Towards an Applicability of Current Network Forensics for Cloud Networks: *A SWOT Analysis*

Suleman Khan[a], Abdullah Gani[a], Ainuddin Wahid Abdul Wahab[a], Salman Iqbal[a], Ahmed Abdelaziz[a], Abdel Muttlib[a], Omar Adil Mahdi[a], Muhammad Shiraz[b], Yusor Rafid Bahar Al-Mayouf[c], Muhammad Khurram Khan[d], Victor Chang[e],

[a] Center for Mobile Cloud Computing Research (C4MCCR), University of Malaya, Kuala Lumpur, Malaysia

[b] Department of Computer Science, Federal Urdu University of Arts, Science and Technology, Pakistan

[c] Department of Electrical, Electronic and Systems Engineering, Universiti Kebangsaan Malaysia, 43600 Bangi, Selangor, Malaysia

[d] Center of Excellence in Information Assurance (CoEIA), King Saud University, Saudi Arabia

[e] IBSS, Xi'an Jiaotong Liverpool University, Suzhou, China

Abstract

In recent years, the migration of the computational workload to computational clouds has attracted intruders to target and exploit cloud networks internally and externally. The investigation of such hazardous network attacks in the cloud network requires comprehensive network forensics methods (NFM) to identify the source of the attack. However, cloud computing lacks NFM to identify the network attacks that affect various cloud resources by disseminating through cloud networks. In this paper, the study is motivated by the need to find the applicability of current network forensics methods (C-NFMs) for cloud networks of the cloud computing. The applicability is evaluated based on strengths, weaknesses, opportunities, and threats to outlook the cloud network. To the best of our knowledge, no research to date has been conducted to assist network forensics investigators and cloud service providers in finding an optimal method for investigation of network vulnerabilities found in cloud networks. To this end and in this study, the state-of-the-art C-NFMs are classified and analyzed based on the cloud network perspective using *SWOT-analysis.* It implies that C-NFMs have a suitable impact on cloud network which further requires for reformation to ensure its applicability in cloud networks.

Keywords: Cloud security, Cloud investigation, Network forensics, Network Security

1. Introduction

The technological advancements in communication and network have emerged as distributed, integrated, and virtualized resources to assist users with additional computing and storage resources in the form of cloud computing (Moeller, 2014). Cloud computing is the fastest growing distributed computational platform in enterprises, industries, academics, and research community today (Baldwin, Pym, & Shiu, 2013). In general, cloud computing is a connected

resource through various distributed networks (Erl, Puttini, & Mahmood, 2013). The network is a crucial part in cloud computing by providing quality of service such as ensuring the time constraints (Choy, Wong, Simon, & Rosenberg, 2012) and without it, cloud computing is unable to integrate various computation and storage resources (Mell & Grance, 2011). Such quality is considered one of the main features of cloud computing to transparently execute user applications and send its result back (Farah, 2013). The network plays two important roles in cloud computing which are connecting the user application to the appropriate resource on the cloud (Gong, Liu, Zhang, Chen, & Gong, 2010) and sending the output to the user when the application is executed. Nevertheless, the significance of networks in cloud computing has drawn intruders to attack cloud networks through malicious attacks (Hassan, Bourgeois, Sunderam, & Li, 2012; Khan, Ahmad, et al., 2014; Xiong et al., 2014) such as illegal access, insertion of malicious code, modification of packets, eavesdropping and sending invalid packets. These malicious attacks will affect the user applications and cloud resources to delay the execution process of the entire cloud computing application.

Therefore, efficient and effective investigation process is required for analyzing cloud networks to protect cloud computing from various malicious attacks (Khan, Shiraz, et al., 2014). However, a prevention strategy from malicious attack will require a proper investigation method to analyze network and determine the root cause of the attack (Diamah, Mohammadian, & Balachandran, 2012). C-NFM is paramount in identifying network attacks through investigation of network packets, logs, application, and various network events (Almulhem, 2009). The C-NFM is adopted by various network administrators, network security officers, and network forensics investigators to assist in the network forensics process. Network forensics is a process to identify, collect, preserve, analyze, and report the digital evidence from the network and investigate the source of the attack (Khan, Gani, Wahab, Shiraz, & Ahmad, 2016; Nguyen, Tran, Ma, & Sharma, 2014). Different methods are proposed by researchers to stop and reduce network vulnerabilities found in the network (L. M. Chen, Chen, Liao, & Sun, 2013; Cohen, 2008; Jeong & Lee, 2013; Pilli, Joshi, & Niyogi, 2010; Shimeall & Spring, 2014; Taylor, Haggerty, Gresty, Almond, & Berry, 2014). However, in cloud computing, C-NFMs are lacking due to various constraints faced by different network forensics investigators (NFIs) (Birk & Wegener, 2011) including unavailability of cloud networks, abundance of network links and devices, network virtualization, volatility of the network data, high bandwidth, fast moving network data, heterogeneity, jurisdiction and multi-tenancy. One of the solutions to overcome these constraints is to develop new methods and ideas to assist NFIs in investigating criminal crimes happening in cloud computing. To do so, researchers are facing problems; intruders are exploiting cloud networks with different attacks so frequently that make it difficult to cope with the novelty of attacks in such situation.

To provide an appropriate solution for such an overwhelming situation, a sound approach is to apply C-NFM for the cloud network rather than re-inventing the wheel again. To adopt such

a strategy, one has to be clear about C-NFM, especially its applicability for cloud network. In this context, the strengths, weaknesses, opportunities, and threats (SWOT) analysis of C-NFM in the perspective of cloud networking is essential. This study is considered unique in this area which helps NFIs and assists cloud service providers (CSPs) in controlling, detecting, and analyzing various network vulnerabilities generated by intruders to exploit cloud resources illegitimately.

The contributions of this paper are organized as follows:

a) A comprehensive discussion and classification of current network forensics methods.
b) A feasibility study on the applicability of C-NFM for cloud networks based on SWOT analysis.
c) Significant discussion with valuable recommendations.

The paper consists of the following sections: Section 2 introduces the useful information about digital forensics, network forensics, cloud computing, and SWOT analysis; Section 3 provides the classification of C-NFM and a brief description is provided. Section 4 describes C-NFM strengths, weaknesses, opportunities, and threats in the perspective of cloud network forensics. Section 5 presents a discussion and recommendation based on the outcomes of Section 4, and finally, Section 6 concludes the paper.

2. Background

2.1 Digital Forensics

At the time when technology is growing rapidly, digital communication plays a vital role in information technology (Madhow, 2008). Digital communication remains an important medium of information exchange. The importance is not to be undermined by intruders performing malicious attacks on the content of digital communication (Schneier, 2011). Frequently, attacks are performed on digital devices by exploiting the data traveling in communication channels. Therefore, investigation of the attacks is crucial to identify the source (Katiravan, Chellappan, & Rejula, 2012). Digital forensics investigate the attacks by accessing the physical of digital devices to search for traces left by the intruders (Kohn, Eloff, & Eloff, 2013). These traces are the evidence against intruders that helps in regenerating the attack again (Geethakumari & Belorkar, 2012).

Digital forensic is defined as *"a process of analyzing digital devices for various traces left by intruders during performing their attacks"*. In the literature, the digital forensic process is explained in steps to isolate the whole process at different levels (Beebe & Clark, 2005; Carrier & Spafford, 2004; Palmer, 2001; Reith, Carr, & Gunsch, 2002). The National Institute of Standard and Technology (NIST) explains the digital forensic process in four steps including collection, examination, analysis, and reporting (Dang;, August 2006). In the collection step, digital evidence is identified while keeping in view of what, where, when and how are the evidence is collected.

The examination step preserves the collected evidence from any modification. The data of the evidence are usually stored and saved in a secure persistent memory that needs to be retrieved in the future for analysis. In the analysis step, the evidence is analyzed to determine susceptibilities performed by intruders. The identified evidence helps the forensic investigators to reach the source of the attacks by re-generating the attack for investigation purpose. In the final step, a complete legal document is generated to report each step performed in the forensic investigation. This legal document is presented in the court as evidence against the intruder.

2.2 Network Forensics

The communication network is a combination of network devices, channels, and protocols for data transmission. The intruders attack the networks and leak confidential information from the networks (Ritchey & Ammann, 2000). Besides, intruders perform various malicious attacks such as DoS (Carl, Kesidis, Brooks, & Rai, 2006), DDoS (Srivastava, Gupta, Tyagi, Sharma, & Mishra, 2011), IP address spoofing (Ferguson, 2000), password based attacks (Raza, Iqbal, Sharif, & Haider, 2012), man-in-the-middle (Desmedt, 2011) and data modification (Schluessler, Goglin, & Johnson, 2007). These attacks divert the system from its normal track to cause the system to produce malfunction outputs.

To overcome such network attacks, C-NFMs are used to identify the root cause of the attacks by reaching its source (Diamah et al., 2012). In general, network forensics identify the malicious attacks by targeting the network traffic as the main source of exploitation by intruders for performing attacks (Khan, Gani, Wahab, & Bagiwa, 2015). Network traffic is stored as network logs that record each network event performed during the network traffic flow (Shanmugasundaram, Memon, Savant, & Bronnimann, 2003). The network logs are stored in a persistent storage that is retrieved later for network analysis (Khan, Gani, Wahab, Shiraz, Mustapha, et al., 2016). However, it is a very challenging task to store all the network traffic in the network logs due to the lack of storage resources (L. Chen, Li, Gao, & Liu, 2009; Fen, Xinchun, & Hao, 2012; H. S. Kim & Kim, 2011). This problem can be solved by using packet marking (Akyuz & Sogukpinar, 2009; X.-J. Wang & Wang, 2010; Yonghui, Yulong, Fangchun, Sen, & Dong, 2010) which mostly used in traceback techniques (Bi, Deng, Xu, Shi, & Hu, 2013; Fen, Hui, Shuang-shuang, & Xin-chun, 2012; Jeong & Lee, 2013; Yu et al., 2013). This technique marks one or more fields of the packet to identify modification in the packet during its communication flow and usually performed at the router level (Fen, Hui, et al., 2012; H. S. Kim & Kim, 2011; Ren & Jin, 2005). Packet marking minimizes the storage problem, however, it is time-consuming and computationally intensive. Other methods adopted to investigate network attacks include Bayesian networks (Kruegel, Mutz, Robertson, & Valeur, 2003), intrusion detection systems (Cusack & Alqahtani, 2013), attack graphs (Zhang, Wang, & Kadobayashi, 2012), distributed systems (Ren & Jin, 2005), honeypots (Li & Schmitz, 2009), machine learning (Callado, Kelner,

Sadok, Alberto Kamienski, & Fernandes, 2010), and data mining (Xu et al., 2013). The main objective of C-NFM is to identify network attacks by searching for evidence on the network in a legal way.

2.3 Cloud Computing

Cloud computing is a combination of resources to deliver services to users transparently and efficiently in terms of computation and storage (Gani et al., 2014; Shiraz, Gani, Shamim, Khan, & Ahmad, 2015). The resources in the cloud, owned or rented out by cloud service providers (CSP) are integrated together to strengthen the feature of computation and storage (Buyya, Yeo, & Venugopal, 2008; Shamsi, Khojaye, & Qasmi, 2013). Users access the cloud resources without having in-depth knowledge or details of its location and ownership. The user only pays for the resources they have used in cloud computing which is known as pay-as-you-go (Armbrust et al., 2010). A single resource can be used by many users to increase efficiency, throughput, and reduce the idle time of the resource in cloud computing. The hypervisor is used to create virtualization in cloud computing to isolate various users on the same resource (Shiraz, Abolfazli, Sanaei, & Gani, 2013).

