm which means the retrieval phase needs two rounds of communication between the cloud server and data user[25]. Further, one of the biggest drawbacks is the complexity of the system and even for basic operations requires significantly more complicated computations.

Thus, scholars propose new methodologies to make a trade-off between privacy/security and efficiency. Table 1 compares LRSE with the previous work. In comparison with SSE schemes, LRSE preserves both the search pattern and co-occurring terms which are not supported in the multi-keyword SSE schemes.

Specifically, in comparison with Curtmola *et al.*[4] LRSE needs more sever computation, but LRSE supports multi-keyword search queries and also preserves the search pattern and co-occurring terms privacy which are not among Curtmola *et al.*[4] achievements. In compare with Cao *et al.*[9], LRSE preserves co-occurring terms. Both methods are at the same level of system costs, because in Cao *et al.*[9] work, to increase the privacy, length of the document vectors are extended with dummy keywords. In LRSE, we expand the length of the document vectors to increase the uncertainty and at the same time hide the search pattern and co-occurring terms (two birds with one stone). Moreover, in Section 6.1 we show that LRSE reaches a higher level result accuracy compared to Cao's approach.

In comparison with a recent work[18], LRSE preserves search pattern and co-occurring terms privacy. Moreover, in Guo *et al.*'s[18] approach, a trusted proxy is considered in the system model which increases the system cost.

6 IMPLEMENTATION AND ANALYSIS

We conducted a comprehensive experimental evaluation of LRSE on 203 English books[26] with more than five million words (excluding stop-words) and more than 2200 keywords. Our experiment includes a user and a server. Both entities are implemented using Java (JDK 1.8.0_111) and are executed on Windows 7 machines with Core2 Duo CPU at 3.17 GHz and 8 GBs of RAM. The user acts as the data owner and data user, and the server acts as the cloud server. We ran the experiments 10 times and difference between the maximum and minimum output of the same experiment was lees than 0.5%. For example, the result accuracy experiment showed less than a 0.2% difference in 10 runs.

Recall that in Section 4 we explained that the user's query is a set of keywords or a sentence in natural language.

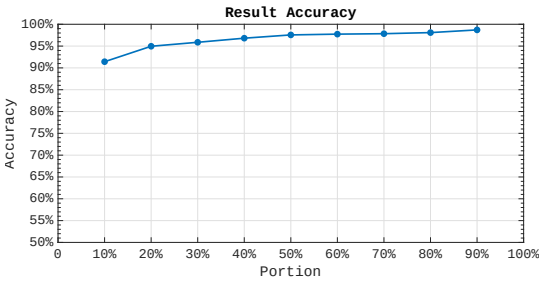


Figure 4. Effect of portion on result accuracy over 5000 queries.

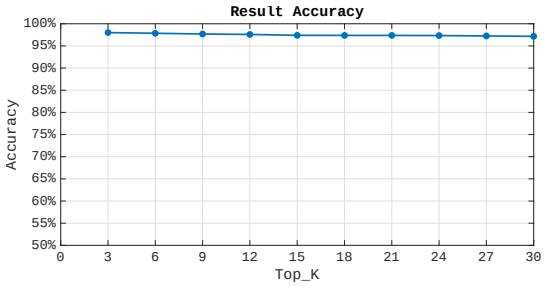


Figure 5. Result accuracy based on top_k over 5000 queries.

We assumed that the number of keywords in each query is between five and ten. In other words, each time our query simulator is generating a random number in this range (5-10) which indicates the number of keywords in the corresponding query.

Our analysis includes result accuracy and privacy assessment. In general, the cloud server observes two groups of vectors: document vectors and query vectors. Our experimental evaluation demonstrates a higher privacy in both groups, and a higher level of result accuracy compared with Cao's result precision [9].

6.1 Result Accuracy

Section 4.1 describes a certain number of available ciphertexts (portion) are selected in every query. Although, the same portion for each keyword is employed in the query, it may effect the accuracy of the results due to the reason we explained in Section 5.3. Thus, there is a small chance to lose some result accuracy when the number of available ciphertexts for a keyword is bigger than its frequency in a specific document (because the encrypted versions that are employed in the document may differ from the ones in the query). In other words, when the cloud server returns the top-k documents based on their similarity to the query some of the real top-k relevant documents may be excluded.

This issue occurs in Cao's [9] work when dummy keywords are inserted into each document vectors. conversely, to boost the privacy level, LRSE does not insert dummy keywords, instead we employ multiple ciphertexts to represents each keyword (based on their frequency) and for this reason (not adding noise to the document vectors) we expect to see higher level of result accuracy in LRSE. To evaluate the accuracy of the LRSE results we define the result accuracy $R_{acc} = \|(K' \cap K)\|/\|K\|$ where K and K' are sets of expected result documents and documents retrieved by cloud server using LRSE. Additionally, $\|A\|$ determines the number of elements in set A. Figure 4 and Figure 5 demonstrate our results.

Recall that the "portion" determines percentage of each keyword ciphertext that will be employed in query encryption process (see Section 4.1). Figure 4 shows the effect of portion on result accuracy over 5000 queries. As the diagram shows LRSE achieves more than 91% result accuracy even when only 10% of the available ciphertexts are employed. Note that in our calculation in Section 5.2 we assumed 40% of the ciphertexts are employed and setting the portion to 10% increases the number of possible permutations and it becomes harder for the cloud server to analyze the access pattern. Moreover, increasing the portion from 10% to 20% raises the result accuracy around 5% and it gets to 95% which seems to be a reasonable trade-off to gain more result accuracy.

Figure 5 demonstrates the effect of top-k on the result accuracy. As the figure shows LRSE achieves to more than 98% result accuracy for top-3 documents. Top-10 or top-15 seems to be a reasonable top-k in our simulation since we have 203 books in our dataset. Even if we consider top-20 (which is 10% of our repository), we have more than 97% result accuracy. In comparison to MRSE (Cao's work), LRSE achieves a higher precision in

results. In MRSE standard deviation (σ) plays an important role which can effect the result accuracy. The bigger the σ is, the less result accuracy we have and it becomes more difficult for the cloud server to obtain information about the user data. Although the σ is not a parameter to be set up in LRSE, we calculate the standard deviation of the document vectors after applying LRSE. The average of σ for document vectors in lRSE is 1.34 with minimum of 0.97. Compared to MRSE (when the σ is 1) LRSE is 7% more accurate than MRSE, and this difference becomes more if the average of LRSE's σ decreases to 1.

In both LRSE and MRSE as the top-k increases, the result accuracy decreases. In MRSE, this is because of the dummy keywords which can effect the similarity scores (dummy keywords may reduce some document scores which are in the real top-k results or increase the score of some documents out of the real top-k results). In LRSE, with increasing top-k, documents with less relevance to the query are placed in the result set. The frequency of the queried keywords in some documents is insufficient to cover all of the available ciphertexts for the corresponding keyword. Thus when the query asks for the missed ciphertext versions, those documents do not get into the resultant set even when they contain the required keywords. In Section 5.3 we propose to inject the missed ciphertext versions to prevent this problem. However, the results show that LRSE loses less than 1% accuracy from top-3 to top-30, which is tolerable. More importantly, this happens to less relevant documents to the query.

