**A Comprehensive Study on Network and Computer Forensic Framework**

Vaghela Rajdipsinh Dhirubhai1, Avinash Sharma2, Dankan Gowda V3 ,Mayuri Kundu4, Arudra Annepu5

1Assistant Professor, Department Of Computer Science & Engineering, Parul Institute of Engineering and Technology Parul University Vadodara, Gujarat, India.

2Professor, Computer Science & Engineering Department, M. M. Engineering College, Maharishi Markandeshwar (Deemed To Be) University, Mullana, Ambala, India,

3Department of Electronics and Communication Engineering, BMS Institute of Technology and Management, Bangalore, Karnataka, India.

4Assistant Professor, School of Computer Science and Engineering, REVA University, Bangalore, Karnataka, India.

5Assistant Professor, Department of Computer Science & Engineering, Rajiv Gandhi Institute of Technology, Cholanagar, R.T Nagar, Bangalore, Karnataka, India.

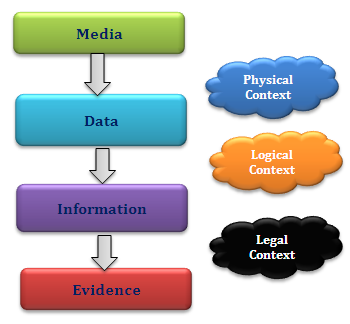
**Abstract**

Penetration of Information and Communication Technology (ICT) in the modern society, has led to expansion of the crime domain to Network and Computer related crimes across cyberspace. The spreading domain of cybercrime is a severe concern for forensic analysis. An extensive evidence source for cyber forensic examination of potential assaults on the privacy and integrity of sensitive information is network node produced traffic. Thus, presenting admissible, well defined and documented evidence in a court of law is the goal of cyber forensics. In this chapter a comprehensive review of the available literature related to sub-domains of the work done. It includes the literature on the basic concept about the previous frameworks for analysis of data that try to optimize the framework or techniques to increase the scalability, efficiency, and accuracy of the forensic architecture on the network.

Keywords – Efficiency, Network, Forensic, Crime, Analysis, Bod-Data, Information.

1. **INTRODUCTION**

Cyber forensic is an investigative process that brings satisfactory evidence in the court of law. The process of scientifically proven methods includes gathering, processing, interpreting and presenting undeniable facts and digital evidence of cybercrimes, in the court of law. In the present world of network computing, forensic techniques have been an area of interest for research and professional because of its continuously expanding domain and continuous increasing excitement and curiosity over the years (S. Vinjosh Reddy, and C. H. Pradeep Reddy. 2010). With the periodical increase in the domain, the definition and requirement of supporting tools and techniques have come a long way. Approximately, in every couple of years, the process phases for network forensics keep changing.

****

**Fig. (1).** Transformation of confiscated material into evidence

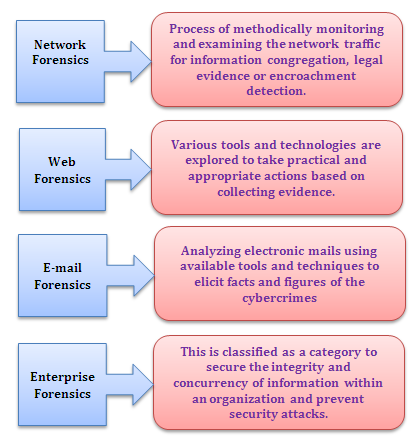
The context of the information keeps changing with the media transformation from data to information to evidence and finally a court-admissible form of evidence document as shown in Figure.1. Based on the target media of cybercrime, cyber forensics terminology is classified as follows and presented in Figure.2:

***Network Forensics:*** Process of methodically monitoring and examining the network traffic for information congregation, legal evidence or encroachment detection (S. Almulla, Y. Iraqi, and A. Jones. 2013). In order to find security assaults, network problems must be gathered and analysed.

***Web Forensics:*** Various tools and technologies are explored to take practical and appropriate actions based on collecting evidence. Hence, it aims at finding digital evidence over the World Wide Web. It is also pronounced as Network Forensics applied over the Web.

***E-mail Forensics:*** Analyzing electronic mails using available tools and techniques to elicit facts and figures of the cybercrimes and presenting evidence found in the investigation process in court.

***Enterprise Forensics:*** This is classified as a category to secure the integrity and concurrency of information within an organization and prevent security attacks. Generally, it follows a protective approach and not much concerned about evidence. System Forensics: For collection and gathering of evidence from computing devices, system forensics used. Unlike other forensics types, it is applicable when a crime has happened via stand-alone devices.

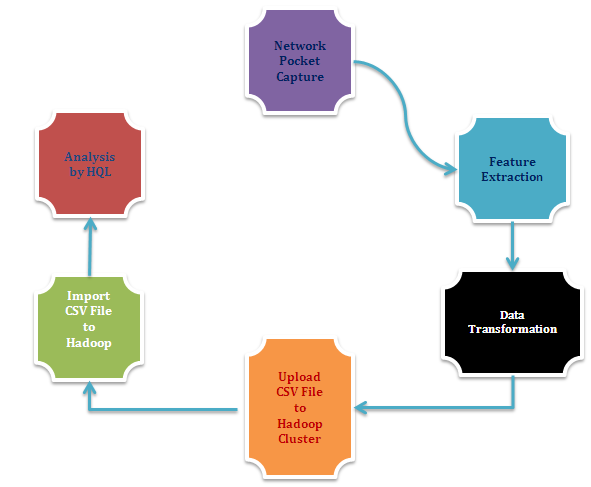
****

**Fig.(2).** Cyber Forensics Terminologies

1. **Frameworks for Big Data Forensics**

(M. Uzun and O. Abul. 2016) author’s proposed an approach for big data analysis that focused on detecting anomalies in network traffic and its effect can easily be prevented with the change in network traffic distribution. These network changes can be done by finding gaps between abnormal traffic and the usual traffic. This literature is further divided into three sub models or categories to detect unusual traffic. (i) The regular traffic selection model uses the concept of differentiating anomalies from normal data traffic instead of focusing on specific anomalous behaviours. (ii) The abnormal traffic selection model focuses to ensure that there isn’t too much normal traffic rather than focusing on abnormal traffic. (iii) Mixed compensation model adds the results from both the above models to produce final results. Like the Normal traffic selection sub model, this model also has the two stages: Training and Test stage.

The results of the analysis were: (i) Random forest classification algorithm can cope up with any change in the traffic that might occur and deploying the proposed model can decrease the false negative rate. (ii) If the amount of training data in a separate group differs significantly, then classify model developed using decision tree would be preferred where the training data set is huge. The decision tree model should be avoided for the abnormal traffic. (iii) With the purpose to reduce false positives, the k-means algorithm is used to alter the number of clustering in the regular traffic selection model thereby increasing the proficiency of the model to find anomaly behaviour.