Nowadays there are a number of CSPs providing different services to millions of users based on their needs, for instance, Microsoft, Amazon, Azure, Google, and others. These CSPs are categorized into three main service categories: Infrastructure-as-a-service (IaaS), (b) Platform-as-a-service (PaaS), and (c) Software-as-a-service (SaaS) (Armbrust et al., 2010). In IaaS model, a user is given access to the virtual resources of cloud computing for executing their application while responsible for its application in terms of security, maintenance, and support itself (Mell & Grance, 2011). The examples include Amazon Web Service (AWS), Google Compute Engine (GCE), Rackspace, and Window Azure. The PaaS model is used by developers to develop new applications on the infrastructure provided by CSPs. In PaaS, CSP assists programmers/developers by providing open/proprietary languages, the initial basic configuration for communication, monitoring, distributing the application, and scalability of an application (Buyya et al., 2008). The examples of PaaS include AWS Elastic Beanstalk, Force.com, Apprenda, and Heroku. However, in SaaS, CSPs provide complete software to its user for execution. The software/application is accessed through a web portal or service oriented architecture (Buyya, Yeo, Venugopal, Broberg, & Brandic, 2009). Users can access any software listed by CSP without concern about its configuration and installation. The examples of SaaS include Google apps, Gmail, Microsoft 365, Salesforce, and Cisco WebEx. Table 1 lists the examples of each service model with its description.

Table 1: Descriptions of cloud service models

Cloud Service Model	Examples	Descriptions

Cloud Service Model	Examples	Descriptions
IaaS	Amazon EC2	It provides computing facility to the user with pay for what you used.
	AT&T	Provides seamless connectivity to high computational resources with virtualized infrastructure.
	Bluelock	Provides secure data center computing with backend withVMware vCloud Datacenter Services
	Google Compute Engine (GCE)	Provides high-performance Virtual Machine and global load balancing on Google infrastructure.
	Verizon	Provides customized computing infrastructure with flexible services.
PaaS	AWS Elastic BeanStalk	Provides handling facility with capacity provisioning, auto-scaling, configuration details, and apps monitoring.
	Appistry	The CloudIQ platform assists developers in terms of scalability, manageability, reliability, and monitoring.
	AppScale	Its open source environment helps developers to implement and monitor their apps with profile and debug facilities.
	Salesforce.com	It assists in developing multitenant applications on the cloud.
	Google Apps Engine	Easy and simplest platform help developers to simply develop their web apps on google infrastructure.
SaaS	Abiquo	Integrating business policies to manage their enterprise resources easily.
	AccelOps	Discover, analyze, and automates IT issues in resources, data, network, servers, and security applications.
	Akamai	Provides optimization in terms of availability, security, and deliverables.
	Oracle's on-demand CRM	Facilitate customers to adopt its software according to their need and enterprise environment.
	Cumulux	Enhance business values in terms of expansion, strategize and operationalize cloud applications.

The aforementioned cloud service models cannot provide their services without the connectivity of cloud networks. The cloud network connects cloud resources to provide the services to users connected to the cloud. The cloud network can be divided into two main categories including data center network and inter-cloud network. Each cloud consists of many data centers, which are connected together to strengthen its services by combining their resources (Qi, Shiraz, Gani, Whaiduzzaman, & Khan, 2014). Different data centers connect to each other through a specific data center network called inter-data center network. However, each data center is buildup from a group of clusters connected to each other through the data center network known as an intra-data center network. Similarly, one cloud is connected to others through inter-cloud networks. Consequently, each of the cloud networks plays a vital role in executing user applications by sending and receiving the data on a timely basis. In this paper, the term of *cloud network* refers as both types of cloud networks i.e. data center network and inter-cloud network. In Figure-1, each cloud network is highlighted with different colors to easily

differentiate between them for understandability. The red dotted color lines in the figure-1 show the intra-data center network that connects groups of clusters into a single data center. The blue rod shape surrounded by black color indicates the inter-data center network that connects two or more data centers into a single cloud. The black double-headed arrows called inter-cloud networks connects two or more clouds among each other.

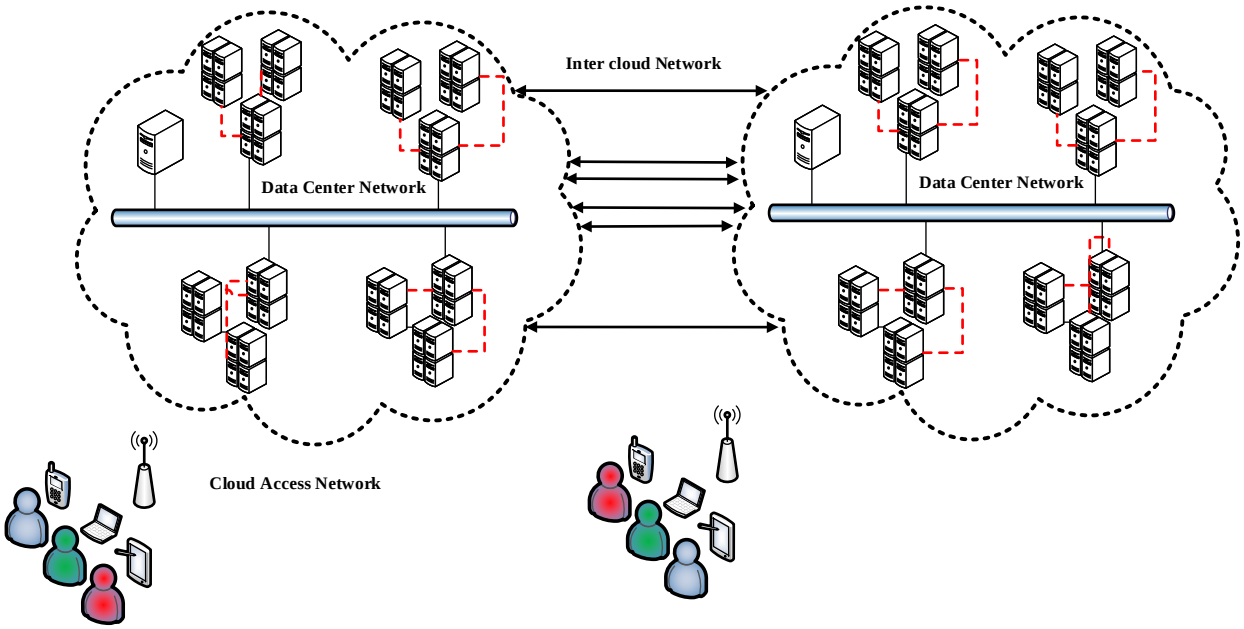

Figure 1: Identification of cloud network types in cloud computing

2.4 Strengths, Weaknesses, Opportunities, and Threats (SWOT)

SWOT analysis also known as SWOT matrix is a tool to assess the strengths, weaknesses, opportunities, and threats of a product, project, or a business (Jackson, Joshi, & Erhardt, 2003) as shown in Figure 2. However, it is not limited to the aforementioned entities as SWOT matrix can be applied in any situation to obtain an overall picture of the situation. A SWOT matrix is a 2x2 matrix and its main objective is to identify the internal factors (strengths and weaknesses) and also external factors including opportunities and threats of an entity for such analysis is performed (Helms & Nixon, 2010), to achieve the desired objective. In the literature, SWOT matrix is used to identify the internal and external factors that are favorable or unfavorable to achieve a specific objective (Chermack & Kasshanna, 2007; Shinno, Yoshioka, Marpaung, & Hachiga, 2006; Vonk, Geertman, & Schot, 2007). However, in this paper, we have conducted a SWOT analysis for C-NFM to study its applicability in the investigation of the cloud network. Our study highlights SWOT of C-NFM and provides a guide for the required modification in a cloud network. As a result, it will reduce the time and effort of CSPs and NFIs in developing new

methods for cloud networks. A comprehensive description of SWOT analysis for each C-NFM is presented in Section 4.

The C-NFMs have potential features that can be applied in cloud networks, whereas it also requires additional efforts to make it fully applicable. Moreover, C-NFMs have opportunities for cloud networks by modifying its current state or integrating it with new methods. However, C-NFM has several limitations that need to overcome. Threats to C-NFM have to be considered in terms of cloud networks, which might reduce the performance and increase the false positive rate. For a full applicability of C-NFM in cloud networks, SWOT matrix provides an overview to reduce its weaknesses by availing opportunities and preventing it against threats through reinforcement of its strengths.

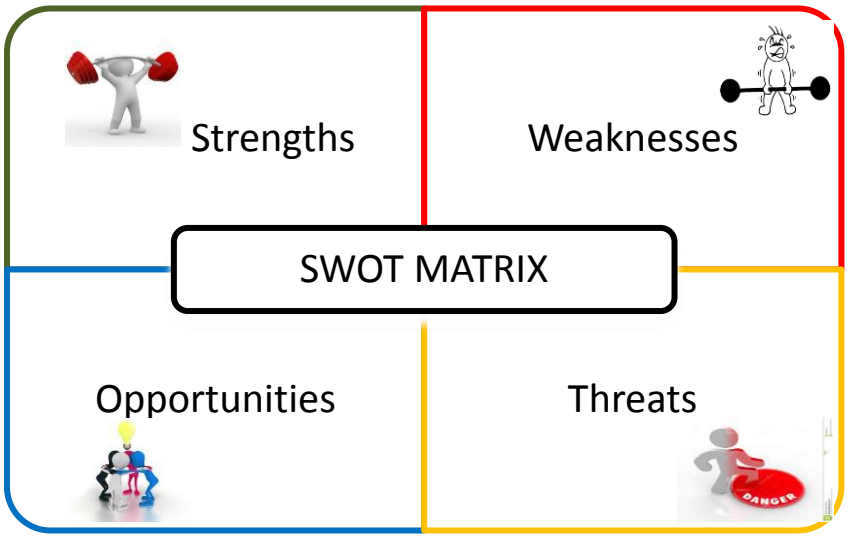

Figure 2: SWOT Matrix

3. Network Forensics Methods

Generally, the investigation is initiated when a criminal activity is detected to restrain the activity. Similarly, an investigation process known as network forensics is conducted in the network after an attack is detected. The objective of network forensics is to identify the root cause of the attack, so it can be tolerated in the future (Zhang et al., 2012). This process is performed by network forensics method (NFM) (Pilli et al., 2010). Different C-NFMs are used with varying objectives such as identification of the source (Akyuz & Sogukpinar, 2009), worst attack (Diamah et al., 2012), malicious code (A. C. Kim, Park, & Lee, 2013), pattern matching (I.-L. Lin, Yen, Wu, & Wang, 2010; Pelaez & Fernandez, 2009), evidence collection (Liu, Singhal, & Wijesekera, 2012), reconstruction of attacks (Ibrahim, Abdullah, & Dehghantanha, 2012), determining the origin of the attack (Fen, Hui, et al., 2012; Jeong & Lee, 2013) and others (Jemili, Zaghdoud, & Ben Ahmed,

2007; Ren & Jin, 2005). Based on SWOT analysis with reference to cloud networks, the C-NFM is classified into four categories which are a) intrusion detection systems, b) traceback, c) distributive, and d) attack graphs. A brief description of these categories is explained in the Table-2.

Table 2: Network forensics methods classification

Classification	Description
Intrusion Detection Systems	The system provides protection and monitoring to the network infrastructure by detecting anomalies entering to the system via a network. It helps in investigating various network anomalies by knowing various patterns, attack time, packet modification, in-depth packet inspection, more and more alerts to forensic investigators.
Traceback	After the network attack, a replay of its sequential steps again for investigating the effectiveness of the attack is considered as traceback techniques. Mostly packet marking mechanism is used to perform traceback while it helps forensic investigators to reach the origin of the attack.
Distributive	The collection of malicious traffic through distributive installed agent or analysis of malicious traffic in various distributed location of the network is considered as distribution in network forensics. It assists forensic investigators to analyze malicious traffic at various part of the network by reducing workload and response time respectively.
Attack Graphs	The visualization of attack paths in networks by facilitating forensic investigators to easily investigate the malicious behavior of the intruder. It reduces the investigating time and helps in reaching the source of the attack especially in large distributed networks.

Each C-NFM in each category is assigned with alpha-numeric code for easy reference in this paper. For instance, code 'I' is assigned to intrusion detection system with its sequential number such as I-1, I-2, I-3 for IDS C-NFMs, code 'T' is assigned for traceback methods, code 'D' is assigned for distributive methods and code 'An' is assigned for attack graph. In Table-3, each C-NFM is explained in terms of its technique, approach, and objective while making its mechanism easy to understand.