6.2 Document Vectors

6.2.1 *Entropy of Document Vectors*

We employed Shanon entropy to calculate entropy of the original and LRSE document vectors: $H(V) = -\sum_{i=1}^{n} p_i \log_2 p_i$, where V is the document vector, n is the number of keywords, and p_i is probability of keyword i.

To calculate LRSE entropy progress, we define a measure as entropy improvement $H_{imp} = (H(V_{li}) - H(V_{oi}))/H(V_{oi})$, where $H(V_{li})$ is the entropy of the document i vector in LRSE and $H(V_{oi})$ is the entropy of the original document vector of the same document.

In Section 5.1 we proved that the entropy of document vectors which are generated by LRSE are greater than or equal to the entropy of original vectors.The simulation results emphasizes our theorem and shows at least a 25% entropy improvement in all of the documents and in some documents around 90%. Figure 6 demonstrates the first 20 documents entropy improvement.

Note that, some documents such as "Document6" in Figure 6 may possess a high frequency of some keywords because specifically discuss a special topic. For example, legal terminologies are used heavily in the congress documents which increases their frequencies and drastically reduces the entropy of the document vectors and threatens owner/user privacy. In LRSE, we break down these high frequency occurrences to a couple of frequencies in the average frequency range (see Section 4.1). For example, assume the frequency of keyword w_i in document D_j is 72 and the threshold is $\tau = 25$, thus $\varphi_i = \lceil \frac{72}{25} \rceil = 3$. Hence, LRSE divides the w_i frequency to 3 smaller parts (say 22, 24, 26) which are close to the frequency average (25), and then generates 3 ciphertexts using the chaining notion for each part. Because of this LRSE feature, our observation and result simulation show a 90% entropy improvement in some of the documents.

6.2.2 *Standard Deviation of Document Vectors*

A low standard deviation (σ) represents that most of the keyword frequencies are very close to the average and consequently to each other. Note that, the keywords can be deduced or identified in a strong attack model that the cloud server is equipped with more knowledge such as the term frequency statistics of the document collection [16]. For example, the frequency of economic terminologies is much higher than the other keywords in a budget document. Thus, the more keyword frequencies become closer to each other the more difficult

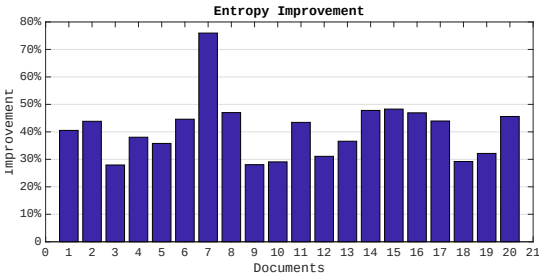


Figure 6. Entropy improvement of the first 20 documents.

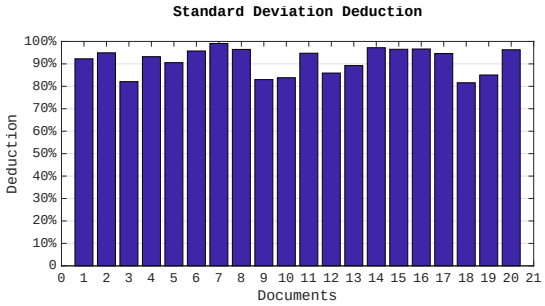


Figure 7. Standard deviation deduction in the first 20 documents.

identifying the keywords becomes.

Figure 7 demonstrates the standard deviation reduction of first 20 documents which are calculated by comparing the standard deviation of the original document vector and the corresponding vector in LRSE. The results show at least a 80%, σ reduction in each document. This means the frequency of the keywords are at least 80% closer to each other which preserve the data privacy against privacy attacks such as frequency statistical analysis mentioned above.

6.3 Query Vectors

Although documents' vectors are constant and barely change, the query vectors are prone to change as the user intentions and demands change. In other words, the number of queries increases over time, more information such as access pattern will be revealed to the cloud. For this reason, we dedicate the third part of our analysis to query vectors. The result analyses shows that LRSE protects the access pattern and privacy of the queries even when the number of queries grows.

6.3.1 Euclidean Distance from Ideal Vector

To preserve the access pattern, the ideal is the cloud server sees all of the queried keywords with the same frequency. In other words, after receiving m search requests the normalized vector of queries on n keywords is: $(\frac{1}{n}, \frac{1}{n}, \frac{1}{n}, \ldots)$, which means that to predict the next queried keywords or discovering the underlying plain keywords, the cloud server has no more chance than flipping a coin, which is the best case scenario.

We apply Euclidean distance measure to determine the distance between the ideal vector and the original/LRSE query vector. The less the Euclidean distance is, the closer we are to the ideal vector, and the more private is the data. To calculate the query vector, we processed the frequency of each queried keyword after every 3000 queries (for both original and LRSE queries). We then calculate its Euclidean distance from the ideal vector.

Figure 8 demonstrates the Euclidean distance improvement. The results show that the query frequency vector is at least 67% closer to the ideal vector after 30000 search requests submitted. Note that this is the minimum improvement due to using uniform distribution. We randomly select some keywords to create the queries. However in the real world users keep asking for documents in their field of expertise or their interests which makes the original frequency vector farther away from the ideal vector.

6.3.2 Standard Deviation of Query Vectors

In Section 6.2.2 we explained the importance of having low standard deviation(σ) and analyzed the σ of LRSE document vectors. In this section we study the σ reduction of the query vectors. Unlike the documents, the number of the queries and consequently query vectors increases during the time and for this reason we show the σ reduction over time. We employed the same methodology in Section 6.2.2 to evaluate the standard deviation reduction.

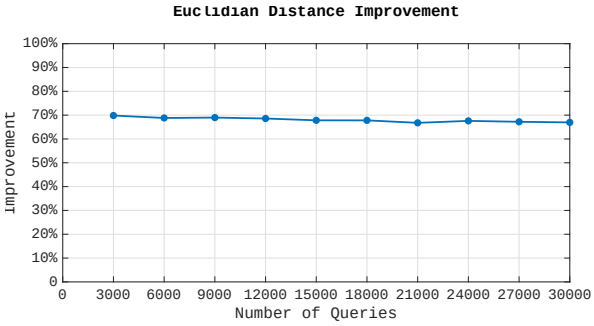


Figure 8. Euclidean distance improvement over 30000 queries.