****

**Fig.(3).** Big Data Analysis model

(C. Wang, and R. Wong. 2015) author’s proposes a new model using big data analytics to analyze network traffic. This model finds anomaly behaviour and malicious data being transferred through the networks by loading the data into hive databases in HDFS environment. This model consists of five phases, as shown in Figure.3. Large chunks of data can be looked upon in no time using this model. Hive queries executed, and the time taken to detect anomalies was calculated. The first dataset consisting of close to 400000 packets and 50 MB, the second data set consisting of close to consisting near 3100000 packets and the third dataset had 1200000 packets adding up to 131 MB.

(R. R. R. Robinson. 2015) author’s proposes the implementation of Hadoop for developing an Intrusion Detection System (IDS) which works in real-time for ultra-high-speed data. The four levels of an Intrusion Detection System are the capture layer, the filtering and load balancing layer, the Hadoop processing layer, and the decision making layer. In implementing machine learning, Java programming was used along with the WEKA 3.6.12 library.

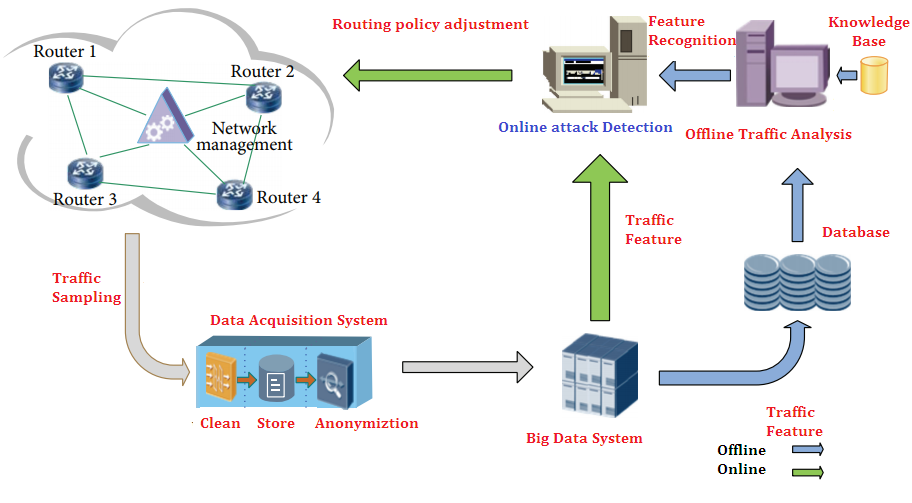
**Stage-1:** For packet captured the data, the filtration process is performed at filtration and load balancing server (FLBS) and then forwards packets to the non-intruder flows, for processing.

**Stage-2:** Check to figure out if the incoming packet belongs to an already existing flow else it is registered as a new flow with unique entries of four tuples, source and destination IP, source port, and destination port, as first line.

**Stage-3:** The master node keeps copying packet information into the sequence file till the duration threshold diverges and then sequence file sent to one of the data nodes for computing flow parameters.

**Stage-4:** The data node, then uses MapReduce to calculate final values of every feature for intrusion detection, which is forwarded to the decision server, which then further decides whether it is an intrusion or normal flow.

Figure.4 shows how Jia, Bin, et al., using a system based on Multivariate Dimensionality Reduction Analysis (MDRA), proposed an unique DDoS assault mechanism that operates in real-time and focuses on distinguishing attack traffic from regular data flows in Big data. Less dimensional and more representative characteristics may be extracted with the help of Principal Component Analysis (PCA). The primary components are a projection of the residual dimensions. The first principal component is the linear combination for the largest variance. If the data for the first PC is insufficient to extract information from the original variables, then the second linear combination is chosen. This approach based on two algorithms, (i) The Triangle area technique It can parse an obtained network traffic record and pull out geometrically related information for any pair of characteristics and (ii) Mahalanobis distance (MD) which is capable of similarity measurement between every two traffic records.

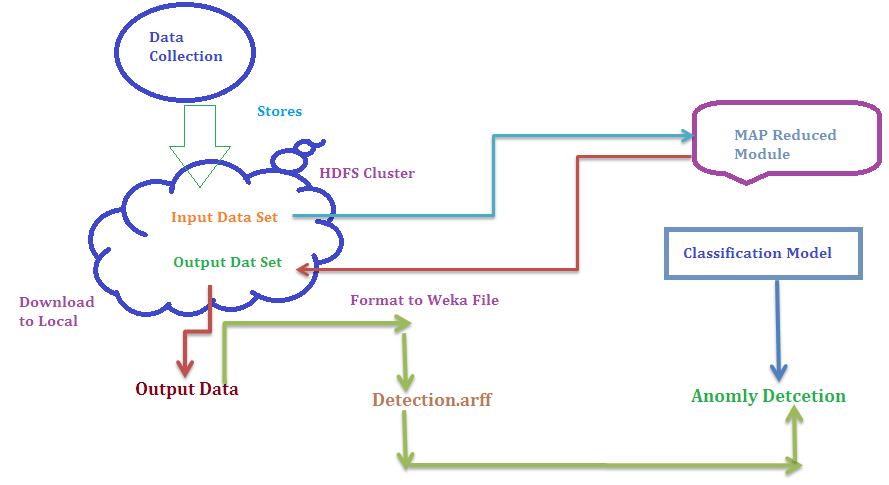
****

**Fig. (4).** Framework for identifying Distributed Denial of Service Attacks in Real Time

Multivariate Correlation Analysis (MCA) of traffic characteristics in the Big Data network environment may be used to identify Distributed Denial of Service (DDoS) attacks. Lastly, we produced benchmark data using a covariance matrix and MD and developed attack traffic identification based on MD and the chosen threshold.