3.1 C-NFM based on Intrusion Detection System

Intrusion detection system (IDS) assists the forensic process by detecting network attacks and inform forensic modules in the system to analyze the malicious traffic. In (Sy, 2009), the analytical intrusion of the detection method is proposed based on probabilistic and inference mechanism. The method detects intrusion alerts from distributed IDS sensor installed at various places in the network. It also detects intrusion through an unknown signature rule. The proposed solution uses inhibiting negative behavior to detect intrusion while focusing on network-based IDS. In (L. Chen et al., 2009), fault tolerance of forensic server is proposed by maintaining its availability even during the attack. Two forensic servers are used to separate and store the normal and malicious traffic identified by the firewall placed before each of the servers. Numerical analysis is used to measure the availability of the forensic servers. The tolerance of forensic servers is achieved when one of the servers is down and its load is transferred to the next server. The

incoming traffic is evaluated by the firewall and on the detection of malicious packets; the traffic is redirected to the shadow server used to store malicious traffic as shown in the Figure-3 (L. Chen et al., 2009). The evidence analyzer collects the evidence from the malicious traffic that sent by shadow server. The evidence is further divided into several encrypted partitions and replicated to different resources for its storage. The evidence stored at multiple resources are used against intruders to prove them as the culprit.

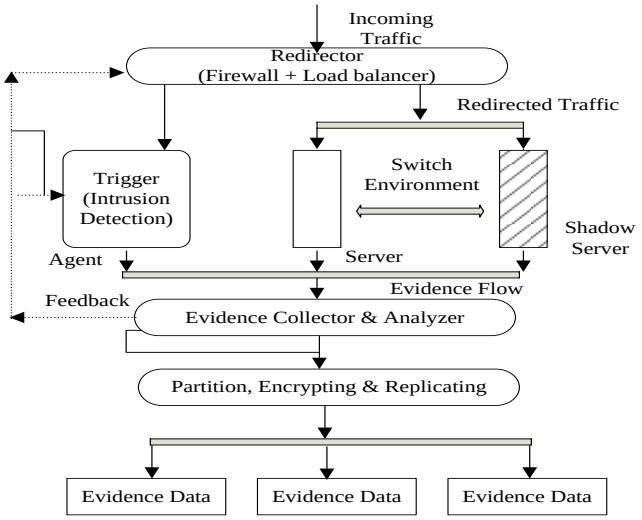

Figure 3: Dynamic forensics intrusion tolerance mechanism

In (Jiang, Tian, & Zhu, 2012), pattern analysis and protocol analysis mechanism are used to identify malicious traffic in the network. Pattern analysis is employed to analyze the network traffic based on previously identified patterns of malicious attack while the protocol analysis is deployed to analyze packets based on protocol types and apply the data analysis techniques to determine the malicious behavior of packets. The proposed solution employs two modules including network data acquisition engine and logs message capturing engine as shown in Figure 4 (Jiang et al., 2012). The network data acquisition engine is used to capture malicious network packet and store with its timestamp and send it to the network data analysis engine for analysis. However, log message capturing engine is used to capture all network traffic in a log and send it to log message analysis engine for identification of malicious patterns. The evidence extracted from logs and network packets is stored in the network forensics analysis record.

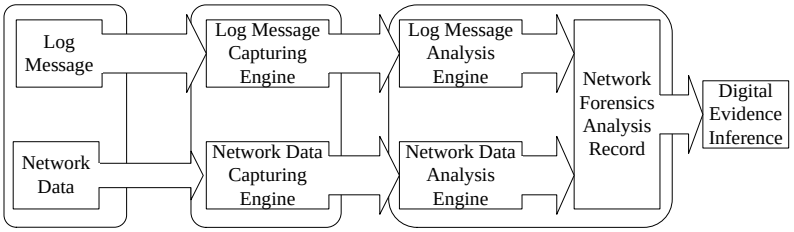

Figure 4: Network forensics intrusion detection analysis

3.2 C-NFM based on Traceback

Traceback is a method used in network forensics with the objective of getting to the source of the attack. Traceback simulates the malicious action performed by an intruder from its initial point until the end. NFI performs traceback by collecting traces from the network. Traceback helps in identifying every step of the malicious attack. However, traceback can be challenging when an intruder deletes or modifies its traces from the network. Single deletion or modification of trace can affect the whole traceback process. Frequently, intruders follow anti-forensics techniques to remove their trace or evidence from the network after exploiting the network.

In (H. S. Kim & Kim, 2011), traceback method uses the packet marking scheme to mark packet at the edge router by inserting an encoding pattern. The marking is used to verify the correctness of the packet further by the routers in the network. The marking scheme of authenticated evidence is used to append encoding pattern which is further decoded at the destination to determine the source of the attack. The packet marking is performed in the following steps: a) First, it selects the packet, b) Second, encoding is performed through AES-256 electronic codebook mode, and c) Third, the encoding pattern is appended to the selected packet. However, in (Fen, Hui, et al., 2012) only time to live (TTL) field of the packet is used to identify the attack path rather than using the encoding code inside the packet. The proposed method works by embedding 8-bits TTL value in the packet. The victim node detects the attack when it receives more than enough packets such as DoS or DDoS attacks. After detecting the attack the proposed C-NFM uses attack tree analysis algorithm to generate the attack path by using the TTL value of packets. The C-NFM (Fen, Hui, et al., 2012) is useful when there is a huge network traffic and it is difficult to store all network traffic due to the scarcity of storage resources.

A C-NFM is proposed in (L. M. Chen et al., 2013) to identify the origin of stealthy self-attacks in the network. The proposed method uses the historical network traffic rather than marking the packet to extract its network behavior profile by training the data. In addition, data are reduced by filtering the traffic into two categories; attack traffic and normal traffic. A random moonwalk algorithm (Xie, Sekar, Maltz, Reiter, & Zhang, 2005) is applied to determine the origin of the attack by analyzing the attack traffic. The method is scalable in terms of space and computation, and can be used for a large network traffic. In (Yu et al., 2013), a spread spectrum approach based on (code, frequency, and time) hopping-direct sequence spread spectrum is used to identify malicious attacks in the anonymous communication. In anonymous communication, the identity of the user is hidden which helps them to be saved from intruders in the network. Similarly, anonymous communication is also used by intruders to perform their malicious act while keeping themselves hidden from users and NFIs. In (Yu et al., 2013), an encrypted pseudocode is sent with the normal communication which is decrypted at the destination. Such process validates the

correctness of the communication by looking at packet alteration in the network traffic. Moreover, it marks the communication in time and frequency domains. To trace back the origin of the attack, a protocol is designed to integrate bloom filter with a hashing table at the router level (Jeong & Lee, 2013). The traffic at the router is filtered by using bloom filter and is hashed and stored in the compressed form in the database. The proposed C-NFM (Jeong & Lee, 2013) uses both real-time analysis as well as periodic analysis. The real-time analysis is performed on the hash tables while periodic analysis is conducted in the compressed hash table stored in the database. As shown in Figure 5 (Jeong & Lee, 2013), there are four managers including system manager, router manager, database DB manager, and attack analysis manager which coordinate together to trace back the attack. System manager sends various packets including attack packet, traceback packet, and sinkhole router selection packet to all the routers in the network that contains a timestamp and MAC fields. Timestamp field is used to retransmit the attack; whereas MAC field is used by the router to integrity the message. Router manager hashes all packets to pass through the router by using compress hash table module. DB manager is responsible for storing the attack information sent by the system manager and compressed hash table information sent by various routers. Attack analysis manager collects attack packets from the sinkhole router and IDS in the network. Thus, the proposed solution replays to the attacks attributed to time-stamp attached to the packets and keeps its integrity using the hashing function.

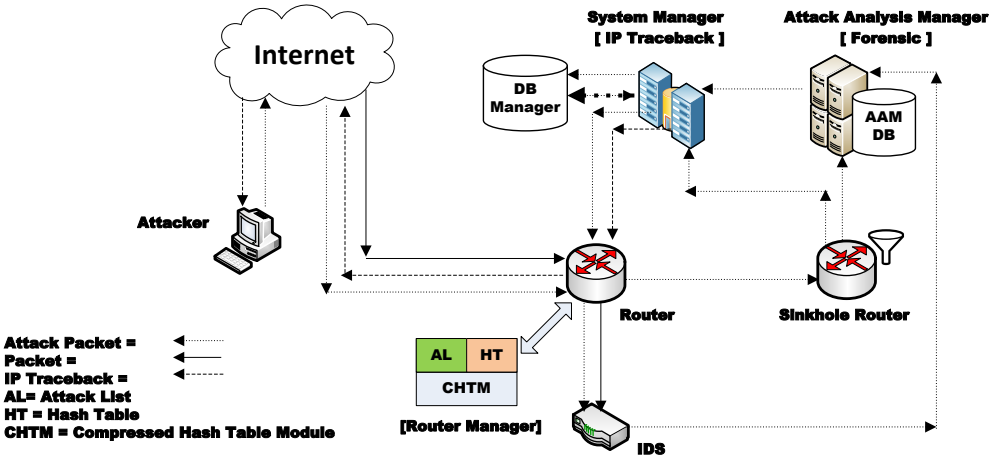

Figure 5: IP traceback protocol

3.3 C-NFM based on Distributive Nature

Distributive C-NFM is used to collect different malicious network traffic from various parts of the network and analyzes the network traffic at different locations of the network. The distributive C-NFM helps in analyzing network traffic in a centralized location by increasing response time due to large distances. However, analysis at different locations decreases the burden on the forensic server by analyzing the network traffic in multiple locations of the network. In (Shanmugasundaram et al., 2003), a C-NFM is proposed to collect malicious network

traffic from various network agents installed in the network. The packet header is analyzed for IP addresses, port connection, and session establishment with the help of bloom filter tracking. The analysis of network traffic is performed in a non-real-time environment. However, to overcome such problem, a real-time C-NFM is proposed to analyze the network traffic soon after the attack taken place (Ren & Jin, 2005). The traffic is collected, encrypted, and forwarded to the network forensic server through the SSL channel by network agent configure at various locations of the network. Network forensics server analyzes the traffic and updates existing malicious signature rule by identifying a new signature of the attack. These rules are sent to network agents to detect the similar attacks in the future. The network forensics server performs traffic and logs analysis to identify malicious traffic in the network traffic.

In (Ren, 2004), a distributive cooperative of C-NFM is proposed to determine possible risk and misbehavior of the network traffic. The method is based on the client-server architecture; wherein, client agents installed at different locations of the network and capture network packets in the form of network logs. The network logs are converted to the file and stored in the database which is sent to the forensic server for examining its pattern. The proposed solution performs the sequential steps as shown in Figure 6 (Ren, 2004). The topology is mapped by mapping engine, the incoming network traffic is filtered by filter & dump module, the converted engine converts traffic into a database which is mined by mining engine, survey engine produces output with network behavior, statistical procedures are applied to the network behavior, and finally a visualized analysis report is generated. The forensic server uses link analysis, sequential analysis, and classification of database files to determine various patterns presented. These patterns help NFI in replaying attacks to identify misbehavior of network packets. A real-time C-NFM based on immune theory is proposed to provide real-time evidence of the network attack (D. Wang, Li, Liu, Zhang, & Liu, 2007). An immune agent collects network traffic and applies antigenic presentation coding for further comparing with the non-self-code. During the matching, agent detects the intrusion and generates an alert message to the forensic server to record the digital evidence. In addition, digital signatures are generated by performing hashing and sent towards the forensic server. The forensic server performs analysis and stores the evidence by itself to generate replays of the attack. The architectural design of the proposed solution is shown in Figure 7 (D. Wang et al., 2007).