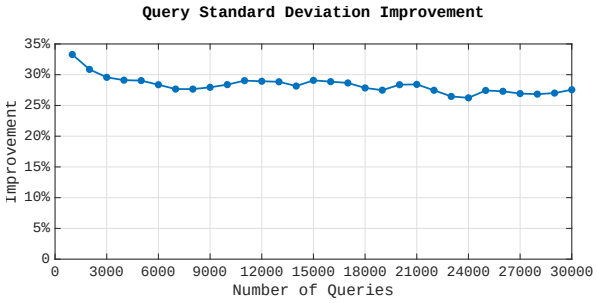


Figure 9. Standard deviation improvement of the queries over 30000 queries.

Figure 9 demonstrates that LRSE reduces the standard deviation 25-30%. In other words, identifying the keywords are 30% more difficult using LRSE. Moreover the reduction amount stays in the same range (25%-30%) as the number of queries increases which shows the stability of LRSE.

7 RELATED WORK

The symmetric searchable encryption (SSE) introduced by Song *et al.*[1], where each word is encrypted under a particular two-layered encryption. Afterward, Goh[15] improved the search request time using Bloom filters. Chang *et al.*[3] and Curtmola *et al.*[4] then enhanced the security definitions, constructions and proposes some improvements. However, traditional symmetric encryption schemes only supports exact keyword search and cannot endure any kind of format inconsistency or minor imperfections. To address this issue, Li *et al.*[27] propose a method in which returned documents are designated according to the predefined keywords or the closest possible matching documents, based on keyword similarity semantics. Kuzu *et al.*[28] also tackle this challenge and propose a method with more efficiency and less overhead.

All these approaches support only Boolean search. Thus, finding the most relevant documents for the data user's multi-keyword search request is a crucial challenge. To resolve this challenge, Cao *et al.*[9] introduce a method that allows data users to apply a multi-keyword search request on the encrypted files with ranking capability. Cao *et al.*[9] chose the similarity measure of "coordinate matching", that is, as many matches as possible. And to capture the relevance of outsourced documents to the query keywords, the "inner product similarity" is employed. Later, Fu *et al.*[10] propose a model that makes the query results more personalized for each user based on their search history. Considering the user search history, they built a user interest model for individual users with the help of the semantic ontology WordNet. Moreover, Yu *et al.*[11] propose a user-ranked multi-keyword method to prevent data privacy leaks in cloud-ranked methods. They employed the vector space model and homomorphic encryption. The vector space model helps to provide sufficient search accuracy, whereas the homomorphic encryption enables users to get involved in the ranking procedure, while the remaining computing work is done on the server side. In a recent work, Guo *et al.*[18] propose a multi-keyword SSE approach which support multi data owners, and to tackle the key management challenges they exploit a trusted proxy.

However, these schemes function based on the symmetric key encryption, where the same key is employed to encrypt and decrypt the data. Another approach is to use public key encryption. Boneh *et al.*[2] defined the concept of the "public key encryption with keyword search", and later, several methods[6,29–32] were introduced to improve the efficiency and system cost of the public-key searchable encryption schemes. Basically, these methods exercise one key for encryption and another key for decryption. Thus, data users who own the private key are able to search the outsourced data encrypted by the public key.

Nevertheless, all these methods suffer from private information leakage such as access pattern, search pattern, and co-occurrence information leakage. Cao *et al.*[9] believe that, to solve this problem, we must "touch" the whole outsourced dataset, which ends in losing the efficiency so other investigators chose not to impede these leaks which kept them out of the designed goals.

8 CONCLUSIONS

The problem of leakless preserving privacy multi-keyword ranked search in SSE schemes, addressed here. We built a private model to prevent two kinds of leakage: search pattern and co-occurrence private information leakage. We employed the asymmetric inner-product to calculate the relevance score of each document with respect to the query. We also introduced our chaining encryption notion to generate multiple ciphertexts for the same keyword. All this leads to more uncertainty and a uniform probability model for the keywords distribution. Furthermore, with our chaining encryption notion, the data user is able to randomly choose a portion of the ciphertexts for each keyword. Thus even if consecutive queries share some keywords, the cloud is not able to find a pattern between the queries due to using different versions of ciphertexts in each query. Moreover, co-occurring terms appear with different ciphertexts in the encrypted documents, and so, finding the co-occurring terms becomes significantly more difficult for the cloud. Next, to tackle the challenge of leakless multi-keyword ranked search, we propose the LRSE scheme and define the privacy requirements. In addition, we explain each level of the LRSE scheme in details and describe the required algorithms.

Furthermore, we performed the security and privacy analysis to show the efficiency of our proposed approach. We proved the the novel chaining notion and consequently LRSE is secure and compared complexity of our proposed scheme with related work in various criteria such as server computation, communication, etc. Looking to the future, we will modify LRSE to prevent access pattern attack.

DECLARATIONS

Authors' contributions

Each author contributed equally to the paper.

Availability of data and materials

Not applicable.

Financial support and sponsorship

This research is partly supported by the Natural Sciences and Engineering Research Council of Canada.

Conflicts of interest

All authors declared that there are no conflicts of interest.