1. **Basic Architecture for Big Data Analytics**

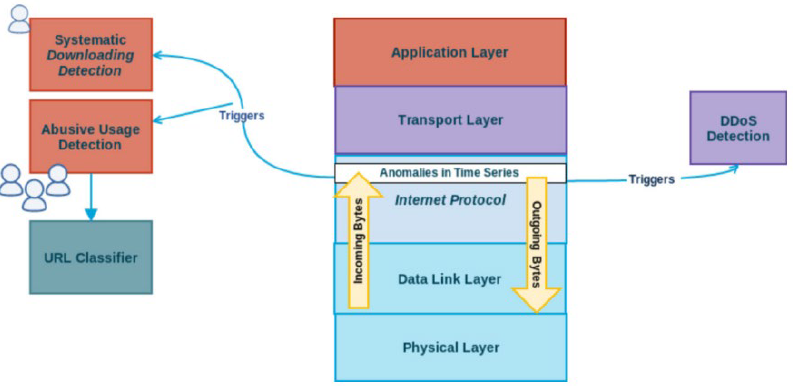
(G. Shrivastava and B. B. Gupta. 2014) author’s considered a model which is a pool cloud computing and machine learning to detect malicious traffic for big real-time data. Anomaly detection module uses HDFS, MapReduce, and Weka in collaboration with Machine Learning (ML) and Cloud Computing to improve the efficiency and accuracy of the test system, of each module as shown in Figure.5. Data processing is a crucial aspect, to detect abnormal behavior of traffic using machine learning Therefore, good and clean data must always be fed to the algorithm so that after some minor algorithm-specific changes or additions, the algorithm can work efficiently. The feature selection module and normalization module are a part of the map function. The normalization module focuses on the availability and sanity of data. It focuses on three steps (i) Unify coding format with UTF-8; (ii) Converting capital letters to lower case; (iii) Filter static files with input data, such as “. jco”, “. mp3”, “.jpg”, “.png”, “. js”, “. gif”. The feature selection module translates traffic data to multi-dimensional vectors using machine learning. To compute the global vector by combining the individual map’s vectors, first reduce function duplicate the input data to separate nodes, which may lessen the quantity of data and enhance the computation performance. For convenient management of numeric data, quantification by log2 is done after duplication. The detection model is set up with label data using the mining process of ML algorithms. Three machine learning models used and their results were compared using Weka interface, and decision tree is proving to be the most efficient among them.

****

**Fig .(5).** Basic Architecture for Big Data Analytics Using Hadoop and Weka

(M. H. Mate and S. R. Kapse. 2015) To identify unusual patterns in big data, the scientists suggested an anomaly detection method they named Adaptive Stream Projected Outlier detector (A-SPOT). For this particular case study of A-spot, we employed the KDD CUP anomaly detection programme from 1999. Because of the presence of a large labelled sample anomaly, Sparse Subspace Template (SST) is generated which contains Supervised SST Subspace (SS) besides Fixed SST Subspace (FS) and Unsupervised SST Subspace (US). For the generation of SS in SST, supervised learning is performed. Multi-Objective Genetic Algorithm (MOGA) is performed to the sample anomalies which belong to the same class to produce a SS for that class Once SST is obtained, Projected Cell Summaries (PCS) are generated for each subspace in SST to find abnormal behavior.

However, the data from PCS is not updated to avoid biases towards the abnormal data, since the original data set contains both normal and abnormal data. As a result, A-SPOT would be less able to spot abnormalities. After identifying the signature subspaces of all anomalies, a lookup table is constructed. We also get a glimpse of other crucial elements, such as the elimination of false positives, the creation of training data, and the classification of anomalies via the use of outlying subspace analysis.

****

**Fig. (6).** Basic Architecture of Multi-Level Anomaly Detection System

(R. Hunt and S. Zeadally. 2012) author’s proposed a Real-Time Hybrid IDS by making use of Apache Storm which is a fault tolerant, a big data stream processor that works in real time. Proposed approach performs hybrid detection, which uses Corner Classification 4 (CC4) neural network for new attacks and Multi Layer Perceptron (MLP) implementation uses misuse-based detection of existing attacks. The Apache Storm topology consists of an Input file reader called spout (stream source) which sends streams of data packets to two bolts: CC4 Launcher and MLP Launcher. The work of MLP Launcher is to gather strictly appropriate weights from the trained signatures and forward them to the next MLP network. CC4 Launcher extracts the input’s respective signatures from the training and forwards it to its respective bolt CC4 network. The results from both MLP and CC4 networks are combined and sent to another bolt called Post- Process which classifies the output as standard data or known attack with a class or an unknown attack. The ISCX Intrusion dataset is used for simulation rather than outdated datasets like KDD ’99 which does not include the latest attacks. The real-time IDS use neural networks, and therefore its average accuracy touched 89% with a false positive rate of 4.32%. This model can be used for all current day network security environments and is useful in real-time detection due to the ability of Apache Storm of real-time data processing while handling big data.

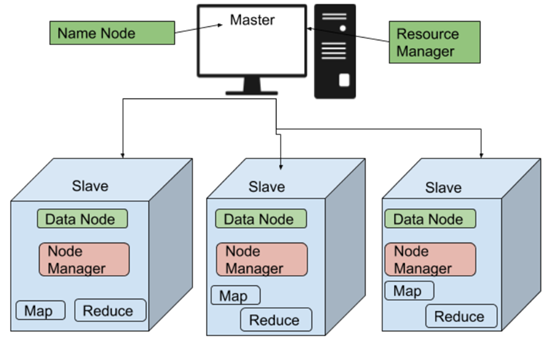
(Pallavi Asrodia and Vishal Sharma. 2013) author’s has given several techniques for anomaly-based intrusion detection with the focus on an on-campus environment at the network edge. The LAN network considered to have more than 9000 users on campus with 90 Mbps of Internet access. Since, this an extensive network, appropriate techniques are required to model user behavior using big data analysis. For classification, URL’s were classified such as entertainment, and academic which are useful for the characterization of users. Models for URLs are built monthly, by using proxy logs. Users were monitored for suspicious usage on a weekly basis as it serves as a more effective timeline to reflect upon the characteristics of abusive users.

Two class labels namely usage-based and user-based used for classification of users. While the abusive usage could be detected using the usage-based models, the characteristics of abusive users can be extracted using the user-based model. Anomaly detection was also effectively displayed through time-series analysis. Machine learning was used to detect abnormal behaviour in Internet usage. The same technique used on the campus edge was deployed in Network Intrusion Detection System (NIDS) in different scenarios. The techniques were performed on large and varied datasets and gave 100% detection rate approximately, with false positives at 0.9 %. To conclude, Proxy server logs of a campus LAN in addition to router traces, machine learning and data-series analysis are used to detect

Anomalies by abusive Internet access, DDoS attacks and systematic downloading.

(Merlette D, Pruthi DP. 2013) author’s focused on using Adjustable Piecewise Entropy (APE) to detect abnormal traffic behavior and implement it in a multi-node Hadoop platform with five servers in a university campus. A MapReduce bases Adjustable Piecewise Shannon Entropy (APSE) which is an instance of APE and is implemented on Hadoop. Two round MapReduce is implemented to handle larger volumes of data in IPFIX format, and APSE value pairs computed for each bin time. The first round of the MapReduce includes algorithms to get essential flow features and count the total number of occurrences in each time bin. Similarly, round two also includes mapper and reducer algorithms. The proposed Adjustable Piecewise Entropy mode could obtain APE of traditional entropies. In the Hadoop cluster with five servers, all are connected to a Gigabit Ethernet switch. Flow data is collected in IPFIX traffic format from one edge router for three days. The performance of APSE was validated both theoretically and experimentally and was compared with Shannon entropy to conclude that APSE showcases better performance than Shannon entropy in traffic anomaly detection.