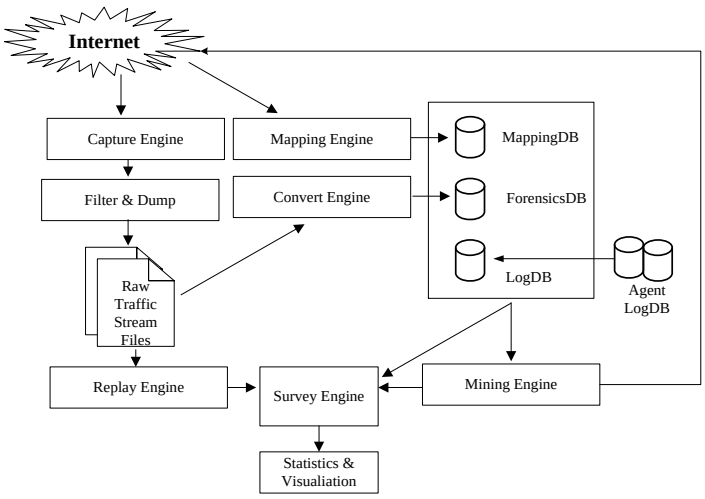

Figure 6: Distributed cooperative network forensics model

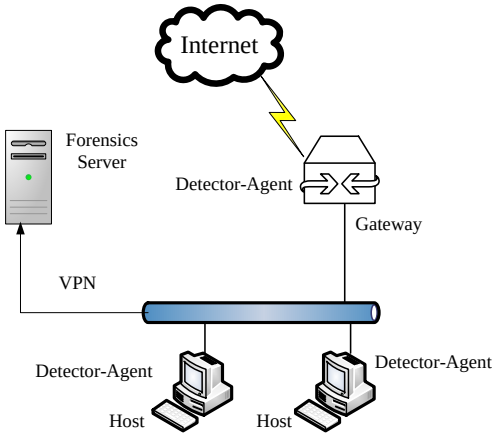

Figure 7: Immune agent based dynamical network forensics

3.4 C-NFM based on Attack Graph

An attack graph is used to identify the attack paths performed by an intruder while conducting its attack. Attack graph assists NFI to focus on routes used by intruder rather than investigating the whole network infrastructure. The attack graph represents the attack nodes as a vertex, and their transition as edges. For simplicity and easy analysis by NFI, attack graph provides a visualized outlook of the attack paths. NFI obtains information from attack graph by exploring attack paths performed by the intruder and use it to detect the similar attacks in the future. In (Albanese, Jajodia, Pugliese, & Subrahmanian, 2011), a scalability analysis is performed to identify the impact of the attacks on the enterprise using an attack graph. First, the proposed solution determines the dependency sequence of the attack by generating a dependency of attack graph. Second, the time span distribution is integrated with the attack graph to identify patterns of temporal attack probabilistic. Finally, a complete attack graph of the whole scenario is generated to associate malicious nodes to the attack paths. Additionally, the proposed solution

uses update index algorithm to update its index in real-time on receiving alert messages for possible attacks and responds accordingly. A multi-level attack in the network is investigated by (Fen, Xinchun, et al., 2012) using attack tree. The network attack is modeled to calculate system risk through examination of its security threats caused by various network attacks. A multi-attribute utility theory is used to determine the system risk, whereas each node in the attack tree is assigned to its weight. An attack sequence is identified after the system risk is calculated and the extra nodes in the attack tree are deleted. The node at the top level of the attack tree represents the node targeted by the intruder during its attack. However, in the case of intruder deletes its traces from the network which further complicates the detection of appropriate attack paths, an anti-forensic technique is incorporated in the attack graph to determine the deleted or altered traces (Liu et al., 2012). The proposed method used the anti-forensic node to monitor malicious activities during the deletion or alteration of various network traces in which, the intruder is unaware of and stores its information in an anti-forensic database. The anti-forensic database is analyzed to identify the intruder and its malicious action during its exploitation in the network.

Table 3: Summary of Current Network Forensics Methods

Classification	Code	Technique	Approach	Objective	References
Intrusion detection systems	I-1	Probabilistic inference	Reactive	Investigation of un-identify signature rule	(Sy, 2009)
	I-2	Formal Analysis	Reactive	Increase availability of forensic server	(L. Chen et al., 2009)
	I-3	Logging	Proactive	Monitoring alteration in log files	(Fan & Wang, 2010)
Traceback	T-1	Packet Marking	Proactive	Tracing attack by providing integrity for the evidence	(H. S. Kim & Kim, 2011)
	T-2	Packet Marking	Proactive	DDoS attack tracing through TTL field of packet header	(Fen, Hui, et al., 2012)
	T-3	Random Moonwalk	Proactive	Identification of stealthy self-propagating attacks	(L. M. Chen et al., 2013)
	T-4	Spread spectrum	Proactive	Investigation of malicious traffic in anonymous communication	(Yu et al., 2013)
	T-5	Logging	Reactive	Real-time investigation of network attacks	(Jeong & Lee, 2013)
Distributive	D-1	Packet Marking	Proactive	Distributive analysis through IP header fields	(Shanmuga sundaram et al., 2003)
	D-2	Logging	Reactive	Collection of potential evidence before they are deleted by intruders	(Ren & Jin, 2005)
	D-3	Logging	Proactive	Identification of origin of the attack	(Ren, 2004)
	D-4	Immune Agent theory	Reactive	Real-time network forensics with evidence integrity, validity and authenticity.	(D. Wang et al., 2007)

Classification	Code	Technique	Approach	Objective	References
Attack graph	A-1	Scalable analysis	Reactive	The impact of the individual attack on the enterprise.	(Albanese et al., 2011)
	A-2	Attack tree	Reactive	Identification of multi-level attacks	(Fen, Xinchun, et al., 2012)
	A-3	Anti-forensics attack graph	Reactive	Monitor intruder malicious behavior such as deletion of various traces.	(Liu et al., 2012)
	A-4	Finite cognitive map	Reactive	Identification of worst attack	(Diamah et al., 2012)
	A-5	Probabilistic Model	Reactive	Identification of root cause of the attack	(Zhang et al., 2012)

An intruder exploits the network by performing multiple attacks such as injecting malicious code, altering packets, the bombardment of packets, eavesdropping and slowing down the communication. All these attacks are needed to be investigated; however, due to real-time constraint; NFI has to identify the worst attack which causes more damage to the network. To determine this attack, an approach using a genetic algorithm to create a fuzzy cognitive map from the attack graphs (Diamah et al., 2012). In the fuzzy cognitive map, the attack nodes are converted to concept nodes and the transition between nodes are represented by edges. A weight is assigned to edges based on the impact of the concept nodes. The effect of concept node is calculated by inserting weights of edges into the adjacent matrix. The value of adjacent matrix is used by the genetic algorithm to find the worst attack in a pool of attack. Similarly, the root cause of the worst attack is very important and must be identified to control its effect on the present and future. It is prudent to handle against the root cause of the attack rather than defending against its effects. A proposed solution (Zhang et al., 2012) integrates Hidden Markov Model with attack graph to identify the root cause of the attack. The solution observes the network situation and generates a dependency graph which is sent to Hidden Markov Model for probabilistic computing between network state and observation. Further, an ant colony optimization technique is applied to identify the root cause of the attack.

4. C-NFM with Cloud Computing Network perspective: *SWOT Analysis*

In this section, the C-NFM is evaluated from the perspective of cloud network to discover its applicability by understanding the strengths, weaknesses, opportunities, and threats of each C-NFM. This will help NFI in using C-NFM for cloud network investigation and facilitates CSP in protecting their cloud networks from vulnerabilities caused by the intruders. The aforementioned C-NFM in Section-3 is explained separately with its SWOT analysis in terms of cloud network in

the following sections and comprehensive summary is provided in Table-4 at the end of section 4.

4.1 SWOT analysis of C-NFM Based on IDS

The strength of I-1 includes the probabilistic approach to extract hidden information from the network attacks and model these attacks to identify various intrusions found in it. The I-1 detects intrusion at different locations through the installation of IDS sensors in the network, which can be applied in cloud networks due to its distributed nature. However, the weaknesses of I-1 include the use of inhibiting negative behavior to detect the intrusion. There are chances for I-1 in generating an alert alarm on detecting an intrusion. Moreover, I-1 lacks a reduction mechanism used to filter the network traffic by only targeting malicious traffic. However, for cloud networks, it is required due to the huge amount of traffic generated by millions of users in a cloud computing environment. The opportunities for I-1 include the utilization of high-performance computing (HPC) resources of cloud computing to perform probabilistic analysis for extracting information from cloud networks. Similarly, data storage data centers can be used to store the network traffic and protect it from being deleted or overwritten (Chohan, Bunch, Krintz, & Canumalla, 2013). The I-1 has an opportunity to extend its scalability by configuring the IDS sensors in various locations of cloud network to detect the intrusion throughout cloud infrastructure. However, threats to I-1 incorporate the complexity in synchronizing distributive IDS sensors, collected distributive intrusion data, data integrity, and real-time analysis. The user or enterprise using the cloud services will have a keen interest in getting the investigation report regarding an attack as quickly as possible. The analysis process can malfunction when the data is being further altered or deleted by an intruder during its transmission to forensic servers.

The strength of I-2 incorporates the tolerance of forensic servers after it is attacked by an intruder due to having a back server. It is useful for cloud network forensics to reduce the time and computational cost by migrating the workload to other servers during the attack. Furthermore, I-2 performs analysis on the malicious traffic which is being separated from the normal traffic. Therefore, it reduces analysis time especially for cloud networks that contain a massive amount of the traffic. However, the weaknesses of I-2 include the inspection of incoming network traffic at a single point location such as an incoming router. The inspection of network packets at a single point is infeasible for investigation of cloud networks due to its connectivity with hundreds of gateways connecting different networks. Each network may contain thousands of user's data that transfer within and outside the cloud through multiple point locations. Moreover, the storage of single data at multiple locations can increase the analysis time and increase the chances to be attacked by the intruder. The opportunities of I-2 contain the storage capability of storing network traffic by using cloud storage resources and high computational intensive resources for analyzing the malicious traffic. A real-time analysis is required for cloud

networks to investigate large networks with a massive amount of traffics generated from numerous network nodes. Therefore, high capacity storage and high computational incentive resources are required which can be made available from cloud computing resources. The threats of I-2 include the false positive rate generated during the investigation process. The cloud networks are considered large networks with thousands of network nodes which increase the probability of malicious traffic. Likewise, there are more chances of all attacks not to detect by the security devices due to high network traffic which results in false positive rates. This could expose the cloud networks by having virtualized resources in the cloud computing. The I-2 has no proper mechanism to investigate virtualized networks and will increase the risk of virtual network attacks.

The multi-dimensional analysis is performed based on log data (static analysis) and network traffic (real-time analysis) which is considered as one of the strengths of I-3, especially for cloud network forensics. Due to high bandwidth data rate of a cloud network, it is difficult to analyze all the network traffic in real-time. Network logs can be generated and stored in the storage resources of the cloud and can be analyzed for intrusion later. Furthermore, the network logs are encrypted before they are sent for storage; it enhances the security level by maintaining the integrity of the evidence. Nevertheless, the weaknesses include the centralized analysis of the data. The analysis server is placed centrally which can be targeted by intruders to foil the analysis process. Moreover, performing analysis at a single point in cloud computing is not a wise decision due to distributive nature of its infrastructure. As a result, it reduces the scalability of the proposed method and increases the burden on the single analysis server. There are some opportunities for I-3 to be adopted in cloud network forensics such as using machine learning techniques to classify the network traffic to reduce analysis time and decrease false positive rate. Moreover, for recording network logs, cloud storage resources are the best option to store the network logs, however; such facility (storage resources) was unavailable in the traditional network environment due to the scarcity of resources. The threats to the I-3 incorporate integrity of the evidence analyzed at the forensic server. An intruder can get access to the forensic server and delete or alter digital evidence by performing the malicious behavior. Moreover, the intruder can attack the forensic server at any time to affect the rest of the cloud infrastructure.

4.2 SWOT analysis of C-NFM based on Traceback

The strength of T-1 includes marking the incoming packets at the edge router rather than storing all the packet in network logs. This feature is useful for cloud network forensics due to the burden of storing a massive amount of network traffic. To store network traffic in storage resources, a secure connection is required and might be far from the router which increases the time delays and latency. Additionally, T-1 uses three-level encoding schemes to increase the security level by keeping the integrity of the evidence. However, the weaknesses for T-1 include

marking the packets at edge router at a single location. In cloud network, marking the incoming packets from multiple incoming routers can be challenging due to geographical limitations. The opportunities for T-1 include the high computational intensive cloud resources to encode and decode a packet at different routers. These computing resources help in investing packets by inserting encoding code in the packet for security and decoding it at the destination. However, one of the threats faced by T-1 is the IP address spoofing. If the intruders change the IP address in the packet before insertion of encoding code, it will be difficult to identify the correct source of the attack, especially in cloud networks. Similarly, T-1 has to work out for IPv6 which is lacking in the T-1 proposed method. Furthermore, routers can also be attacked by an intruder to alter the encoding scheme which might be difficult to decode it at the destination.