Ethical approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

REFERENCES

1. Song DX, Wagner D, Perrig A. Practical techniques for searches on encrypted data. Proceeding 2000 IEEE Symposium on Security and Privacy. S&P 2000; 2000 May 14–17; Berkeley, USA. New York: IEEE; 2000.
2. Boneh D, Di Crescenzo G, Ostrovsky R, Persiano G. Public key encryption with keyword search. In: International conference on the theory and applications of crypto-graphic techniques; 2004 May 2–6; Interlaken, Switzerland. Berlin: Springer; 2004. pp. 506–522.
3. Chang YC, Mitzenmacher M. Privacy preserving keyword searches on remote encrypted data. Proceedings of the Third international conference on Applied Cryptography and Network; 2005 June 7–10; New York, USA. Berlin: Springer; 2005.
4. Curtmola R, Garay J, Kamara S, Ostrovsky R. Searchable symmetric encryption: improved defifinitions and effificient constructions. Proceedings of the 13th ACM conference on Computer and communications security 2006; Alexandria Virginia, USA. New York: J Comput Secur; 2006.
5. Golle P, Staddon J, Waters B. Secure conjunctive keyword search over encrypted data. In: International Conference on Applied Cryptography and Network Security; 2004 June 8–11; Yellow Mountains, China. Springer, 2004. p. 31–45. Berlin: Springer; 2004.
6. Liu Q, Wang G, Wu J. Secure and privacy preserving keyword searching for cloud storage services. J Netw Comput Appl 2012; 35: 927–33.
7. Cash D, Jarecki S, Jutla C, Krawczyk H, Roşu MC, et al. Highly-scalable searchable symmetric encryption with support for boolean queries. In: Advances in Cryptology–CRYPTO 2013; 2013 August 18–22; Santa Barbara, USA. Berlin: Springer, 2013. pp. 353–373.
8. Gai K, Zhu L, Qiu M, Xu K, Choo KKR. Multi-access filtering for privacy-preserving fog computing. IEEE Trans on Cloud Comput 2019; 1–1.
9. Cao N, Wang C, Li M, Ren K, Lou W. Privacy-preserving multi-keyword ranked search over encrypted cloud data. IEEE Trans Parallel Distrib Syst 2014; 25: 222–33.
10. Fu Z, Ren K, Shu J, Sun X, Huang F. Enabling personalized search over encrypted outsourced data with efficiency improvement. IEEE Trans Parallel Distrib Syst 2016; 27: 2546–59.
11. Yu J, Lu P, Zhu Y, Xue G, Li M. Toward Secure Multikeyword Top-k Retrieval over Encrypted Cloud Data. IEEE Trans Dependable Secure Comput 2013 July; 10: 239–50.
12. Liu C, Zhu L, Wang M, Tan Ya. Search pattern leakage in searchable encryption: Attacks and new construction. Inf Sci 2014; 265: 176–88.
13. Perc M. Evolution of the most common English words and phrases over the centuries. J R Soc Interface 2012:rsif20120491.
14. Wong WK, Cheung DW, Kao B, Mamoulis N. Secure knn computation on encrypted databases. In: Proceedings of the 2009 ACM SIGMOD International Conference on Management of data; 2009, 29 June-2 July; Rhode Island, USA. New York: ACM, 2009. pp. 139–152.
15. Goh EJ. Secure indexes. Cryptology ePrint Archive, Report 2003/216. Avialable from: http://eprint.iacr.org/2003/216/. [Last accessed on 25 Sep. 2020]
16. Xia Z, Wang X, Sun X, Wang Q. A Secure and Dynamic Multi-Keyword Ranked Search Scheme over Encrypted Cloud Data. IEEE Trans Parallel Distrib Syst 2016; 27: 340–52.
17. Zerr S, Demidova E, Olmedilla D, Nejdl W, Winslett M et al. Zerber: r-confifidential indexing for distributed documents. In: Proceedings of the 11th international conference on Extending database technology: Advances in database technology; 2008 March 25–29; Nantes, France. New York: ACM; 2008. pp. 287–298.
18. Guo Z, Zhang H, Sun C, Wen Q, Li W. Secure multi-keyword ranked search over encrypted cloud data for multiple data owners. J Syst Softw 2018; 137: 380–95.
19. Ramos J. Using tf-idf to determine word relevance in document queries. In: Proceedings of the first instructional conference on machine learning. vol. 242. New Jersey, USA; 2003. pp. 133–42.
20. Begum RS, Sugumar R. Novel entropy-based approach for cost-effective privacy preservation of intermediate datasets in cloud. Cluster Computing 2019, 22: 9584–88.
21. Niu B, Li Q, Zhu X, Cao G, Li H. Enhancing privacy through caching in location-based services. In: 2015 IEEE conference on computer communications (INFOCOM); 2015 26 April-1 May; Hong Kong, China. New York: IEEE, 2015. pp. 1017–1025.
22. Palanisamy B, Liu L. Mobimix: Protecting location privacy with mix-zones over road networks. In: 2011 IEEE 27th International Conference on Data Engineering, 2011 April 11–16; Hannover, Germany; New York: IEEE, 2011. pp. 494–505.
23. Hulth A. Improved Automatic Keyword Extraction given More Linguistic Knowledge. In: Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing. EMNLP '03. USA: Association for Computational Linguistics; 2003. p. 216–223. Available from: https://doi.org/10.3115/1119355.1119383.
24. Sun W, Wang B, Cao N, Li M, Lou W, et al. Privacy-Preserving Multi-Keyword Text Search in the Cloud Supporting Similarity-Based Ranking. In: Proceedings of the 8th ACM SIGSAC Symposium on Information, Computer and Communications Security. ASIA CCS '13. New York, NY, USA: Association for Computing Machinery; 2013. p. 71–82. Available from: https://doi.org/10.1145/2484313.2484322.
25. Goldreich O, Ostrovsky R. Software protection and simulation on oblivious RAMs. Journal of the ACM (JACM) 1996;43:431–73.
26. Publication G. Archived Textbooks; Available from: https://www.gutenberg.org/ebooks. Last accessed: 27-August-2020.
27. J L, Q W, C W, N C, K R, et al. Fuzzy Keyword Search over Encrypted Data in Cloud Computing. In: 2010 Proceedings IEEE INFOCOM. San Diego, CA, USA; 2010. pp. 1–5.
28. Kuzu M, Islam MS, Kantarcioglu M. Effificient similarity search over encrypted data. In: Data Engineering (ICDE), 2012 IEEE 28th International Conference on, 2012 April 1-5; Washington, USA; New York: IEEE, 2012. pp. 1156–67.
29. Bellare M, Boldyreva A, O'Neill A. Deterministic and effificiently searchable encryption. In: Annual International Cryptology Conference,

2007 August 19–23; Santa Barbara,USA; Berlin: Springer, 2007. pp. 535–52
30. Attrapadung N, Libert B. Functional encryption for inner product: Achieving constant-size ciphertexts with adaptive security or support for negation. In: International Workshop on Public Key Cryptography; 2010 May 26–28; Paris, France; Berlin: Springer, 2010. pp. 384–402.
31. Boldyreva A, Chenette N, Lee Y, O'neill A. Order-preserving sym-metric encryption. In: Annual International Conference on the Theory and Applications of Crypto-graphic Techniques, 2009 April 26–30; Cologne, Germany; Berlin: Springer, 2009. pp. 224–41.
32. Ocansey SK, Wang C. Search over encrypted cloud data with secure updates. In: 2019 IEEE 19th International Conference on Software Quality, Reliability and Security Companion (QRS-C), 2019July 22–26; Sofia, Bulgaria; New York: IEEE, 2019. pp. 380–86.

Chen. *J Surveill Secur Saf* 2020;1:x
DOI: 10.20517/jsss.2020.26

Journal of Surveillance, Security and Safety

Editorial

Open Access

Welcome to the *Journal of Surveillance, Security and Safety*: A New Open-Access Scientific Journal

Xiaofeng Chen

School of Cyber Engineering, Xidian University, Xi'an 710126, Shaanxi, China.

Correspondence to: Prof. Xiaofeng Chen, School of Cyber Engineering, Xidian University, Xi'an 710126, Shaanxi, China. E-mail: xfchen@xidian.edu.cn

How to cite this article: Chen X. Welcome to the *Journal of Surveillance, Security and Safety*: A New Open-Access Scientific Journal. *J Surveill Secur Saf* 2020;1:102-5. http://dx.doi.org/10.20517/jsss.2020.26

Received: 23 Sep 2020 **Accepted**: 23 Sep 2020 **Published**: xx Sep 2020

Academic Editor: Xiaofeng Chen **Copy Editor**: Cai-Hong Wang **Production Editor**: Jing Yu

INTRODUCTION

This journal has been anxiously awaited by those interested in the security and safety problems associated with artificial intelligence, the blockchain, databases, cloud computing, multimedia, wireless networks, IoT, and other computer science and cryptography technologies.