(Shanmugasundaram K. 2003) author’s presented the data acceleration patterns used and the complexities of employed analytics framework. It also demonstrated the development of highly interactive visualisations, bringing the idea of a data acceleration pipeline and analytics to life. The proposed model aims at improving security alerts by training the system with network agent behaviours, thus figure out the extent to which the network is malicious. A data acceleration framework was used to analyze and implement the findings by developing a model in a distributed parallel manner. This article summarizes how machine learning algorithms were implemented to learn typical behaviors and how the machine itself develops security protocol and implementing it to a large dataset. The hurdles that data acceleration faces were that of movement, processing, and interactivity of data. The model proposed took inspiration from existing research in this field, explain the algorithms and environmental setup, exploring the discovered insights. It provided new insight on already existing patterns as well.

****

**Fig .(7).** Illustration of Hashdoop Architecture

(Ren W. 2004) In particular, the author zeroed in on Hashdoop, a MapReduce architecture that use hash functions to partition traffic while maintaining its underlying structures. By this, it benefits the detection of abnormal traffic behavior by distributed computing infrastructure. The anomaly detectors use these spatial and temporal features to derive statistical information. In the proposed model, the splitting of datasets done by a hash function to preserves both spatial and temporal structures.

An overview of Hashdoop is shown in Figure.7. After the data splits, detectors detected anomalies in the smaller chunk of data created by hashing and reported to network operators.

The traffic traces were sliced longitudinally into buckets using IP address as the key. In this process, the time duration and flow remain consistent. For accuracy, hashing is done using both source and destination IP address as key one by one thereby ensuring the same transmission of data into a single bucket. The function used for hashing is the CRC algorithm. The typical application of the CRC algorithm is traffic load balancing. Almost every Anomaly Detection System (ADS) applies to Hashdoop for mapping due to its simplicity.

The data set used was real-life data with the link operating at 18Mbps and was later updated to 150 Mbps. Hashdoop, when applied to the provided dataset, increased packet throughput from 360 kpps to 1.27 Mpps and from 25 kpps to 375 kpps, respectively, resulting in a fifteen-fold reduction in processing time. Anomaly detection results improved using Hashdoop due to a more efficient mapping procedure. The only drawback of the proposed model is that if the number of splits more than necessary, anomaly detection can be challenging to execute on small chunks of data in each bucket.

(Jing YN. 2015) author’s literature aims at building upon the progress of tools like Hadoop, Mahout, and Hive to implement quasi-real-time IDS. MapReduce, Hadoop, Libpcap, and Mahout make up of some of the technologies used in this framework. Extraction of required fields out of the packets is done by Tshark that uses the libpcap library to work as a network protocol analyzer and extract data from a live network. The packets extracted may be customized and decoded and then outputted to a file. MapReduce is then used for feature extraction after the sniffer module is completed by extracting valuable information. The tool used for sniffer module used for extracting packets from the network is Dumpcap which forwards the data to Tshark to extract fields corresponding to its feature. The fields extracted are then sent to HDFS. Mahout, a Hadoop library was used to accomplish the scalability. The classification along with clustering is carried out as part of the MapReduce’s tasks. It allows the cluster to generate very efficient results using high computational power. The module mentioned in this literature provides a packet capture module which is scalable and processes high speed and bandwidth of data within 5 to 30s delay. For characterizing the flow of packets, a dynamic feature extraction framework implemented. A peer-to-peer threat detection module which can classify abnormal traffic. The use of TCP relay resulted in packet drops, and more research is going on to minimize it. In the current scenario, the time taken to detect the anomalies ranges in tens of seconds. The performance of the cluster can be increased by adding new nodes and making specific changes in Hadoop cluster which inturns reduces the processing time to less than 10 seconds.

The proposed Botnet architecture provides better fault tolerances and can adapt to changing network through messages sent in the form of multiple small hops to each bot node. The hops are structured by following Degree Constrained Minimum Spanning Tree. Most Intrusion detection systems based on detection above have some threshold value, and therefore the intrusions having low frequency may get ignored.

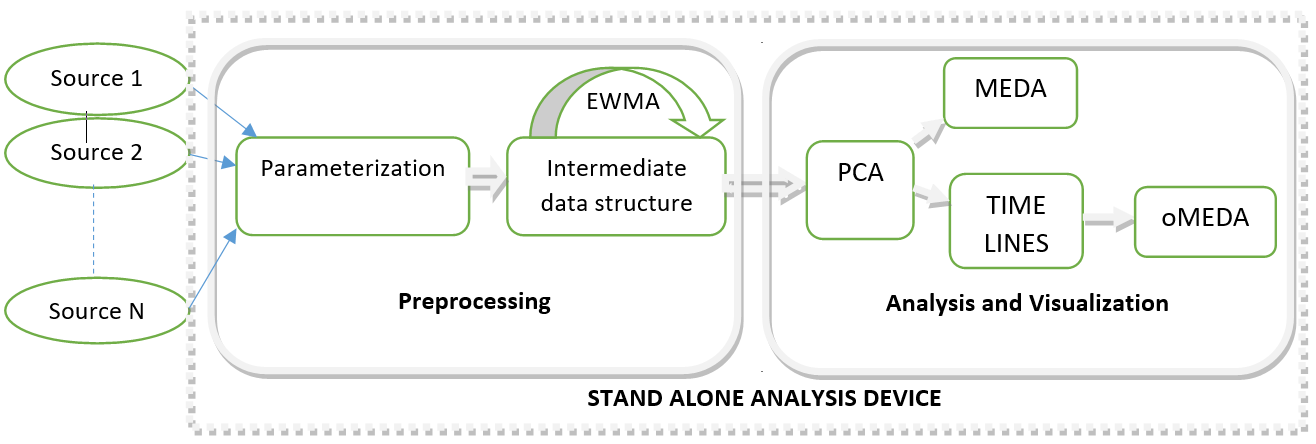
(Tang Y, Daniels TE. 2015) author’s has given us insight into exploring substantial Netflow traffic through Hadoop. It also focused on the aspect of using MapReduce and SQL for common types of analysis. The use of Hadoop SequenceFiles in converting text to binary is useful in minimizing storage usage. Moreover, the MapReduce framework uses the SequenceFile as an

input/output format. SequenceFiles are quite efficient to parse as they are made up of key-value pairs. The parsing of data into simple computational stats is an exploratory analysis which takes place across some features or a complete set of data. The underlying root of calculating these statistics is filtering and counting. To calculate the extreme points is also an important aspect in the analysis. The mapping algorithm is more or less the same as before but the reduce algorithm allows the user to find the Top-k records and arrange it in a list array and the list keeps on getting updated as it keeps on getting out from the mapping algorithm. The Top-K list after the reduce function is complete final list. The reduce function used multiple times, and it stores an array of length k (mostly a small number), and then the array list is initialized and finalized. The focus of this analysis was to detect attacks such as the watering hole in the Netflow dataset by recognizing patterns. For this, several algorithms executed in sync that includes finding Top-k operations like counting, and aggregation.