The strengths of T-2 include marking TTL field of the packet header and exploring the rest of packet header fields. It is useful in cloud network forensics in terms of time, computation, and complexity. As a result, packets are updated at each router in the cloud network and it requires fewer time delays as compared to explore the rest of the packet fields. The T-2 also uses a filter at the router level to reduce network traffic for analysis which is beneficial for cloud network forensics due to the huge amount of the network traffic. However, the weaknesses of T-2 include the use of attack tree analysis algorithm to trace back the attack paths. To analyze a large scale of cloud networks, attack tree analysis algorithm will cause more time delays due to numerous network nodes present in the cloud. Assigning values to each node is challenging due to the wide distributed network infrastructure with thousands of network nodes. However, cloud computational resources can act as an opportunity for T-2 to apply attack tree analysis on the cloud network to investigate various network attacks. The attack tree requires complete information regarding network attacks generated on the cloud networks. Moreover, intelligent routers can be used to monitor and analyze network packet at the router level. The intelligent routers will speed up the forensic process by providing a quick incident response to the management. The threats to T-2 include modification of TTL value in the packet header. An intruder can alter the value of TTL in the packet header which might not be detected and may create more problems in the network. Moreover, an unexploited mechanism for the IPv6 address is also a threat for T-2. An intruder can attack through IPv6 packet header which requires a counter-measure to investigate the malicious behavior of the intruder.

The strength of T-3 includes the ability to trace the origin of stealthy self-propagating attacks in the network. It is helpful for cloud network forensics due to a widespread distributed network of cloud computing. The NFIs faces problems in tracing out stealthy self-propagating attacks due to its concealed and propagated effect in the network. The stealth self-propagating attack effects are hidden and difficult to identify especially in large cloud networks. However, the weaknesses of T-3 incorporate the construction of causal attack tree based on the directed host contact graph. The generation of causal attack tree in the cloud network would be more sophisticated and time

consuming. Casual attack tree consists of causal edges with frequent occurrence and thousands of numbers in cloud computing which could decrease the performance of cloud network forensics. The opportunities for T-3 include the use of artificial intelligent mechanism to reduce computational time in determining the origin of the stealthy self-propagating attacks in a causal tree. The inference based on intelligence will help NFI to investigate stealthy self-propagating attacks especially in the cloud network. The threats of T-3 include the improper identification of evidence against stealthy self-propagating attacks in the cloud network. Each causal node in the causal attack tree affects its receiver and propagates the attack to the next node. However, if all the causal edges are not properly identified in the causal attack tree, it further produces ambiguous output in determining the origin of the attack.

The strength of T-4 comprises of trace-backing attacks in anonymous communication in the network. It is significant for cloud network forensics because most intruders perform their attacks and hide by using characteristics of anonymous communication. Furthermore, T-4 also provides secrecy and accuracy by injecting secret pseudo-noise code with normal traffic to determine malicious activities of the intruder. However, in cloud networking environment, it is difficult to capture all network traffic. Therefore, the best solution is to use secret pseudo-code feature of T-4 to discover the deviation in the network traffic. The weaknesses of T-4 include the frequent changing of communication in multiple frequency ranges that might difficult for real-time monitoring. Furthermore, network virtualization is used to assign one network link to multiple users with a feature of isolation. It is difficult to capture a single user network traffic in real-time as it requires an equal number of pseudo-noise code that equivalent to the number of virtualized network channels. The opportunities of T-4 include the installation of intelligent network devices such as an intelligent router to send pseudo-noise code after regular interval of time with more precision and accuracy on a real-time basis. The high bandwidth of cloud networks assists such method to respond quickly in a time frame to identify malicious activities while using pseudo-noise code. Threats to T-4 incorporate leakage of the pseudo-noise code in the network communication. The moment an intruder gains access to the code, the intruder can change and keep their identity hidden. An intruder can perform further attacks on other cloud resources and make the attack more dangerous due to a large number of network nodes in the cloud network.

The strengths of T-5 include real-time analysis and periodic analysis. Real-time analysis is performed on hash tables while periodic analysis is done on a compressed hash table stored in a database. It is useful for cloud network forensics to analyze network traffic in real-time and later by using files from a database such as periodic analysis. Moreover, the proposed method uses timestamp and MAC field of the packet to investigate the packet for malicious behavior. In cloud networking, most probably the packet is exploited by an intruder that requires investigation and by looking at its packet header information. However, the weaknesses of T-5 include communication between various manager modules in the proposed method such as router

manager, system manager, database manager, and attack analysis manager. Many manager modules cause time delays in cloud network forensics due to their dependency on each other and its location distance in cloud computing. A slow or late response to investigation queries can delay the entire forensic process. The opportunities of T-5 include cloud storage resources especially for storing hash tables. The router memory is insufficient for the storage of hash tables although it has been compressed due to the scarcity of its memory. Storing hash tables in the persistent storage memory of the cloud increases the performance of the router and prevents hash tables from being attacked. The threats to T-5 incorporate the accessibility of hash tables in the router by an intruder. The intruder identifies the router containing hash tables and exploits the tables by modifying it with malicious behaviors. Similarly, an intruder can also modify the timestamp and MAC field of the packet that used by the router to replay the packet again for investigation of malicious attacks. A timestamp modification may prevent a packet from reaching its destination while the sender retransmits the packet again and exploited by the intruder.

4.3 SWOT analysis of C-NFM based on Distributive Nature

The strength of D-1 lies in its ability to perform an in-depth inspection of the packet header to investigate malicious attack performed by the intruder. The inspection is performed periodically on ports and sessions created between network devices. It is imperative for cloud network forensics to identify more information regarding the identity of the intruders due to a large number of users and huge network traffic in the cloud. Along with the packet header, D-1 also investigates the metadata information of the network to benefit NFI in investigating cloud networks where packet header information is spoofed by the intruder after the attack. However, D-1 has its weaknesses relating to centralize analysis performed at the forensic server. The forensic server can be exploited by the intruder to tamper with evidence and trace records. It is not advisable to analyze the entire cloud network traffic at a single point. To analyze cloud networks, it requires dedicated bandwidth, more computation, large storage, and secure communication to proceed with. Additionally, the proposed method uses a lightweight intrusion detection at multiple places in the network; some of the attacks might be undetected and cause malfunctions in the whole system. However, the opportunities of D-1 include the transfer of network logs to the forensic server in the secure communication. The current cloud environment helps to provide a secure communication between various devices in most of the cases. Likewise, distributed analysis is also possible for cloud networks to analyze various regions of cloud by using distributed resources of cloud computing. The distributed analysis will reduce time delays, upturn quick response, and boosts system performance for cloud network forensics. The threats faced by D-1 are single point analysis such as the forensic server. For instance, if an intruder gets the access to the forensic server and changes the evidence by altering the traces through attacks, it will be difficult for NFI in cloud networks to identify the source of the attack due to incorrect information found in network traces.

The strengths of D-2 consist of real-time analysis based on traffic and log analysis based on network logs. In cloud network, analyzing network traffic in real-time produces a quick response for investigation queries while log analysis provides in-depth investigation to trace back the attack accurately. The network traffic is analyzed for attack patterns and stored in a traffic database. However, network logs are analyzed in the forensic server to investigate the attack events and identify its source. The log file contains the network events occurred during the transaction of the attack. Furthermore, the proposed method uses an encryption to send the evidence and store it in a database which protects the network logs from being altered. However, the weaknesses of D-2 include single point forensic analysis using a forensic server at the central position. The disconnection of communication with forensic server stops the entire process by affecting the investigation performance. The single point analysis for cloud network forensic is unwise due to the dispersed network of the cloud computing such as large-scale data centers. Moreover, opportunities of D-2 include the scalability that can be achieved by using distributed network agents in the cloud network to detect malicious attacks. Distributed network agents capture a vast area in the cloud network by synchronizing network agents with each other and sending updates to the forensic server for data analysis. Using various cloud forensic servers (F.-Y. Lin, Huang, & Chang, 2015) to analyze the data based on location, it distributes the workload and reduces the response time for investigating queries. The threats to D-2 include the placement of network agents in distributed cloud networks. It is possible for the agent not to be installed at a position in cloud network and having a higher probability to be attacked thus, generates more evidence regarding its malicious behavior. The identification of appropriate place for collecting network traffic in cloud network is one of the challenging tasks for NFI.

D-3 has its strength in investigating network traffic at multiple locations in the network. The D-3 supports the cloud network forensics in investigating the distributive cloud networks with minimal time delays. Forensic servers installed at multiple locations collect network traffic from various security devices such as firewalls, and IDS. The nearest forensic server with the security devices is used to analyze network traffic consequently, the response time and network latency are decreased. However, prior to the transmission of network traffic to the forensic servers, it is stored in the database where its format can be compromised. Despite its weakness, D-3 has the opportunities in terms of computational resources in cloud computing for forensic analysis such as link and sequential analysis. The link analysis is used to determine the correlation between the forensic data while the sequential analysis is used to discover steps taken in the attack. These analyses benefiting the cloud network forensics in finding the correlation between the exploited nodes and steps performed by the intruder. The threats to D-3 include the verification of capturing a complete network traffic at various network agents devices installed in the network. Hover, how could one know that the network agents collect a complete traffic? The improper

capturing of network traffic could produce a malfunction result and slow down the entire forensic process.

The strength of D-4 consists of real-time analysis based on the immune theory using immune agent installed at various locations in the network. It assists cloud network forensics by collecting malicious traffic from distributed location in the cloud. The immune agent applies antigenic presentation code on the network traffic to perform a further comparison with the non-self-code. Upon obtaining the match, the immune agent generates an alert to the forensic server for the detection of intrusion. The forensic server starts collecting digital evidence which is analyzed for replaying different attacks. However, the bottleneck of the centralized forensic server is one of the weaknesses of D-4. Collection and storage of network traffic from various locations in the cloud network creates congestion near forensic server and decreases the performance of the server by producing fewer responses. In cloud networking, thousands of agents are installed in a large network infrastructure and this can be difficult for the forensic server to analyze all the network traffic from various locations. However, the opportunities for D-4 include cloud computing servers to analyze the traffic distributive. Once distributed forensic servers are installed, computational burden is distributed among servers which result in a better response to investigation queries in the large cloud network. The threat to D-4 is the leakage of the code used to detect the malicious attack. Once an intruder identifies the code, the communication contents can be easily inserted, modified, and altered which results in malfunction consequences. Additionally, if an attack is not detected, such as the match between non-self-code and antigenic presentation code, it may be difficult to trace out attacks in large cloud networks due to the proposed method.

4.4 SWOT analysis of C-NFM based on Attack Graph

The scalability analysis used by the attack graph in large networks with a huge amount of data is one of the strengths of A-1. The scalability analysis can assist cloud network forensics as it is exposed to the maximum part of the cloud network. The proposed method can also identify each attack with its impact on the enterprise and measure alert correlation in terms of its scalability. The identification of each attack with its impact on cloud resources provides an in-depth analysis of malicious behavior performed by the intruder during the attack. However, the weakness of A-1 is in the automatic generation of dependency attack graphs for the network devices that interact with each other during the attack. Moreover, the manual creation of attack graphs, especially for cloud networks, is inadequate for real-time responses to the forensic investigation queries. Besides its weaknesses, A-1 has the opportunities to use high computational cloud resources for generation of attack graphs with further investigation into the attack path. High computational resources generate and execute attack graph quickly by providing an accurate and efficient response in a real-time. The other opportunity of A-1 is the

generation of regional-wise attack graphs for cloud networks rather than generating a single huge attack graph for the whole cloud network. The mechanism requires an in-depth knowledge of the cloud network components to interact with each other during the attack. Nevertheless, threats to A-1 are the identification of all network components in a real-time to play a part during the attack. The improper selection of components generates incorrect attack graphs while producing erroneous attack paths. The incorrect attack graphs can be generated more for cloud networks due to its network virtualization feature. An intruder can also perform anti-forensics techniques to hide its traces by deleting its evidence in the network to complicate the investigation.