In recent years, security threats faced by new technologies are emerging without end, while the security requirements of traditional technologies are increasing. Interest in these areas has grown rapidly, mainly including the security issues from the perspectives of AI, data, network, computing, cryptography, access control, industries, policies, models, *etc*. The deeper is the awareness of private data that people have, the higher is their need for application security.

Explosive growth in the number and scale of machine learning models, requiring robustness in their training and testing periods against adversarial attacks, is one of the most striking characteristics of the current technological landscape about artificial intelligence. The expansion of research related to both machine learning-based attacks and their interpretation has driven the rapid growth of the research area in secure machine learning models. The urgent need for security research is not a unique trend in a certain field, and tricky challenges regarding security issues also appear in other popular areas, e.g., the blockchain, where multiple system components, such as consensus mechanism and smart contract, have been found susceptible to malicious attacks that destroy the credibility of the system.

JOURNAL OF SURVEILLANCE, SECURITY AND SAFETY: WHY A NEW JOURNAL?

While related Special Sections of existing journals have appeared regularly, research results in the fields of surveillance, security, and safety have been published in many different journals, creating a somewhat dispersed audience. In response, the *Journal of Surveillance, Security and Safety* (*JSSS*) has been badly needed as a platform for publishing research results in the foundations, methods, and mechanisms, with surveillance, security, and safety tending to have been to be treated separately.

Surveillance involves relevant observation methods that aim at preventing or detecting crime behavior. Security, with its classic foundations in cryptography, involves issues related to data and privacy protection, intrusion detection, authentication, protocols, and reliable transaction, as well as other security-related fields. Safety is primarily defined as including the means designed to prevent inadvertent or hazardous operation. I am also certain that this publication will be of interest to everyone working in the general area of surveillance, security, and safety.

The journal is the result of the dedication of many individuals who are selflessly willing to put in long hours of work in an attempt to give back to the community. We owe thanks to those who completed the initial work and those who will follow through in the next few years.

WHAT IS SPECIAL IN THE FIRST ISSUE?

The journal has an impressive opening in the first issue. These papers illustrate various studies that can advance our understanding of surveillance, security, and safety when considered in a forum.

Shmidt *et al.*[1] in "Learning and unlearning from disasters: an analysis of the Virginia Tech, USA shooting and the Lion Air 610 Airline crash" provide an analysis about organizational learning and theories of learning from failures. This work attempts to stimulate organizational learning and improve organizational processes to mitigate disasters from happening again. I expect this work to continue and extend the proposed methodology to other cases in the fields of surveillance, security, and safety.

Xu *et al.*[2]'s "Big data analytics of crime prevention and control based on image processing upon cloud computing" presents a cloud computing-based image processing technology to identify individual crimes and subject segmentation, which uses statistical methods to collect the characteristics of criminal behavior, addressing the issues of non-real-time observation of criminal behavior.

In "A survey of domain name system vulnerabilities and attacks", Kim and Reeves[3] efficiently analyzed the vulnerabilities of DNS and four categories of representative DNS attacks. I expect the defense mechanisms introduced in this paper will motivate greater participation in this effort.

"Stereo storage structure assisted one-way anonymous auditing protocol in e-health system" by Jiang *et al.*[4] examines the challenges of the design of the storage structure in the cloud and the data integrity verification in the medical environment with clients' privacy protection. A novel stereo storage structure-assisted one-way anonymous auditing protocol in the e-health system is proposed, which can implement the storage and fast search of medical data.

Salmani and Barker[5]'s "Leakless Privacy-Preserving Multi-keyword Ranked Search over Encrypted Cloud Data" presents an efficient encrypted cloud data search service that employs the secure inner product similarity and chaining encryption notion and addresses the problem of leakless privacy-preserving multi-keyword ranked search over encrypted cloud data.

Future issues will contain additional papers solicited for the first issues and the latest solicited papers of the journal. These papers will continue this trend, that is the first issue begins with high hope.

WHAT IS THE SCOPE OF *JSSS*?

As an international and interdisciplinary scholarly open-access journal, Journal of Surveillance, Security and Safety publishes original research articles, reviews, and communications that offer substantially new insights into the variety of theoretical, methodological, epistemological, empirical, and practical issues reflected in the field of information security, cyber security, machine learning, emerging technologies, and their applications. Papers are solicited from, but not limited to, the following topics:

- AI-based surveillance and security
- Privacy protection based on machine learning
- Security of machine learning algorithms
- Deep learning for attack and defense
- Database security
- Data-driven cybersecurity incident prediction
- Big data security
- Cloud/fog computing security
- Outsourcing and crowdsourcing security
- Security and privacy in pervasive/ubiquitous computing
- Cyber-physical systems security
- Security, privacy, and resilience in critical infrastructures
- Multimedia security
- Wireless network security
- Social networks and IoT security
- Information hiding, forensics, and security
- Theory and applications of cryptography
- Identity management, authentication, and access control
- Security policies, models, and architectures
- Electronic commerce security
- Blockchain and finance security
- Intrusion detection
- Phishing and spam prevention
- Biometrics
- Regulation of the security industry
- Risk analysis, security measures, and management
- Evaluations of Security Measures

This journal aspires to provide a venue to support constructive communications across different related areas of security. Occasionally, Special Sections on unique topics of high interest will be organized, receiving more exposure to readers by being presented alongside relevant articles. We welcome your suggestions for special issues and appropriate Guest Editors.

I warmly welcome the participation of all researchers interested in this challenging and compelling field. This inaugural journal is the realization of the long-term desire of many in the computer industry for surveillance, security, and security issues. I hope you enjoy it and support us with the launch of this new and exciting open-access journal that will help us to shape a safer information age.

DECLARATIONS

Authors' contributions

The author contributed solely to the article.

Availability of data and materials

Not applicable.

Financial support and sponsorship

The author has not declared a specific grant for this editorial from any funding agency in the public, commercial, or not-for-profit sectors.

Conflicts of interest

The author declared that there are no conflicts of interest.

Ethical approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

REFERENCES

1. Shmidt B, Labib A, Hadleigh-Dunn S. Learning and unlearning from disasters: an analysis of the Virginia Tech, USA shooting and the Lion Air 610 Airline crash. J Surveill Secur Saf 2020;1:1-15.
2. Xu Z, Cheng C, Sugumaran V. Big data analytics of crime prevention and control based on image processing upon cloud computing. J Surveill Secur Saf 2020;1:16-33.
3. Kim TH, Reeves D. A survey of domain name system vulnerabilities and attacks. J Surveill Secur Saf 2020;1:34-60.
4. Jiang LH, Wang C, Shen J. Stereo storage structure assisted one-way anonymous auditing protocol in e-health system. J Surveill Secur Saf 2020;1:61-78.
5. Salmani K, Barker K. Leakless privacy-preserving multi-keyword ranked search over encrypted cloud data. J Surveill Secur Saf 2020;1:79-101.