There are three parameters based on which the performance of Hadoop implementation was evaluated are: exploratory analysis based on SQL on a public cloud, pattern recognition and MapReduce based exploratory analysis on server with multiple cores. A three node cluster with each node of M1 large type possessing four virtual cores and RAM size of 15GB. Apache Hadoop 2.2.0 used along with the default Amazon Web services (AWS) configuration. It was worth noting that space and time significantly optimized by converting text to binary in HDFS using SequenceFile. Shared state aggregation was also used, and noticed that determining the exact number of reducers is essential corresponding to a specific task to achieve good efficiency. For aggregation, combiners were used. AWS and HiveQL with RC File format was used, and it proved to be as effective as Sequence Files.

1. **Analytics Strategy**

(Wang D. 2017) author’s literature talks about introducing a new framework for forensics in Big Data environments, as shown in Figure.8. The proposed idea hurdles the problem of Big Data 4Vs (Variety, Veracity, Volume and Velocity) against basic 3Vs (velocity, veracity and volume). Unstructured data is converted to structured thereby seeing to the variety problem. The Principal Component Analysis (PCA) is a potential method that tackles the low veracity problem as it can extract features from high dimensional sets of data. A kernel computation form of PCA is used to tackle its limitation of working on low sizes of data as it avoids large computations and promotes parallelism.



**Fig .(8).** Analytics Strategy with Methodology and Tools

The proposed framework compared to the VAST 2012 challenge and “a growing botnet infection” was identified in the network. Background activities were reported to be recognised by both Minimum Entropy Deconvolution Adjusted (MEDA) and Optimal Minimum Entropy Deconvolution Adjusted (oMEDA). The timelines or oMEDA could pinpoint the location of the event which helps in gathering more information on the case. Hence, MEDA and oMEDA should use in tandem in analyzing existing active networks. Using the proposed framework allows dealing with different data directly and concisely, making it a good match for security monitoring environments. It also makes the environment feasible to process large volumes of data and moreover, as the information increases, the tool performs better. The number of events to be analyzed in Big data significantly reduced too.

The tool used for sniffer module used for extracting packets from the network is Dumpcap which forwards the data to Tshark to extract fields corresponding to its feature. The fields extracted are then sent to HDFS. Mahout, a Hadoop library was used to accomplish the The dataset used was of the VAST 2012 mini challenge 2 to verify its performance. The proposed framework proved to be outperforming the proposals in the Vast challenge in 2012 and tackles the three challenges of veracity, volume and velocity effectively.

(Kim JS, Kim M, Noh BN. 2014) author’s focused on a new technique for the investigation of Random-UDP flooding attack. In this model can find the source of these attacks. The model works in three stages, (i) standard transmission of data (ii) flooding attack and (iii) flood source recognition.

The working of this framework consists of 3 parts. It analyses the normal flow of UDP. For this, it (proposed technique) analyzed a user on a travel site, where the website finds the cheapest travel package and displays it on the screen tempting the user to use the link. Then it generates a DoS attack using Random UDP flooding. The user of the website is harmed if the web server which has now become a bot, extracts the user information like IP address and floods it by sending UDP datagrams, denying the use of resources and services. Finally, for analyzing the attack, Wireshark tools were used to capture packets sent in the flooding attack, and it helped to identify the attack source.

(Liu Z, Feng D. 2015) author’s focused on a new model which provides a set of ML classifiers used upon a stream of sessions that come up with a specific device and figure out the type of device (PC, a known IoT device or smartphone). For providing training data, traffic data labelled and gathered from 9 different IoT based devices, smartphones and PCs. Supervised learning was used to obtain a multi-stage meta-classifier. This classifier was capable of differentiating whether the data transmitted is from an IoT device or not. To evaluate the model, sample data was collected from all types of devices mentioned above, and they were connected to a WiFi access point. For analysis, their traffic data was stored in the pcap file format. Identifying the device works in three phases. Initially, binary classifiers for a single session is introduced. Then the optimal threshold for these classifiers is found, and lastly, their size is determined. The proposed model for identifying the type of IoT device by using network data is evaluated. It was clear that there is a difference between IoT and non-IoT devices based on their IP addresses. The proposed model was able to identify smartphones with 100% efficiency by studying the “user agent” HTTP property. Single session classifier was used to identify PCs, and its efficiency was close to 100%. IoT devices were identified with 99.28% efficiency with parameters such as model and brand. Thus, the proposed model can be used in real-world organizations to identify any authorized or unauthorized connections of IoT devices to the network.

(Anaya EA. 2019) author’s proposes a new traffic-aware system to choose essential nodes to patch for an IoT based system with constraints on process times and patching resources. The proposed model was compared to a mobile social network’s real-time traffic, which consisted of 60 devices and 1750 APs. Users can communicate with each other through (i) infrastructure link via APs and (ii) direct link within transmission range. Patching can be done to keep the infrastructure links safe while the direct link still poses a threat. Instead of directly patching IoT devices already compromised, the IoT devices were secured by patching essential nodes based on traffic volume to avoid huge losses and avoid threat propagation thereby restricting it to direct links only. Experiments were conducted to verify the efficiency of this patching system, and some exciting new aspects of IoT security gazed on.

New models to prevent the propagation of malware were built upon based on this study where constraints like limited resources and non-user friendly interface still exist. The results of experiments on real-world datasets were still encouraging despite these problems.

(Rekhis S, Krichene J, Boudriga N. 2018) author’s proposed a new model called Stacked Auto Encoder (SAE) that is a stack of autoencoders, is based on traffic flow prediction using deep learning. A broad network is created using auto encoders as the roots of this model. This model is trained layer by layer while getting to know the features of traffic flow. The model parameters were then updated to increase the prediction efficiency. The dataset used to be on a freeway system with 15000 detectors and data collected every half of a minute. The dataset was updated every five minutes. The final dataset contained data collected over the period of 3 months where the first two months of data was used as the training data, and the rest for testing data. It was found that SAE had a higher prediction accuracy than the other four current models (Support Vector Machines (SVM), Radial Basis Function (RBF), Neural network (NN), and Random Walk (RW) forecast) when compared to SAE's performance.

(Lin C,. 2019) author’s focused on a new model in Big Data called Device to Device (D2D) Big Data, whose purposes to promote wireless one device to another connectivity and communication among users efficiently for promotion of content effectively along with implementation of intelligence offloading. The existing models of the short-range wireless device to device communication and content sharing was based on unrealistic assumptions, and small-scale data, whereas the proposed model has the same application over large datasets (approx. 3.5TB) using a popular device to device sharing app which is used by 850 million users in 13 weeks. Features like online behavior, location, meeting dynamics, privacy policies, content sharing are extracted and then analyzed, and then the device to device Big Data platform is evaluated by predicting content sharing habits. It also plans to reduce the use of mobile traffic with big data techniques. Large volumes of content can also be shared with the proposed model. For future research purposes, the focus lied in efficient prediction of the content shared and generated recommendations that were more meaningful and popular. Multidisciplinary collaborations with Mobile Network Operators’ (MNOs’) also have a bright future regarding D2D Big Data. Various vendors such as the government, content service providers and other industries can make effective use of this model.