A-2 has its strength in the identification of an attack sequence by constructing an attack tree with assigned weighted. The assigned weighted in the attack tree nodes measure the system risk by adopting multi-attribute utility theory (Keeney & Raiffa, 1993). The cloud network has thousands of network resources that make the attack graph larger which make it difficult to trace the attack path accurately. In the proposed method, once a system risk is measured, it eliminates unsolicited nodes from the attack tree to make it easier for NFIs to explore the attack sequences in the attack tree. The applicability of A-2 in cloud network simplifies the investigation with more visibility for NFIs. However, A-2 is lacked automatic generation of attack tree. In cloud networks, an attack can affect the resources that incorporated into the attack tree for tracing the attack sequences. Performed this entire procedure manually in the cloud network is almost impossible due to abundant resources. The opportunity of A-2 is in the generation of distributed attack trees at various regional zones of cloud networks. It distributes the burden by generating different attack tree based on the location and administrative authority. Additionally, the proposed method can also use the computational resources of cloud computing to speed up the process by generating attack trees and finding the correlation between attacks to discover complete sequences of the attack. Nevertheless, the threat to A-2 is the improper identification of malicious nodes in the cloud network. If the nodes in cloud networks are not identified properly, it generates incorrect attack tree and make it more difficult to trace the complete sequence of the attack accurately.

The strength of A-3 includes the integration of anti-forensics features with the attack graph. Most intruders perform the attacks on the network and erase traces to make the investigation process difficult for NFI to track malicious activities. The situation is even more complex in cloud networks due to extended boundaries with virtualization containing a massive network traffic. The proposed method monitors intruder activities by inviting them to alter their traces in the network. Each activity of the intruder is monitored by anti-forensics nodes based on its attack type, access nodes, alteration, privileges, and various tools used. The information collected reduces the investigation time in cloud networks by analyzing anti-forensics files to highlight intruder malicious activities. However, the weakness of A-3 is the scalability of inserting anti-forensics nodes to monitor intruder malicious activities at different places of the network. One

intruder may perform multiple attacks at various locations in the cloud network which requires anti-forensics nodes to monitor each and every attack. The proposed method lacks of multi-level anti-forensics mechanism to trace multiple malicious activities of the intruder at the same time. Furthermore, the opportunity of A-3 includes the mining of intelligence with attack graphs to classify various attacks performed by the intruder based on attack types and locations of the attack. It assists cloud network forensics by investigating the attack paths categorically in large cloud networks. Similarly, generating distributive anti-forensics attack graphs for cloud network will decrease the burden on central forensic servers and generate a quick investigation response. The threat to A-3 includes attacks on virtual networks and resources in cloud computing. The A-3 has no up to mark strategy to identify attacks performed in virtual networks. The attacks performed in virtual networks can propagate it's intrude further to various resources of the cloud computing.

The strength of A-4 is the identification of the worst attack in the network by converting attack graphs to fuzzy cognitive maps by using a genetic algorithm. The proposed method A-4, assists cloud network forensics by identifying the worst attack in a pool of the attacks spread throughout the cloud network. In cloud computing, a service provided on a timely basis to user is one of the main priorities. However, investigating more attacks at the same time reduces the response time to various queries. The best solution is to identify the worst attack that has the highest risk to maliciously affecting other cloud resources. Nevertheless, the weakness of A-4 includes the centralized investigation of the worst attack in networks. In cloud network forensics, traffic collected from distributed networks increases the network latency and time delays. The situation becomes difficult for the proposed method to collect the network traffic and investigate at central position with real-time response. In addition, it also increases the adjacent matrix used for calculating the worst attack by fuzzy cognitive map that require high computational resources for its computation. The high computational resources of cloud provide an opportunity for A-4 to overcome the aforementioned problem by generating distributive fuzzy cognitive maps. It will distribute the workload based on regional location of the cloud and decreases the network latency, time delays, and increases high response. The threat to A-4 includes the improper generation of attack graphs due to overlooked of several malicious network nodes. Once the attack graph is generated, it is converted into fuzzy cognitive maps by transferring nodes into concepts, and edges into causal influences. However, if an attack graph is generated improperly, the fuzzy cognitive maps will produce false results thus, complicates the identification of the worst attack.

The strength of A-5 includes the identification of root cause of the attack by providing a cost effective security hardening in large networks. The A-5 can facilitate cloud network forensics by its scalability in terms of identification of the root cause of the attack in distributed large networks. It is better to perform a corrective action for the identified root cause rather than investigating its

effect on the network. Once the root cause is identified, NFI can easily be applied and corrective action can be taken to stop its effect. The A-5 first generates dependency attack graph by observing the network. Second, Hidden Markov Model (Fink, 2014) is applied to dependency attack graph to find the probabilistic nature between network nodes and observations. Third, ant colony optimization algorithm is applied to determine the root cause of the attack. However, the weakness of A-5 incorporates the generation of automatic dependency attack graphs. In large cloud networks, with network virtualization, the situation becomes more complex in generating dependency attack graph manually. The opportunity for A-5 includes the use of high computational cloud resources to generate attack graphs for numerous malicious network nodes. The cloud computational resources can overwhelm the problem of scarce resources faced by the proposed method using the traditional networks. However, the threat to A-5 includes the use of human observation during probabilistic findings. The human error simply leads to erroneous generation of attack graphs. Nonetheless, in cloud networks it is important to observe the entire malicious network nodes prudently based on coordinated network events.

Table 4: Network forensics Methods implications in cloud networks: *A SWOT analysis*

Network Forensics Methods	Strength	Weakness	Opportunities	Threats	Applicability level
I-1	- Detection of intrusion from different locations - Extraction of hidden information	- Inhibiting negative behavior -Lack of data reduction mechanism.	-Usage of HPC resources and cloud storage -Extending scalability	-Complexity -Data integrity	Moderate level
I-2	- Forensic server tolerance - Separation of traffic (normal + malicious)	- Single point network inspection - Storing data in multiple places	- Storage capability	- False positive rate - Virtualization	Moderate level
I-3	- Multi-dimensional analysis - Data encryption	- lack of a forensic mechanism for huge network traffic - Forensic server bottleneck - Scalability	- ML techniques for classification of network traffic - Cloud storage resources	- Data integrity of digital evidence at forensic server	Low Level
T-1	- Marking packet - Three level encoding scheme	- Recording at edge router - Low scalability	- Traceback source of the attack	- IP and MAC address spoofing - IPv6 incompatibility	Moderate level
T-2	- TTL field of packet header - Router level filter	- Time-consuming in attack tree analysis	- Computational resource - Intelligent routers	- IPv6 - Modification of TTL value	Low level
T-3	-Identifying stealthy self-propagating attacks	- Time-consuming in construction of causal tree	- Artificial intelligence techniques	- Ambiguous results due to improper identification of causal edges	Moderate level
T-4	- Traceback attack in anonymous communication - Secrecy and accuracy	- Multiple frequency range - Network virtualization	- Intelligent network devices	- Misuse of Pseudo-noise code	Moderate level
T-5	-Real-time and periodic analysis	- More communication between various modules	- Cloud storage resources	- Access to hash tables - Modification of packet fields	High level
D-1	- In-depth analysis (Metadata)	- Centralize analysis - False Positive	- Secure communication - Distributive analysis	- Bottleneck of forensic server	High level
D-2	- Real-time analysis (traffic & log analysis) - Secure communication	- Disconnection with forensic server	- Scalability - Distributive forensic analysis	- Network agent placement	Moderate level
D-3	- Distributive forensic analysis	- Data integrity	- Computational resources	- Verification of capturing complete data	High level
D-4	-Real-time analysis (Immune Theory)	- Centralized forensic server	- Distributive computational servers	- Code leakage - False positive	Moderate level
A-1	- Scalability analysis - Identification of each attack with its impact	- Automated generation of dependency attack graphs	- Computational resources - Region-wise attack graphs	- Identification of network components	Moderate level

Network Forensics Methods	Strength	Weakness	Opportunities	Threats	Applicability level
A-2	- Identification of attack sequence	- Automatic generation of attack tree	- Distributed attack trees - Computational resources	- Improper identification of affected nodes	Moderate level
A-3	-Anti-forensics in attack graph	- Scalability of anti-forensics nodes	- Machine learning - Distributive anti-forensics attack graphs	- Attack on virtual networks and resources	Moderate level
A-4	- Identification of worst attack	- Centralize investigation of worst attack	- Computational resource	-Improper generation of attack graphs	Moderate level
A-5	-Identification of root cause of the attack - Scalability	- Automatic generation of dependency attack graphs	- Computational resource	- Dependency on human observation	Moderate level

5. Discussion & Recommendation

As discussed in Section 4, each C-NFM has been evaluated through SWOT analysis. The objective of the evaluation is to identify the applicability of C-NFM to cloud networks in cloud computing paradigm. The study is motivated by virtualized and multi-tenant environment of the cloud computing. Each resource can be divided and shared by many users in a single event within a time frame. Therefore, complications increase in performing network forensics during the investigation of virtualized networks. In cloud computing environment, a comprehensive method of network forensic is unavailable for enlarged cloud network with a large number of multi-tenant resources. However, with easy access to and adoption of cloud computing, intruders divert their malicious intention to cloud networks. The intruder can obtain benefits of cloud computing due to the heterogeneity of data available at multiple locations of the cloud. Therefore, investigating malicious behaviors performed by intruders is necessitated to protect a huge amount of the user data in the cloud. One of the options is to explore the currently vulnerable environment created by intruders in the cloud by generating new ideas, techniques, algorithms, methods, frameworks, approaches and much more. This option is costly and time-consuming due to the frequent advancement of cloud computing technologies. The second option is to ensure the applicability of C-NFM in the cloud network by targeting the malicious behavior of the intruder.

Based on the aforementioned of a SWOT analysis in Section-4, the recommendations for C-NFM in cloud network perspective are highlighted in Table-5. The study recommends modification of C-NFM to ensure its applicability for cloud network forensics in a cloud network. Most of the cloud network traffic is massive that requires separation between normal and malicious network traffic for easy forensic analysis. The separation reduces the burden of storage and time required to analyze the malicious network traffic. For instance, an appropriate location is important, whereas security devices (network agents) are configured to detect network traffic containing malicious attacks. To find an appropriate location for security devices is easy in the traditional network due to physical accessibility of the network, whereas it is difficult in cloud networks due to virtualization, multi-tenancy, distribution, and inaccessibility factors of the cloud computing. The NFI can have access to the cloud network by having legal permission from the court. Moreover, a CSP itself can use C-NFM to prevent cloud network from the attack however, it might violate privacy and security of user's data on the cloud network. The collection of evidence against an intruder on the network might require access to the innocent users' data lying on the same network at the same time.

Furthermore, distributive forensic is required due to an enlarged network of cloud computing. It is difficult to analyze the whole cloud network traffic at a single location which requires dedicated, secure, and high bandwidth networks to send data at a single point from

distributed locations. The high computational intensive cloud resources can be used to analyze network traffic based on specified distributed locations to help NFI in identifying the origin of the attack in a real-time. Additionally, cloud network traffic uses IPv4 or IPv6 packets for data transmission which has to be considered by C-NFM while analyzing the network traffic. A problem arises when a proposed C-NFM only investigates IPv4 address while the attack is performed through the IPv6 address. To minimize this overhead, C-NFM has to investigate the payload data for various vulnerabilities during their investigation. Most intruders insert malicious code in the payload of the packet to propagate its attack in the network. By investigating payload data, it increases the chances of detecting malicious attacks while on the other hand; it reveals the privacy of the user data. With attack graphs, machine learning techniques and probabilistic models should be integrated to classify the huge amount of network traffic and calculate the uncertainty of attacks in cloud networks. The significance of intelligent mining is increased due to a large network with dispersed data. An intruder can easily hide by performing a malicious attack and deleting their traces while diverting to other resources in the cloud network. For instance, virtual machine users on a cloud resource could perform a side-channel attack on its neighbor virtual machine on the same resource and suddenly close down its virtual machine. The side channel attack will delete all the traces of the intruder on the virtual machines until and unless virtual machine logs has been generated.