AUTHOR INSTRUCTIONS

1. Submission Overview

Before you decide to publish with us, please read the following items carefully and make sure that you are well aware of Editorial Policies and the following requirements.

1.1 Topic Suitability

The topic of the manuscript must fit the scope of the journal. Please refer to Aims and Scope for more information.

1.2 Open Access and Copyright

The journal adopts Gold Open Access publishing model since its establishment and has been distributing contents under Attribution 4.0 International License since October 2017, whereas Attribution-NonCommercial-ShareAlike 3.0 Unported had been adopted by then. Please make sure that you are well aware of these policies.

1.3 Publication Fees

Authors are required to pay Article Processing Charges of 360 US Dollars after the manuscript is officially accepted. For more details, please refer to Article Processing Charges.

1.4 Language Editing

All submissions are required to be presented clearly and cohesively in good English. Authors whose first language is not English are advised to have their manuscripts checked or edited by a native English speaker before submission to ensure the high quality of expression. A well-organized manuscript in good English would make the peer review even the whole editorial handling more smooth and efficient.

If needed, authors are recommended to consider the language editing services provided by Charlesworth to ensure that the manuscript is written in correct scientific English before submission. Authors who publish with OAE journals enjoy a special discount for the services of Charlesworth via the following two ways.

Submit your manuscripts directly at http://www.charlesworthauthorservices.com/~OAE;

Open the link http://www.charlesworthauthorservices.com/, and enter Promotion Code “OAE” when you submit.

1.5 Work Funded by the National Institutes of Health

If an accepted manuscript was funded by National Institutes of Health (NIH), the author may inform editors of the NIH funding number. The editors are able to deposit the paper to the NIH Manuscript Submission System on behalf of the author.

2. Submission Preparation

2.1 Cover Letter

A cover letter is required to be submitted accompanying each manuscript. It should be concise and explain why the study is significant, why it fits the scope of the journal, and why it would be attractive to readers, *etc.*

Here is a guideline of a cover letter for authors’ consideration:

In the first paragraph: include the title and type (e.g., Original Article, Review, Case Report, *etc.*) of the manuscript, a brief on the background of the study, the question the author sought out to answer and why;

In the second paragraph: concisely explain what was done, the main findings and why they are significant;

In the third paragraph: indicate why the manuscript fits the Aims and Scope of the journal, and why it would be attractive to readers;

In the fourth paragraph: confirm that the manuscript has not been published elsewhere and not under consideration of any other journal. All authors have approved the manuscript and agreed on its submission to the journal. Journal’s specific requirements have been met if any.

If the manuscript is contributed to a special issue, please also mention it in the cover letter.

If the manuscript was presented partly or entirely in a conference, the author should clearly state the background information of the event, including the conference name, time and place in the cover letter.

2.2 Types of Manuscripts

There is no restriction on the length of manuscripts, number of figures, tables and references, provided that the manuscript is concise and comprehensive. The journal publishes Original Article, Review, Meta-Analysis, Case Report, Commentary, *etc*. For more details about paper type, please refer to the following table.

Manuscript Type	Definition	Abstract	Keywords	Main Text Structure
Original Article	An Original Article describes detailed results from novel research. All findings are extensively discussed.	Structured abstract including Aim, Methods, Results and Conclusion. No more than 250 words.	3-8 keywords	The main content should include four sections: Introduction, Methods, Results and Discussion.
Review	A Review paper summarizes the literature on previous studies. It usually does not present any new information on a subject.	Unstructured abstract. No more than 250 words.	3-8 keywords	The main text may consist of several sections with unfixed section titles. We suggest that the author includes an "Introduction" section at the beginning, several sections with unfixed titles in the middle part, and a "Conclusion" section in the end.
Case Report	A Case Report details symptoms, signs, diagnosis, treatment, and follows up an individual patient. The goal of a Case Report is to make other researchers aware of the possibility that a specific phenomenon might occur.	Unstructured abstract. No more than 150 words.	3-8 keywords	The main text consists of three sections with fixed section titles: Introduction, Case Report, and Discussion.
Meta-Analysis	A Meta-Analysis is a statistical analysis combining the results of multiple scientific studies. It is often an overview of clinical trials.	Structured abstract including Aim, Methods, Results and Conclusion. No more than 250 words.	3-8 keywords	The main content should include four sections: Introduction, Methods, Results and Discussion.
Systematic Review	A Systematic Review collects and critically analyzes multiple research studies, using methods selected before one or more research questions are formulated, and then finding and analyzing related studies and answering those questions in a structured methodology.	Structured abstract including Aim, Methods, Results and Conclusion. No more than 250 words.	3-8 keywords	The main content should include four sections: Introduction, Methods, Results and Discussion.
Technical Note	A Technical Note is a short article giving a brief description of a specific development, technique or procedure, or it may describe a modification of an existing technique, procedure or device applied in research.	Unstructured abstract. No more than 250 words.	3-8 keywords	/
Commentary	A Commentary is to provide comments on a newly published article or an alternative viewpoint on a certain topic.	Unstructured abstract. No more than 250 words.	3-8 keywords	/
Editorial	An Editorial is a short article describing news about the journal or opinions of senior editors or the publisher.	None required	None required	/
Letter to Editor	A Letter to Editor is usually an open post-publication review of a paper from its readers, often critical of some aspect of a published paper. Controversial papers often attract numerous Letters to Editor	Unstructured abstract (optional). No more than 250 words.	3-8 keywords (optional)	/
Opinion	An Opinion usually presents personal thoughts, beliefs, or feelings on a topic.	Unstructured abstract (optional). No more than 250 words.	3-8 keywords	/
Perspective	A Perspective provides personal points of view on the state-of-the-art of a specific area of knowledge and its future prospects. Links to areas of intense current research focus can also be made. The emphasis should be on a personal assessment rather than a comprehensive, critical review. However, comments should be put into the context of existing literature. Perspectives are usually invited by the Editors.	Unstructured abstract. No more than 150 words.	3-8 keywords	/

2.3 Manuscript Structure

2.3.1 Front Matter

2.3.1.1 Title

The title of the manuscript should be concise, specific and relevant, with no more than 16 words if possible. When gene or protein names are included, the abbreviated name rather than full name should be used.

2.3.1.2 Authors and Affiliations

Authors' full names should be listed. The initials of middle names can be provided. Institutional addresses and email addresses for all authors should be listed. At least one author should be designated as corresponding author. In addition, corresponding authors are suggested to provide their Open Researcher and Contributor ID upon submission. Please note that any change to authorship is not allowed after manuscript acceptance.

2.3.1.3 Abstract

The abstract should be a single paragraph with word limitation and specific structure requirements (for more details please refer to Types of Manuscripts). It usually describes the main objective(s) of the study, explains how the study was done, including any model organisms used, without methodological detail, and summarizes the most important results and their significance. The abstract must be an objective representation of the study: it is not allowed to contain results which are not presented and substantiated in the manuscript, or exaggerate the main conclusions. Citations should not be included in the abstract.