1. **Cyber Forensic Techniques**

This section explains the standardized set of techniques that are used each phase of the cyber forensics process, namely, collection, investigation and dispensing.

**Data Collection Techniques**

***Disk Mirroring.*** The offline disk mirroring includes creating a duplicate image of a storage or logging disk. That is done to maintain the integrity of data in the original data sources. For instance, the analyst receives a confiscated disk from the crime scene for evidence extraction. So, it becomes the primary duty of the analyst to create an image of the original disk for analysis and not to start working on the original copy. Seized item's original fingerprints are required in the court.

***Network Sniffing.*** Statistics show that more than 90% of the cybercrime occur on the wire. Network sniffing is of two distinct types: active sniffing and passive sniffing. When the switched network relies on an injected packet in the network for sniffing, is called active sniffing. Whereas sniffing that is done through a network, in which all packets flowing on the wire are scanned and captured, is called passive sniffing. Traditionally the passive sniffing was quite useful, as most of the network has the same collision domain. Since it is now obsolete, nowadays it is exhibited by SOHO networks. Hence, active sniffing is mainly used for scanning the network.

***Network Mapping.*** It is getting more complicated with the introduction of IPv6. Especially in case of the real-time corporate environment, distributed geographically, network traffic topology is logged with servers, specially designated for the task of network mapping.

***Encryption/decryption.*** There are a lot of advances and sophisticated algorithms for storing or hiding the information. So, every single piece of information retrieved during the confiscated process, is crucial and may contain crucial evidence for the court of law.

**Data Investigation Techniques**

This section illustrates the techniques proposed by different researchers for investigation of the crime, using different aim and infrastructural setups.

***Trackback.*** Topology assisted Deterministic Packet Marking (TDPM) uses packet marking with hash codes mechanism to traceback DoS attacks. The router uses hash values in the packet to compare it to the victim machine. Using TDPM approach, the investigator recursively identifies the ingress router or network node that generates substantial malicious packets using the breadth first search algorithm.

The intruder's first hops may be traced back to their point of origin with more efficiency thanks to the Network Forensic Evidence Acquisition (NFEA) scheme's straightforward packet tagging method. Collaborative Forensics Scheme does not examine router data or packet headers, while NFEA does (CFS). In this way, the system's throughput is less likely to drop. The NFEA reduces the burden on network performance by only tracking routers at the network's periphery. However, if the attacker conceals their MAC address, NFEA may confuse investigators as to where the attack originated. It's important to note that IPv6 addresses are incompatible with NFEA. When the number of table entries on the edge router grows, this method becomes problematic due to its impact on router memory. Due to these obstacles, it has difficulty scaling to large networks and maintaining dependability.

The Time-To-Live (TTL) field in an IP header is used by the lightweight IP traceback protocol (LWIP). Accordingly, LWIP may probe large-scale Distributed DoS assaults to piece together the assault chain. LWIP uses tree analysis approach for analysis and routing filter algorithm for storage management to give better accuracy on colossal traffic, even if the intruder is aware of running IP traceback scheme on the victim machine for detection. The problem with the LWIP method is that the target time to live (TTL) number decreases as traffic travels from router to router before reaching the victim node. The problem with the LWIP strategy is that the TTL value may be tampered with and set to zero before it reaches its destination, which can be disastrous. Also, it is only used for IPv4 headers, but not for IPv6 packets, which has started to dominate the Internet domain.

Scalable Network Forensic (Scalable-NF) scheme specially designed for stealthy selfpropagating intrusions to traceback the origin of the attacker, is based on massive, reliable historical network traffic trace records. Two significant issues identified in (Wang D. 2017) are addressed in Scalable-NF, namely complexity due to size of data, and low data integrity issue faced by forensic analysts. Scalable-NF deals with space and time challenge efficiently and provides high accuracy to traceback stealthy attacker’s origin. Logged network traffic traces are categorized into malicious traffic and regular traffic. Differentiating between regular traffic, which occurs between endpoints without intervention, and malicious or attack traffic, which is launched by an intruder that has to be probed, may be difficult. The employed random moonwalk technique aids in tracing the attacker's origin by putting up a casual tree, with the root node representing the intruder's point of origin and the object of the forensic inquiry. The issue with this approach is its depending on the accuracy and availability of historical data. The second challenge is its complexity due to the separation process of legitimate and malicious traffic. Last is the missing automation process of capturing real-time traffic and visualizing the clear image of the scenario.

Hopping Based Spread Spectrum Technique (HB-SST) is a code, frequency and time hopping-direct sequence spread spectrums technique, used to tackle attacks through anonymous communication on the network and trace back the source of an intruder. However, besides these, there are more frameworks used to aid the user in remaining anonymous while online, such as. The HB-SST embed invisible pseudo-noise code with regular traffic, and this pseudocode is used to monitor malicious activity/traffic. However, scalability and time are again the limitations, especially with large datasets. Further, we cannot ignore the possibility of the pseudo-code getting identified by the intruders. Also, frequent shuffling among different frequency bands in frequency hopping-direct sequence spread spectrum, for optimizing the communication are further complicating the investigation process.

IP Traceback Protocol (ITP) is real-time online and offline periodic analysis technique for traffic, flowing through the router, to find the origin of malicious packet. A router’s hash table is used for real-time analysis and compress table is used for periodic analysis. For periodic traffic investigation, router generated compressed hash tables are stored in a database. The IP packet's timestamp and MAC fields are used in ITP trace back, which keeps the integrity by using a hash function, intact. MAC field in the ITP technique helps in finding the routers through which packet has passed through, and a timestamp is used for attack’s retransmission. For Big Data investigation in a distributed environment, ITP has high latency, and vast resources are required. Another issue with ITP is that the unknown attack patterns are untraceable due to the un-updated list available at router.

***Attack Graph-Based Network Forensics Techniques.*** Attack graphs also analyze impact and scale, figure out network attack, collect evidence and harden outer network layer, with the best cost. However, effective embedded visualization in attack graphs, especially in large complex networks was required.

Real-time Attack Visualization Environment (RAVEN) architecture has effectively made it easy for the investigators, through visualizing interfaces. RAVEN approach uses human computer interaction platform with an extensive collection of gesture controls and give multiple investigators a collaborative analytical environment. Thus makes user-attach graph interaction, manageable. However, the issue with Real-time Attack Visualization Environment (RAVEN) is that has non real-time visual support, especially while dealing with the real-time environment. Another issue is the lack of a composite layout that is required in multiple attack parallel paths and reconstructing these paths. Also, the support for efficient data mining is not there, which makes it difficult for the investigator to do the retrieval of relevant information from the entire network's data.