Hence, the integration of machine learning, data mining, statistical analysis, and mathematical modeling can improve the applicability of C-NFM in cloud network infrastructure. Nowadays, more attacks are performed on cloud networks; and these call for the development and application of more forensic methods for investigating cloud network susceptibilities to restrict them in the future.

Table 5: Recommendation for Current Network Forensics Methods in Cloud Networks

Classification	Code	Recommendations
Intrusion detection systems	I-1	Using intelligent network traffic filter to analyze only malicious traffic and keep the integrity of the evidence by storing it in persistent storage of cloud.
	I-2	The accurate location identification from where the network traffic should be capture especially including virtual network links.
	I-3	Distributive forensic should be performed to reduce the load on the single forensic server as well as minimize it bottleneck problem.
Traceback	T-1	An intelligent mechanism requires identifying spoofing in both IPv4 and the IPv6 packet header.
	T-2	Investigating payload data as well to assist investigation rather than only focusing on the header field of the packet such as TTL value.
	T-3	Separation of normal nodes from attack node in the early stage of the investigation, it will generate causal tree quickly with responding in the real-time.

	T-4	Accurately identify each user's network traffic and send pseudo code in virtualized environment by differentiating them at the destination.
	T-5	The communication between various modules of the system should be reduced by organizing central communication management.
Distributive	D-1	The forensic analysis should be performed based on jurisdictional separation while keeping synchronization between forensic servers.
	D-2	Configure dynamic allocation of the network agents based on the load of the network traffic in the cloud.
	D-3	Collect and store malicious traffic in cloud storage close to the forensic server while ensuring the integrity.
	D-4	Perform distributive forensic analysis at various locations and incorporate verification mechanism to know the malicious traffic is collect and analyze completely.
Attack graph	A-1	Automatic generation of the attack graph for huge and spread cloud network in the cloud computing. It will reduce the time and visualize the whole attack scenario.
	A-2	Distributive probabilistic mechanisms require identifying nodes for attack tree, especially in the large cloud networks.
	A-3	An intelligent mechanism to identify the relevant locations to configure anti-forensic nodes to assist attack graphs.
	A-4	Require integration of probabilistic techniques with existing method to identified uncertainty of the attack as well.
	A-5	Machine learning techniques require supporting the existing method with more accurate inference for retrieving information from the network traffic.

6. Conclusion

Recently, cloud computing has attracted users, businesses, and enterprises due to its applicable services; however, intruders are free to use cloud resources to perform malicious attacks on the cloud computing. Most of the network attacks are investigated once it starts to exploit the networks through an accessing network similar to the cloud computing. However, due to virtualization and multi-tenancy in cloud computing, investigating the attacks in cloud network have become more sophisticated and complex. For example, virtualization causes the volatility issues of the data while multi-tenancy causes the privacy issues in investigating the network traffic in the cloud. Nevertheless, there is no well-established of cloud network forensics method that is available to investigate the network attack in the cloud computing infrastructure. Therefore, we performed a comprehensive study based on SWOT analysis of C-NFMs to determine the applicability of C-NFMs towards cloud networks. It is found that C-NFMs, at moderate levels, are applicable to the cloud network for investigating the network attacks in the cloud computing at its current implementation. To enhance its applicability, C-NFMs have to adopt artificial intelligence techniques such as classification, support vector machine, fuzzy systems, and neural networks to improve investigations in cloud networks while identifying

various network susceptibilities. The mining of intelligence is an important factor in collecting, preserving, and analyzing a huge amount of network traffic as evidence while keeping its integrity intact.

The study assists NFI from re-inventing the wheel in the development of new algorithms, methods, techniques, frameworks, and models. It provides a guideline for CSPs to make firm strategies to halt malicious attacks in the cloud network. This research study lays the groundwork for researchers, academics, network administrators, security firms, and consultants to explore the deficiencies highlighted in this paper regarding the cloud network forensics. However, a robust method is required to overcome the weaknesses highlighted while adopting defensive mechanisms to prevent cloud networks against cloud network threats.

Acknowledgement

This work is fully funded by Bright Spark Unit, University of Malaya, Malaysia and partially funded by Malaysian Ministry of Higher Education under the University of Malaya High Impact Research Grant UM.C/625/1/HIR/MOE/FCSIT/03 and RP012C-13AFR. The authors also extend their sincere appreciations to the Deanship of Scientific Research at King Saud University for its funding this Prolific Research Group (PRG-1436-16).

References:

Akyuz, Turker, & Sogukpinar, Ibrahim. (2009). *Packet marking with distance based probabilities for IP traceback.* Paper presented at the Networks and Communications, 2009. NETCOM'09. First International Conference on.

Albanese, Massimiliano, Jajodia, Sushil, Pugliese, Andrea, & Subrahmanian, VS. (2011). Scalable analysis of attack scenarios *Computer Security–ESORICS 2011* (pp. 416-433): Springer.

Almulhem, Ahmad. (2009). *Network forensics: Notions and challenges.* Paper presented at the Signal Processing and Information Technology (ISSPIT), 2009 IEEE International Symposium on.

Armbrust, Michael, Fox, Armando, Griffith, Rean, Joseph, Anthony D, Katz, Randy, Konwinski, Andy, . . . Stoica, Ion. (2010). A view of cloud computing. *Communications of the ACM, 53*(4), 50-58.

Baldwin, Adrian, Pym, David, & Shiu, Simon. (2013). Enterprise information risk management: Dealing with cloud computing *Privacy and Security for Cloud Computing* (pp. 257-291): Springer.

Beebe, Nicole Lang, & Clark, Jan Guynes. (2005). A hierarchical, objectives-based framework for the digital investigations process. *Digital Investigation, 2*(2), 147-167.

Bi, Jun, Deng, Hui, Xu, Mingwei, Shi, Fan, & Hu, Guangwu. (2013). A General Framework of Source Address Validation and Traceback for IPv4/IPv6 Transition Scenarios.

Birk, D., & Wegener, C. (2011, 26-26 May 2011). *Technical Issues of Forensic Investigations in Cloud Computing Environments.* Paper presented at the Systematic Approaches to Digital Forensic Engineering (SADFE), 2011 IEEE Sixth International Workshop on.

Buyya, Rajkumar, Yeo, Chee Shin, & Venugopal, Srikumar. (2008). *Market-oriented cloud computing: Vision, hype, and reality for delivering it services as computing utilities.* Paper presented at the High Performance Computing and Communications, 2008. HPCC'08. 10th IEEE International Conference on.

Buyya, Rajkumar, Yeo, Chee Shin, Venugopal, Srikumar, Broberg, James, & Brandic, Ivona. (2009). Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility. *Future Generation computer systems, 25*(6), 599-616.

Callado, Arthur, Kelner, Judith, Sadok, Djamel, Alberto Kamienski, Carlos, & Fernandes, Stênio. (2010). Better network traffic identification through the independent combination of techniques. *Journal of Network and Computer Applications, 33*(4), 433-446. doi: http://dx.doi.org/10.1016/j.jnca.2010.02.002

Carl, Glenn, Kesidis, George, Brooks, Richard R, & Rai, Suresh. (2006). Denial-of-service attack-detection techniques. *Internet Computing, IEEE, 10*(1), 82-89.

Carrier, Brian, & Spafford, Eugene H. (2004). *An event-based digital forensic investigation framework.* Paper presented at the Digital forensic research workshop.

Chen, Li Ming, Chen, Meng Chang, Liao, Wanjiun, & Sun, Yeali S. (2013). A Scalable Network Forensics Mechanism for Stealthy Self-Propagating Attacks. *Computer Communications*.

Chen, Lin, Li, Zhitang, Gao, Cuixia, & Liu, Yingshu. (2009). *Modeling and Analyzing Dynamic Forensics System Based on Intrusion Tolerance.* Paper presented at the Computer and Information Technology, 2009. CIT'09. Ninth IEEE International Conference on.

Chermack, Thomas J, & Kasshanna, Bernadette K. (2007). The use and misuse of SWOT analysis and implications for HRD professionals. *Human Resource Development International, 10*(4), 383-399.

Chohan, Navraj, Bunch, Chris, Krintz, Chandra, & Canumalla, Navyasri. (2013). Cloud platform datastore support. *Journal of grid computing, 11*(1), 63-81.

Choy, Sharon, Wong, Bernard, Simon, Gwendal, & Rosenberg, Catherine. (2012). *The brewing storm in cloud gaming: A measurement study on cloud to end-user latency.* Paper presented at the Proceedings of the 11th annual workshop on network and systems support for games.

Cohen, M. I. (2008). PyFlag – An advanced network forensic framework. *Digital Investigation, 5, Supplement*(0), S112-S120. doi: http://dx.doi.org/10.1016/j.diin.2008.05.016

Cusack, Brian, & Alqahtani, Muteb. (2013). Acquisition Of Evidence From Network Intrusion Detection Systems.

Dang;, Karen Kent; Suzanne Chevalier; Tim Grance; Hung. (August 2006). Guide to Integrating Forensic Techniques into Incident Response. *National*

Institute of Standards and Technology, (NIST).

Desmedt, Yvo. (2011). Man-in-the-middle attack *Encyclopedia of Cryptography and Security* (pp. 759-759): Springer.

Diamah, Aodah, Mohammadian, Masoud, & Balachandran, Bala M. (2012). Network Security Evaluation Method via Attack Graphs and Fuzzy Cognitive Maps *Intelligent Decision Technologies* (pp. 433-440): Springer.

Erl, Thomas, Puttini, Ricardo, & Mahmood, Zaigham. (2013). *Cloud Computing: Concepts, Technology & Architecture*: Pearson Education.

Fan, Ya-Ting, & Wang, Shiuh-Jeng. (2010). *Intrusion Investigations with Data-Hiding for Computer Log-File Forensics.* Paper presented at the Future Information Technology (FutureTech), 2010 5th International Conference on.

Farah, Badie N. (2013). A Model for Managing Uncertainty on the Cloud. *Journal of Management, 14*(6), 19.

Fen, Yan, Hui, Zhu, Shuang-shuang, Chen, & Xin-chun, Yin. (2012). A Lightweight IP Traceback Scheme Depending on TTL. *Procedia Engineering, 29*, 1932-1937.

Fen, Yan, Xinchun, Yin, & Hao, Huang. (2012). An Network Attack Modeling Method Based on MLL-AT. *Physics Procedia, 24*, 1765-1772.

Ferguson, Paul. (2000). Network ingress filtering: Defeating denial of service attacks which employ IP source address spoofing.

Fink, Gernot A. (2014). Hidden Markov Models *Markov Models for Pattern Recognition* (pp. 71-106): Springer.

Gani, Abdullah, Nayeem, Golam Mokatder, Shiraz, Muhammad, Sookhak, Mehdi, Whaiduzzaman, Md, & Khan, Suleman. (2014). A review on interworking and mobility techniques for seamless connectivity in mobile cloud computing. *Journal of Network and Computer Applications, 43*, 84-102.

Geethakumari, G, & Belorkar, Abha. (2012). Regenerating Cloud Attack Scenarios using LVM2 based System Snapshots for Forensic Analysis. *International Journal of Cloud Computing and Services Science (IJ-CLOSER), 1*(3), 134-141.

Gong, Chunye, Liu, Jie, Zhang, Qiang, Chen, Haitao, & Gong, Zhenghu. (2010). *The characteristics of cloud computing.* Paper presented at the Parallel Processing Workshops (ICPPW), 2010 39th International Conference on.

Hassan, S. R., Bourgeois, J., Sunderam, V., & Li, Xiong. (2012, 22-24 Oct. 2012). *Detection of Distributed Attacks in Hybrid & Public Cloud Networks.* Paper presented at the Semantics, Knowledge and Grids (SKG), 2012 Eighth International Conference on.

Helms, Marilyn M, & Nixon, Judy. (2010). Exploring SWOT analysis–where are we now?: A review of academic research from the last decade. *Journal of Strategy and Management, 3*(3), 215-251.

Ibrahim, Mohammed, Abdullah, Mohd Taufik, & Dehghantanha, Ali. (2012). *VoIP evidence model: A new forensic method for investigating VoIP malicious attacks.* Paper presented at the Cyber Security, Cyber Warfare and Digital Forensic (CyberSec), 2012 International Conference on.