2.3.1.4 Keywords

Three to eight keywords should be provided, which are specific to the article, yet reasonably common within the subject discipline.

2.3.2 Main Text

Manuscripts of different types are structured with different sections of content. Please refer to Types of Manuscripts to make sure which sections should be included in the manuscripts.

2.3.2.1 Introduction

The introduction should contain background that puts the manuscript into context, allow readers to understand why the study is important, include a brief review of key literature, and conclude with a brief statement of the overall aim of the work and a comment about whether that aim was achieved. Relevant controversies or disagreements in the field should be introduced as well.

2.3.2.2 Methods

Methods should contain sufficient details to allow others to fully replicate the study. New methods and protocols should be described in detail while well-established methods can be briefly described or appropriately cited. Experimental participants selected, the drugs and chemicals used, the statistical methods taken, and the computer software used should be identified precisely. Statistical terms, abbreviations, and all symbols used should be defined clearly. Protocol documents for clinical trials, observational studies, and other non-laboratory investigations may be uploaded as supplementary materials.

2.3.2.3 Results

This section contains the findings of the study. Results of statistical analysis should also be included either as text or as tables or figures if appropriate. Authors should emphasize and summarize only the most important observations. Data on all primary and secondary outcomes identified in the section Methods should also be provided. Extra or supplementary materials and technical details can be placed in supplementary documents.

2.3.2.4 Discussion

This section should discuss the implications of the findings in context of existing research and highlight limitations of the study. Future research directions may also be mentioned.

2.3.2.5 Conclusion

It should state clearly the main conclusions and include the explanation of their relevance or importance to the field.

2.3.3 Back Matter

2.3.3.1 Acknowledgments

Anyone who contributed towards the article but does not meet the criteria for authorship, including those who provided professional writing services or materials, should be acknowledged. Authors should obtain permission to acknowledge from all those mentioned in the Acknowledgments section. This section is not added if the author does not have anyone to acknowledge.

2.3.3.2 Authors' Contributions

Each author is expected to have made substantial contributions to the conception or design of the work, or the acquisition, analysis, or interpretation of data, or the creation of new software used in the work, or have drafted the work or substantively revised it.

Please use Surname and Initial of Forename to refer to an author's contribution. For example: made substantial contributions to conception and design of the study and performed data analysis and interpretation: Salas H, Castaneda WV; performed data acquisition, as well as provided administrative, technical, and material support: Castillo N, Young V.

If an article is single-authored, please include "The author contributed solely to the article." in this section.

2.3.3.3 Availability of Data and Materials

In order to maintain the integrity, transparency and reproducibility of research records, authors should include this section in their manuscripts, detailing where the data supporting their findings can be found. Data can be deposited into data repositories or published as supplementary information in the journal. Authors who cannot share their data should state that the data will not be shared and explain it. If a manuscript does not involve such issue, please state "Not applicable." in this section.

2.3.3.4 Financial Support and Sponsorship

All sources of funding for the study reported should be declared. The role of the funding body in the experiment design, collection, analysis and interpretation of data, and writing of the manuscript should be declared. Any relevant grant numbers and the link of funder's website should be provided if any. If the study is not involved with this issue, state "None." in this section.

2.3.3.5 Conflicts of Interest

Authors must declare any potential conflicts of interest that may be perceived as inappropriately influencing the representation or interpretation of reported research results. If there are no conflicts of interest, please state "All authors declared that there are no conflicts of interest." in this section. Some authors may be bound by confidentiality agreements. In such cases, in place of itemized disclosures, we will require authors to state "All authors declare that they are bound by confidentiality agreements that prevent them from disclosing their conflicts of interest in this work.". If authors are unsure whether conflicts of interest exist, please refer to the "Conflicts of Interest" of OAE Editorial Policies for a full explanation.

2.3.3.6 Ethical Approval and Consent to Participate

Research involving human subjects, human material or human data must be performed in accordance with the Declaration of Helsinki and approved by an appropriate ethics committee. An informed consent to participate in the study should also be obtained from participants, or their parents or legal guardians for children under 16. A statement detailing the name of the ethics committee (including the reference number where appropriate) and the informed consent obtained must appear in the manuscripts reporting such research.

Studies involving animals and cell lines must include a statement on ethical approval. More information is available at Editorial Policies.

If the manuscript does not involve such issue, please state "Not applicable." in this section.

2.3.3.7 Consent for Publication

Manuscripts containing individual details, images or videos, must obtain consent for publication from that person, or in the case of children, their parents or legal guardians. If the person has died, consent for publication must be obtained from the next of kin of the participant. Manuscripts must include a statement that a written informed consent for publication was obtained. Authors do not have to submit such content accompanying the manuscript. However, these documents must be available if requested. If the manuscript does not involve this issue, state "Not applicable." in this section.

2.3.3.8 Copyright

Authors retain copyright of their works through a Creative Commons Attribution 4.0 International License that clearly states how readers can copy, distribute, and use their attributed research, free of charge. A declaration "© The Author(s) 2020." will be added to each article. Authors are required to sign License to Publish before formal publication.

2.3.3.9 References

References should be numbered in order of appearance at the end of manuscripts. In the text, reference numbers should be placed in square brackets and the corresponding references are cited thereafter. Only the first five authors' names are required to be listed in the references, other authors' names should be omitted and replaced with "et al.". Abbreviations of the journals should be provided on the basis of Index Medicus. Information from manuscripts accepted but not published should be cited in the text as "Unpublished material" with written permission from the source.

References should be described as follows, depending on the types of works:

Types	Examples
Journal articles by individual authors	Weaver DL, Ashikaga T, Krag DN, Skelly JM, Anderson SJ, et al. Effect of occult metastases on survival in node-negative breast cancer. N Engl J Med 2011;364:412-21. [PMID: 21247310 DOI: 10.1056/NEJMoa1008108]
Organization as author	Diabetes Prevention Program Research Group. Hypertension, insulin, and proinsulin in participants with impaired glucose tolerance. Hypertension 2002;40:679-86. [PMID: 12411462]
Both personal authors and organization as author	Vallancien G, Emberton M, Harving N, van Moorselaar RJ; Alf-One Study Group. Sexual dysfunction in 1,274 European men suffering from lower urinary tract symptoms. J Urol 2003;169:2257-61. [PMID: 12771764 DOI: 10.1097/01.ju.0000067940.76090.73]
Journal articles not in English	Zhang X, Xiong H, Ji TY, Zhang YH, Wang Y. Case report of anti-N-methyl-D-aspartate receptor encephalitis in child. J Appl Clin Pediatr 2012;27:1903-7. (in Chinese)
Journal articles ahead of print	Odibo AO. Falling stillbirth and neonatal mortality rates in twin gestation: not a reason for complacency. BJOG 2018; Epub ahead of print [PMID: 30461178 DOI: 10.1111/1471-0528.15541]
Books	Sherlock S, Dooley J. Diseases of the liver and billiary system. 9th ed. Oxford: Blackwell Sci Pub; 1993. pp. 258-96.
Book chapters	Meltzer PS, Kallioniemi A, Trent JM. Chromosome alterations in human solid tumors. In: Vogelstein B, Kinzler KW, editors. The genetic basis of human cancer. New York: McGraw-Hill; 2002. pp. 93-113.
Online resource	FDA News Release. FDA approval brings first gene therapy to the United States. Available from: https://www.fda.gov/NewsEvents/Newsroom/PressAnnouncements/ucm574058.htm. [Last accessed on 30 Oct 2017]
Conference proceedings	Harnden P, Joffe JK, Jones WG, editors. Germ cell tumours V. Proceedings of the 5th Germ Cell Tumour Conference; 2001 Sep 13-15; Leeds, UK. New York: Springer; 2002.
Conference paper	Christensen S, Oppacher F. An analysis of Koza's computational effort statistic for genetic programming. In: Foster JA, Lutton E, Miller J, Ryan C, Tettamanzi AG, editors. Genetic programming. EuroGP 2002: Proceedings of the 5th European Conference on Genetic Programming; 2002 Apr 3-5; Kinsdale, Ireland. Berlin: Springer; 2002. pp. 182-91.
Unpublished material	Tian D, Araki H, Stahl E, Bergelson J, Kreitman M. Signature of balancing selection in Arabidopsis. Proc Natl Acad Sci U S A. Forthcoming 2002.

For other types of references, please refer to U.S. National Library of Medicine.
The journal also recommends that authors prepare references with a bibliography software package, such as EndNote to avoid typing mistakes and duplicated references.

2.3.3.10 Supplementary Materials

Additional data and information can be uploaded as Supplementary Material to accompany the manuscripts. The supplementary materials will also be available to the referees as part of the peer-review process. Any file format is acceptable, such as data sheet (word, excel, csv, cdx, fasta, pdf or zip files), presentation (powerpoint, pdf or zip files), image (cdx, eps, jpeg, pdf, png or tiff), table (word, excel, csv or pdf), audio (mp3, wav or wma) or video (avi, divx, flv, mov, mp4, mpeg, mpg or wmv). All information should be clearly presented. Supplementary materials should be cited in the main text in numeric order (e.g., Supplementary Figure 1, Supplementary Figure 2, Supplementary Table 1, Supplementary Table 2, *etc.*). The style of supplementary figures or tables complies with the same requirements on figures or tables in main text. Videos and audios should be prepared in English, and limited to a size of 500 MB or a duration of 3 minutes.

2.4 Manuscript Format

2.4.1 File Format

Manuscript files can be in DOC and DOCX formats and should not be locked or protected.

2.4.2 Length

There are no restrictions on paper length, number of figures, or amount of supporting documents. Authors are encouraged to present and discuss their findings concisely.

2.4.3 Language

Manuscripts must be written in English.

2.4.4 Multimedia Files

The journal supports manuscripts with multimedia files. The requirements are listed as follows:
Videos or audio files are only acceptable in English. The presentation and introduction should be easy to understand. The frames should be clear, and the speech speed should be moderate.
A brief overview of the video or audio files should be given in the manuscript text.
The video or audio files should be limited to a duration of 3 min and a size of up to 500 MB.

Please use professional software to produce high-quality video files, to facilitate acceptance and publication along with the submitted article. Upload the videos in mp4, wmv, or rm format (preferably mp4) and audio files in mp3 or wav format.

2.4.5 Figures

Figures should be cited in numeric order (e.g., Figure 1, Figure 2) and placed after the paragraph where it is first cited;
Figures can be submitted in format of tiff, psd, AI or jpeg, with resolution of 300-600 dpi;
Figure caption is placed under the Figure;
Diagrams with describing words (including, flow chart, coordinate diagram, bar chart, line chart, and scatter diagram, *etc*.) should be editable in word, excel or powerpoint format. Non-English information should be avoided;
Labels, numbers, letters, arrows, and symbols in figure should be clear, of uniform size, and contrast with the background;
Symbols, arrows, numbers, or letters used to identify parts of the illustrations must be identified and explained in the legend;
Internal scale (magnification) should be explained and the staining method in photomicrographs should be identified;
All non-standard abbreviations should be explained in the legend;
Permission for use of copyrighted materials from other sources, including re-published, adapted, modified, or partial figures and images from the internet, must be obtained. It is authors' responsibility to acquire the licenses, to follow any citation instruction requested by third-party rights holders, and cover any supplementary charges.

2.4.6 Tables

Tables should be cited in numeric order and placed after the paragraph where it is first cited;
The table caption should be placed above the table and labeled sequentially (e.g., Table 1, Table 2);
Tables should be provided in editable form like DOC or DOCX format (picture is not allowed);
Abbreviations and symbols used in table should be explained in footnote;
Explanatory matter should also be placed in footnotes;
Permission for use of copyrighted materials from other sources, including re-published, adapted, modified, or partial tables from the internet, must be obtained. It is authors' responsibility to acquire the licenses, to follow any citation instruction requested by third-party rights holders, and cover any supplementary charges.

2.4.7 Abbreviations

Abbreviations should be defined upon first appearance in the abstract, main text, and in figure or table captions and used consistently thereafter. Non-standard abbreviations are not allowed unless they appear at least three times in the text. Commonly-used abbreviations, such as DNA, RNA, ATP, *etc*., can be used directly without definition. Abbreviations in titles and keywords should be avoided, except for the ones which are widely used.

2.4.8 Italics

General italic words like *vs*., *et al., etc*., *in vivo*, *in vitro*; *t* test, *F* test, *U* test; related coefficient as *r*, sample number as *n*, and probability as *P*; names of genes; names of bacteria and biology species in Latin.

2.4.9 Units

SI Units should be used. Imperial, US customary and other units should be converted to SI units whenever possible. There is a space between the number and the unit (i.e., 23 mL). Hour, minute, second should be written as h, min, s.

2.4.10 Numbers

Numbers appearing at the beginning of sentences should be expressed in English. When there are two or more numbers in a paragraph, they should be expressed as Arabic numerals; when there is only one number in a paragraph, number < 10 should be expressed in English and number > 10 should be expressed as Arabic numerals. 12345678 should be written as 12,345,678.

2.4.11 Equations

Equations should be editable and not appear in a picture format. Authors are advised to use either the Microsoft Equation Editor or the MathType for display and inline equations.

2.5 Submission Link

Submit an article via http://www.oaemesas.com/jsss.

OAE Publishing Inc.
www.oaepublish.com

Journal of Surveillance, Security and Safety
(*JSSS*)
Los Angeles Office
245 E Main Street ste122, Alhambra,
CA 91801, USA
Tel: +1 323 9987086
E-mail: editorialoffice@jsssjournal.net
Website: www.jsssjournal.com