***Distributed Network Forensics Techniques (Distributed-NFT).*** From scattered data agents at different locations, traffic data is collected by forensic servers for analysis. Distributed forensics techniques identify attack origin, react to emerging responses, and an assortment of evidence. However, distributed based NFT, have a considerable overhead for securing the server (forensic) from intruders that is scattered by the forensics network. Various distributed NFT are explained below in brief. Usually, manual analysis of hosts logs and packets leads to less response time, an erroneous mechanism for logging, and logs synchronization. ForNet deals with these issues and generates evidence to be presented in the court of law. Additionally, the ForNet's application, SynApps works on each network node (switches/routers) to extract evidence flowing through the network and every logs event on the network for a longer time. Packet header information is more admissible than payload for collecting evidence of a crime.

The issue with ForNet framework is the storage of massive data in a vast network. Many malicious traces remain undetected, due to the use of lightweight IDS. Usually, undetected attacks are DoS, modifying packet headers and utilizing forensic server’s resources, through rogue queries. The network logs in a node’s log file are at modification risk from intruders, on insecure communication channels.

Investigating the different network system logs is the primary focus of the Distributed Agent-based Real-Time Network Intrusion Forensics System (DRNIFS), which is a mechanism for providing a rapid reaction to adaptive assaults. DRNIFS fixes the problems that non-real-time frameworks like ForNet and On A Reference Model of Distributed Cooperative Network have with their lack of functionality in real time environments. The Distributed Network Information and Forensics System (DRNIFS) is made up of four different modules: network monitors, network investigators, network agents, and forensic network servers. Coordination between modules helps in performing real-time analysis, without any data loss, as and when an attack occurs. Network agents extract useful information about data encryption, digital signatures, and packet header information from the host forensic server through secure sockets layer. Network monitor collects network traffic, timestamp attached to every packet, and store it on disk. Forensic server changes dump and filter rule. An investigator needs to identify and a mass malicious event. Network's investigator’s prime goal is to develop an intruder’s biography such as a logical and physical address, phone numbers and machine configuration.

***Intrusion Detection System based NFT.*** Usually, dynamic forensics, forensic explanation, analyze network intrusion data, and reliability of the evidence, are described by using logging approach. IDS approach is employed by many Network Forensic Tools to help investigators, in their process of finding security breaches in the network. For multiclass intrusion detection in network traffic, an ensemble based neural network approach has been suggested by [9]. This approach train by learning from interconnected neural networks. Some of the NFTs are briefly explained in the subsequent para.

Analytical Intrusion Detection Framework (AIDF) has been proposed for integrating forensic analysis and coordinates alarm data from IDS sensors for intrusion detection, resulting in the legal description based on the non-documented signature rule and IDS alerts. AIDF uses probabilistic methods that reveal covered information and pre-structured possible attacks, thus reducing the scope of network attacks by intruders. That is done by matching each network packet, with pre-defined signature’s encoded patterns. Snort is a useful probabilistic tool used by AIDF to trigger a rule, if network traffic data match with an encoded pattern in the signature rule. The similar attacks can be detected, understood and prevented, by continually updating rules, with the network investigators.

The problem with the AIDF framework is limited to, sufficient rule base for unhandled hidden data, which is necessary for generating future intrusion detection alarms. Adequate checking rules save investigators time and increase accuracy in alerts. AIDF is especially useful for raising intrusion alarms, in a real-time environment.

**Evidence Dispensing Techniques**

Briefly, introduces the techniques used by the investigator for making the evidence court ready. This section introduces some of those techniques.

***Proof Visualization.*** Multiple existing techniques and framework that have embedded proof visualization module are discussed, such as EnCase, FTK, Nuix, Oxygen, and XRY. Notwithstanding the significance of visualization in the forensic analysis process, this has all the earmarks of being an understudied region.

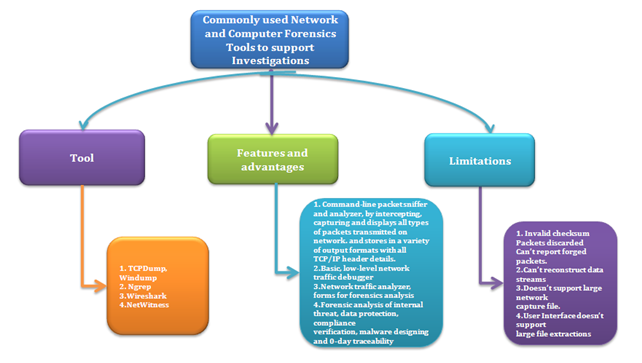
***Crime Simulation and Reconstruction.*** With digital crimes, the focus is usually on the identification and cataloguing of digital evidence using an item's characteristics, but not on series of the events that resulted in a crime. So, the focus on the identifying the crime sequence is equally crucial regarding presenting the evidence in the court of law. Usually, the crime event reconstruction model consists of five steps: a) Evidence Examination; b) Role Classification; c) Event construction and testing; d) Event sequencing; e) Hypothesis testing .

The first step is to examine the evidence item thoroughly. It includes all steps from recognizing the potential items from the objects seized from the crime scene to elicit the pieces of evidence. The second step after each piece of evidence is recognized, is to interpret the condition of an evidence item and the conceivable occasions it engaged with the sequence of events. The role classification step intends to recognize the features that are linked to the crime and are the outcome of an incident or the cause of an episode. The third phase is initiated with the gathering of objects and their functions and properties and conclude with a bunch of unordered or partly established events that may have happened, which led to the crime. Then each possible event or scenarios are tested to check if the event may have caused the assault. Fourth step is event sequencing, in which series of events are identified in the previous step, should to correlate. If the control stream of an executable or process is known, then that knowledge can be used to sequence happenings that led to a charge. Finally, the fifth step in the sequence is hypothesis testing. The phenomenon of hypothesis testing for cybercrime scenes is similar to usual crime scenes. In the final stage of the investigation, a hypothesis established about the absent events after the order of event successions. A confidence level is attached to each hypothesis, and each hypothesis are examined to identify which was most likely, and the evidence refuted. Knowledge of ways to launch cyber-attack plays a vital role in this phase. It helps to identify all possible sequences the events that may have occurred, resulting in an assault. Stephenson’s Petri net hypotheses model for hypothesis testing is used in this phase. Researchers have significantly worked on event reconstruction using hard disk (using signatures, using event correlation and using miscellaneous techniques; using memory evidences and using network evidences. Moreover, at the end of the network and computer forensics process, these simulated series of events need to be presented in a legal document format along with the origin trace back and other pieces of evidence, to prove the occurrence of cybercrime and the black hat, guilty.