Jackson, Susan E, Joshi, Aparna, & Erhardt, Niclas L. (2003). Recent research on team and organizational diversity: SWOT analysis and implications. *Journal of management, 29*(6), 801-830.

Jemili, Farah, Zaghdoud, Montaceur, & Ben Ahmed, M. (2007). *A framework for an adaptive intrusion detection system using Bayesian network.* Paper presented at the Intelligence and Security Informatics, 2007 IEEE.

Jeong, EunHee, & Lee, ByungKwan. (2013). An IP Traceback Protocol using a Compressed Hash Table, a Sinkhole router and data mining based on network forensics against network attacks. *Future Generation Computer Systems*.

Jiang, Liu, Tian, Guiyan, & Zhu, Shidong. (2012). *Design and Implementation of Network Forensic System Based on Intrusion Detection Analysis.* Paper presented at the Control Engineering and Communication Technology (ICCECT), 2012 International Conference on.

Katiravan, Jeevaa, Chellappan, C, & Rejula, J Gincy. (2012). Detecting the Source of TCP SYN Flood Attack using IP Trace Back. *European Journal of Scientific Research ISSN*, 78-84.

Keeney, Ralph L, & Raiffa, Howard. (1993). *Decisions with multiple objectives: preferences and value trade-offs*: Cambridge university press.

Khan, Suleman, Ahmad, Ejaz, Shiraz, Muhammad, Gani, Abdullah, Wahab, Ainuddin Wahid Abdul, & Bagiwa, Mustapha Aminu. (2014). *Forensic challenges in mobile cloud computing.* Paper presented at the Computer, Communications, and Control Technology (I4CT), 2014 International Conference on.

Khan, Suleman, Gani, Abdullah, Wahab, Ainuddin Wahid Abdul, & Bagiwa, Mustapha Aminu. (2015). *SIDNFF: Source identification network forensics framework for cloud computing.* Paper presented at the Consumer Electronics-Taiwan (ICCE-TW), 2015 IEEE International Conference on.

Khan, Suleman, Gani, Abdullah, Wahab, Ainuddin Wahid Abdul, Shiraz, Muhammad, & Ahmad, Iftikhar. (2016). Network forensics: Review, taxonomy, and open challenges. *Journal of Network and Computer Applications*.

Khan, Suleman, Gani, Abdullah, Wahab, W. A. Ainuddin, Shiraz, Muhammad, Mustapha, Bagiwa A., Samee, khan U., . . . Albert, Zomaya Y. (2016). Cloud Log Forensics: Foundations, State-of-the-art, and Future Directions. *ACM Computing Surveys*.

Khan, Suleman, Shiraz, Muhammad, Abdul Wahab, Ainuddin Wahid, Gani, Abdullah, Han, Qi, & Bin Abdul Rahman, Zulkanain. (2014). A Comprehensive Review on Adaptability of Network Forensics Frameworks for Mobile Cloud Computing. *The Scientific World Journal, 2014*, 27. doi: 10.1155/2014/547062

Kim, Ae Chan, Park, Won Hyung, & Lee, Dong Hoon. (2013). A Study on the Live Forensic Techniques for Anomaly Detection in User Terminals. *International Journal of Security & Its Applications, 7*(1).

Kim, Hyung Seok, & Kim, Huy Kang. (2011). *Network Forensic Evidence Acquisition (NFEA) with Packet Marking.* Paper presented at the Parallel and Distributed Processing with Applications Workshops (ISPAW), 2011 Ninth IEEE International Symposium on.

Kohn, M. D., Eloff, M. M., & Eloff, J. H. P. (2013). Integrated digital forensic process model. *Computers & Security, 38*(0), 103-115. doi: http://dx.doi.org/10.1016/j.cose.2013.05.001

Kruegel, Christopher, Mutz, Darren, Robertson, William, & Valeur, Fredrik. (2003). *Bayesian event classification for intrusion detection.* Paper presented at the Computer Security Applications Conference, 2003. Proceedings. 19th Annual.

Li, Shujun, & Schmitz, Roland. (2009). *A novel anti-phishing framework based on honeypots*: IEEE.

Lin, Feng-Yu, Huang, Chien-Cheng, & Chang, Pei-Ying. (2015). A cloud-based forensics tracking scheme for online social network clients. *Forensic Science International, 255*, 64-71. doi: http://dx.doi.org/10.1016/j.forsciint.2015.08.011

Lin, I-Long, Yen, Yun-Sheng, Wu, Bo-Lin, & Wang, Hsiang-Yu. (2010). *VoIP network forensic analysis with digital evidence procedure.* Paper presented at the Networked Computing and Advanced Information Management (NCM), 2010 Sixth International Conference on.

Liu, Changwei, Singhal, Anoop, & Wijesekera, Duminda. (2012). *Using Attack Graphs in Forensic Examinations.* Paper presented at the Availability, Reliability and Security (ARES), 2012 Seventh International Conference on.

Madhow, Upamanyu. (2008). *Fundamentals of digital communication*: Cambridge University Press.

Mell, Peter, & Grance, Tim. (2011). The NIST definition of cloud computing.

Moeller, Robert R. (2014). Cloud Computing, Virtualization, and Wireless Networks. *Executive's Guide to COSO Internal Controls: Understanding and Implementing the New Framework*, 203-215.

Nguyen, Khoa, Tran, Dat, Ma, Wanli, & Sharma, Dharmendra. (2014). *An approach to detect network attacks applied for network forensics.* Paper presented at the Fuzzy Systems and Knowledge Discovery (FSKD), 2014 11th International Conference on.

Palmer, Gary. (2001). *A road map for digital forensic research.* Paper presented at the First Digital Forensic Research Workshop, Utica, New York.

Pelaez, Juan C, & Fernandez, Eduardo B. (2009). *Voip network forensic patterns.* Paper presented at the Computing in the Global Information Technology, 2009. ICCGI'09. Fourth International Multi-Conference on.

Pilli, Emmanuel S., Joshi, R. C., & Niyogi, Rajdeep. (2010). Network forensic frameworks: Survey and research challenges. *Digital Investigation, 7*(1–2), 14-27. doi: http://dx.doi.org/10.1016/j.diin.2010.02.003

Qi, Han, Shiraz, Muhammad, Gani, Abdullah, Whaiduzzaman, Md, & Khan, Suleman. (2014). Sierpinski triangle based data center architecture in cloud computing. *The Journal of Supercomputing, 69*(2), 887-907.

Raza, Mudassar, Iqbal, Muhammad, Sharif, Muhammad, & Haider, Waqas. (2012). A survey of password attacks and comparative analysis on methods for secure authentication. *World Applied Sciences Journal, 19*(4), 439-444.

Reith, Mark, Carr, Clint, & Gunsch, Gregg. (2002). An examination of digital forensic models. *International Journal of Digital Evidence, 1*(3), 1-12.

Ren, Wei. (2004). *On A Reference Model of Distributed Cooperative Network, Forensics System.* Paper presented at the iiWAS.

Ren, Wei, & Jin, Hai. (2005). *Distributed agent-based real time network intrusion forensics system architecture design.* Paper presented at the Advanced Information Networking and Applications, 2005. AINA 2005. 19th International Conference on.

Ritchey, Ronald W, & Ammann, Paul. (2000). *Using model checking to analyze network vulnerabilities.* Paper presented at the Security and Privacy, 2000. S&P 2000. Proceedings. 2000 IEEE Symposium on.

Schluessler, Travis, Goglin, Stephen, & Johnson, Erik. (2007). *Is a bot at the controls?: Detecting input data attacks.* Paper presented at the Proceedings of the 6th ACM SIGCOMM workshop on Network and system support for games.

Schneier, Bruce. (2011). *Secrets and lies: digital security in a networked world*: Wiley. com.

Shamsi, Jawwad, Khojaye, Muhammad Ali, & Qasmi, Mohammad Ali. (2013). Data-intensive cloud computing: requirements, expectations, challenges, and solutions. *Journal of grid computing, 11*(2), 281-310.

Shanmugasundaram, Kulesh, Memon, Nasir, Savant, Anubhav, & Bronnimann, Herve. (2003). ForNet: A distributed forensics network *Computer Network Security* (pp. 1-16): Springer.

Shimeall, Timothy J., & Spring, Jonathan M. (2014). Chapter 11 - Network Analysis and Forensics. In T. J. Shimeall & J. M. Spring (Eds.), *Introduction to Information Security* (pp. 235-251). Boston: Syngress.

Shinno, Hidenori, Yoshioka, Hayato, Marpaung, Sihar, & Hachiga, Soichi. (2006). Quantitative SWOT analysis on global competitiveness of machine tool industry. *Journal of engineering design, 17*(03), 251-258.

Shiraz, Muhammad, Abolfazli, Saeid, Sanaei, Zohreh, & Gani, Abdullah. (2013). A study on virtual machine deployment for application outsourcing in mobile cloud computing. *The Journal of Supercomputing, 63*(3), 946-964.

Shiraz, Muhammad, Gani, Abdullah, Shamim, Azra, Khan, Suleman, & Ahmad, Raja Wasim. (2015). Energy Efficient Computational Offloading Framework for Mobile Cloud Computing. *Journal of Grid Computing*, 1-18.

Srivastava, A, Gupta, BB, Tyagi, A, Sharma, Anupama, & Mishra, Anupama. (2011). A recent survey on DDoS attacks and defense mechanisms *Advances in Parallel Distributed Computing* (pp. 570-580): Springer.

Sy, Bon K. (2009). Integrating intrusion alert information to aid forensic explanation: An analytical intrusion detection framework for distributive IDS. *Information Fusion, 10*(4), 325-341.

Taylor, Mark, Haggerty, John, Gresty, David, Almond, Peter, & Berry, Tom. (2014). Forensic investigation of social networking applications. *Network Security, 2014*(11), 9-16. doi: http://dx.doi.org/10.1016/S1353-4858(14)70112-6

Vonk, Guido, Geertman, Stan, & Schot, Paul. (2007). A SWOT analysis of planning support systems. *Environment and Planning A, 39*(7), 1699.

Wang, Diangang, Li, Tao, Liu, Sunjun, Zhang, Jianhua, & Liu, Caiming. (2007). *Dynamical network forensics based on immune agent.* Paper presented at the Natural Computation, 2007. ICNC 2007. Third International Conference on.

Wang, Xiao-Jing, & Wang, Xiao-Yin. (2010). Topology-assisted deterministic packet marking for IP traceback. *The Journal of China Universities of Posts and Telecommunications, 17*(2), 116-121.

Xie, Yinglian, Sekar, Vyas, Maltz, David A, Reiter, Michael K, & Zhang, Hui. (2005). *Worm origin identification using random moonwalks.* Paper presented at the Security and Privacy, 2005 IEEE Symposium on.

Xiong, Wei, Hu, Hanping, Xiong, Naixue, Yang, Laurence T, Peng, Wen-Chih, Wang, Xiaofei, & Qu, Yanzhen. (2014). Anomaly secure detection methods by analyzing dynamic characteristics of the network traffic in cloud communications. *Information Sciences, 258*, 403-415.

Xu, Jianlin, Yu, Yifan, Chen, Zhen, Cao, Bin, Dong, Wenyu, Guo, Yu, & Cao, Junwei. (2013). MobSafe: cloud computing based forensic analysis for massive mobile applications using data mining. *Tsinghua Science and Technology, 18*(4).

Yonghui, Li, Yulong, Wang, Fangchun, Yang, Sen, Su, & Dong, Yan. (2010). *Deterministic packet marking based on the coordination of border gateways.* Paper presented at the Education Technology and Computer (ICETC), 2010 2nd International Conference on.

Yu, Wei, Fu, Xinwen, Blasch, Erik, Pham, Khanh, Shen, Dan, Chen, Genshe, & Lu, Chao. (2013). *On Effectiveness of Hopping-Based Spread Spectrum Techniques for Network Forensic Traceback.* Paper presented at the Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), 2013 14th ACIS International Conference on.

Zhang, Zonghua, Wang, Shuzhen, & Kadobayashi, Youki. (2012). Exploring attack graph for cost-benefit security hardening: A probabilistic approach. *Computers & Security*.