***Legal Document Formatting.*** The final step in the investigation of the cybercrime is to bring the digital evidence along with the physical evidence, like photographs, sketches of events, call graphs, and data records. The analyzed and documented digital evidence shuffled, in a legally accepted format, to be acceptable in the court of law. Also, the pieces of evidence that need to adhere and the process to elicit them may be the same in a distributed geographically divided area, or it may differ from one jurisdiction area to another. Reason being the variance in the acceptability and advancements in the cyber laws from one country or state to another. That is, cyberstalking may be legal in one and illegal in another. Moreover, also the sentence in cyber laws may differ, like misusing credential in one state may be penalized for six months, and three years in another state.

1. **Cyber Forensic Tools**

The standardized set of available tools for each phase of the cyber forensics process, is illustrated in this section. These tools facilitate or support the task of the investigator in the analysis. Most of the sophisticated legally accepted frameworks used by legal cybercrime team for forensic investigations, consist of bunch (kit/suite) of multiple tools used at multiple stages. Well-known suite is used at multiple phases of cyber forensics. Hence, the suite/toolkits/bunch are listed below in Figure.9, with corresponding stage/phases, at which the toolkit is used.



**Fig .(8). Commonly used Network and Computer Forensics Tools to support Investigations**

1. **Conclusion**

Forensics is a set of activities applied to a possible entity that involves the collection, analysis, and presentation, to extract the evidence acceptable in the court of law. Network and Computer Forensics is a field, which deals with an in-depth analysis of the captured data, to reveal evidence, admissible in the court of law. The audit, examination of Network and Computer data for information congregation, encroachment detection, and legal evidence presentation is an integral part of the forensic process, Ontologies and related technologies for Big Data network traffic analysis have been used by the majority of researchers in prior work, especially in support of the problem of optimising results while dealing with scalability and distributed computing, as described in this chapter.

**REFERENCES**

1. S. Vinjosh Reddy, K. Sai Ramani, K. Rijutha, S. Mohammad Ali, and C. H. Pradeep Reddy, (2010), “Wireless hacking-a WiFi hack by cracking WEP,” in *ICETC 2010 - 2010 2nd International Conference on Education Technology and Computer*, vol. 1., Jan 2010.
2. S. Almulla, Y. Iraqi, and A. Jones, “A distributed snapshot framework for digital forensics evidence extraction and event reconstruction from cloud environment,” in *Proceedings of the International Conference on Cloud Computing Technology and Science, CloudCom*, vol. 1, pp. 699–704, Mar 2013.
3. M. Uzun and O. Abul, “End-to-end internet speed analysis of mobile networks with mapReduce,” in *2016 International Symposium on Networks, Computers and Communications, ISNCC 2016*, May 2016.
4. C. Wang, C. Chi, W. Zhou, and R. Wong, “Coupled Interdependent Attribute Analysis on Mixed Data,” *Proc. Twenty-Ninth AAAI Conf. Artif. Intell. Coupled*, pp. 1861– 1867, Dec 2015.
5. R. R. R. Robinson and C. Thomas, “Ranking of machine learning algorithms based on the performance in classifying DDoS attacks,” in *2015 IEEE Recent Advances in Intelligent Computational Systems (RAICS)*, pp. 185–190, Aug 2015.
6. G. Shrivastava and B. B. Gupta, “An Encapsulated Approach of Forensic Model for digital investigation,” in *2014 IEEE 3rd Global Conference on Consumer Electronics (GCCE)*, pp. 280–284, March 2015.
7. M. H. Mate and S. R. Kapse, "Network Forensic Tool -- Concept and Architecture*," 2015 Fifth International Conference on Communication Systems and Network Technologies*, pp. 711-713, Jan 2015. doi: 10.1109/CSNT.2015.204.
8. R. Hunt and S. Zeadally, "Network Forensics: An Analysis of Techniques, Tools, and Trends," in *Computer*, vol. 45, no. 12, pp. 36-43, Dec 2012. doi: 10.1109/MC.2012.252.
9. Pallavi Asrodia and Vishal Sharma, "Network Monitoring and Analysis by Packet Sniffing Method", *International Journal of Engineering Trends and Technology (IJETT),* pp.101-120, Feb 2013.
10. Merlette D, Pruthi DP, Network security; NetDetector: identifying real threats and securing your network. [Online]. Available: <https://www.niksun.com/>, 09 Dec 2013.
11. Shanmugasundaram K, Memon N, Savant A, Bronnimann H ForNet: a distributed forensics network. In: Gorodetsky V, Popyack L, Skormin V (eds) *Computer network security*, vol 2776. Springer, Berlin/Heidelberg, pp 1–16, Jan 2013.
12. Ren W, On a network forensics model for information security. *In: 3rd international conference on Information Systems Technology and its Applications (ISTA 2015)*, Utah, USA, pp 229–234, April 2015.
13. Jing YN, Tu P, Wang XP, Zhang GD Distributed-log-based scheme for IP traceback. *In: The fifth international conference on Computer and Information Technology (CIT’ 15),* Shanghai, China, pp 711–715, Feb 2015.
14. Tang Y, Daniels TE , A simple framework for distributed forensics. In: *25th IEEE International Conference on Distributed Computing Systems Workshops (ICDCS 15),* Columbus, OH, USA, (2015), pp 163–169, June 2015.
15. Wang D, Li T, Liu S, Zhang J, Liu C, Dynamical network forensics based on immune agent. In: *Third International Conference on Natural Computation (ICNC 2017),* Haikou, Hainan, China, pp 651–656, July 2017.
16. Kim JS, Kim M, Noh BN A fuzzy expert system for network forensics. In *Computational science and its applications*, vol 3043. Springer, Berlin/Heidelberg, pp 175–182, Feb 2014.
17. Liu Z, Feng D, Incremental fuzzy decision tree-based network forensic system. In: Hao Y, Liu J, Wang YP, Cheung YM, Yin H, Jiao L, Ma J, Jiao YC (eds) *Computational intelligence and security*, vol 3802. Springer, Berlin/Heidelberg, pp 995–1002, Jan 2015.
18. Anaya EA, Nakano-Miyatake M, Perez Meana HM, Network forensics with Neurofuzzy techniques. In: *52nd IEEE international Midwest Symposium on Circuits and Systems (MWSCAS ’19)*, Cancun, Mexico, pp 848–852, Feb 2019.
19. Rekhis S, Krichene J, Boudriga N, DigForNet: digital forensic in networking. In: Jajodia S, Samarati P, Cimato S (eds) IFIP TC-11 *23rd international information security conference*, vol 278. Springer, Boston, pp 637–651, Jan 2018.
20. Lin C, Zhitang L, Cuixia G, Yingshu L, Modeling and analyzing dynamic forensics system based on intrusion tolerance. In: *Ninth IEEE international conference on computer and information technology*, pp 230–235, March 2019.