CHALLENGES AND OPPORTUNITIES FOR INTRUSION DETECTION SYSTEM IN CLOUD COMPUTING ENVIRONMENT

HESHAM M. EL MASRY[1], AYMAN E. KHEDR[2], HATEM M. ABDUL-KADER[3]

Faculty of Commerce and Business Administration, Future University in Egypt (FUE),
Cairo, Egypt,
Faculty of Computers and Information Technology, Information Systems Department, Future University in Egypt (FUE), Cairo, Egypt,
Faculty of Computers and Information, Menofyia University
Menofyia, Egypt,
E-mail: Hesham.elmasry@fue.edu.eg , Ayman.Khedr@fue.edu.eg, hatem6803@yahoo.com

ABSTRACT

The paper suggests an anomaly multi Phases intrusion detection technique for discovering the zero-day, fast-spreading and complicated network attacks on the cloud platform with the least amount of false alarm rate. Cloud computing is commonly regarded as an attractive business model, because it minimizes expenditure and its costs are directly related to use and demand. But the distributed nature of cloud computing environments, their vast aggregation of resources, large user access, efficient and automated resource sharing allow Intruders to use cloud to their advantage. The objective of this paper is to use the comparative approach to review, analyze and evaluate all of the existing Intrusion Detection Systems (IDSs) types, techniques, algorithms and all of the previous attempts related to securing and detecting the attacks on the cloud environment. The article concluded that the existing IDS techniques and algorithms are not capable in detecting the unknown attacks with minimum false alert rate in cloud platform. The effectiveness of the new intrusion detection technique can be evaluated by measuring the accuracy and false alert rate.

Keywords: *Intrusion Detection Systems (IDS), Hybrid Based IDS, Binary Classification and K-NN, PSO and Machine Learning, Genetic Algorithm and Artificial Neural Network.*

1. INTRODUCTION

Cloud paradigm is currently an efficient solution for modern systems according to the immense need for strong infrastructure with the lowest possible cost [1] and availability for continuous tuning [2]. The efficiency of cloud paradigm environment has led to the deployment of efficient systems under the cloud environment to ensure its efficiency and effectiveness [3] [4]. Moreover, cloud paradigm also has different research directions besides such as green computing and intrusion detection [5].

An Intrusion Detection System (IDS) is a program or hardware built to stand in the way of any malicious pastime, device or network attacks. IDS gather info from extraordinary assets within a pc or a network together with device command, device log, system accounting, safety log and network log. Then, it analyzes them to choose out attainable security violation, and eventually, it issues associate degree conscious of the computer user to wear down the intrusion.

The dispersed structure and nature of cloud environment makes it one of the most defenseless and appealing environment for the intruders. Intrusion detection systems may be used by regularly analyzing the logs, network traffic and configurations to improve the security of these systems. However, the main problem with this paper is to proving that the Current intrusion detection techniques are not suitable for cloud environments because they cannot locate the hidden trail of the attack. The network-based IDS is unable to detect any incident in the case of encrypted node contact, so if the hypervisor is broken the attacker can gain management over the virtual machines mounted.

The compromised hypervisor is utilized by attackers to realize management over the host. Since the IDS procedures have not been built beneath thought of the specific setting of the virtualization, they don't give the same security in such situations. There are other trade-offs that need to be addressed when installing IDS in the virtual world, mainly due to their inability to examine the operating systems ' internal workings. Given the tremendous advantages that virtualization provides, there are a range of security considerations related to this. This gifts a range of recent issues that weren't present in a very typical computing environment.

The main purpose of this article is to prove the lack of existing IDS techniques and algorithms in detecting the zero-day attacks with minimum false alert rate in cloud environment, given the fact that there are already a few survey papers targeting this area. A large portion of these study papers are out of date or stinting in their inclusion on cloud-based IDS.

2. Research Methodology

Methods of study are broad word. Although data collection methods and data analysis represent the basics of research methodology, you will need to tackle a number of additional elements within your research framework. The key elements of research methodology anticipated to be discussed in this review paper include research theory, research type, research approach and data collection methods. Let's look at each one separately for a brief moment.

2.1 Research Theory

Research theory is concerned with clarification of the belief about the existence and source of information. All studies are based on some form of assumptions about the environment and the way the environment is understood. Research theory, in simple words, refers to your belief about how to collect, analyze and use data. Accordingly, clarifying the theory of research is a starting point for selecting research methods. This paper will use the positivism approach which is focused on evidence and quantitative data.

2.2 Research Type

The qualitative research type has been used to reviewing, analyzing and evaluating the existing intrusion detection techniques for cloud platform. It also seeks to contribute, without immediate practical consequences, to the overall reach of research awareness. Fundamental studies results cannot be used for addressing immediate and complex market problems.

2.3 Research Approach

Regardless of a spread of variables, resembling the realm of study, analysis theory and so the quality of the analysis issue, the inductive methodology is utilized to hunt out answers to the analysis question exhibit at the start of the analysis cycle.

2.4 Research Question

The main research question is whether the current IDS techniques and algorithms are capable to detecting the unknown attacks with minimal false warning rates on the cloud platform or not.

2.5 Data Collection

The secondary data approach is used to gathering the information which other researchers have already collected from previous scientific studies.

3. BACKGROUND

Identification of intrusion is a process of detection and response to malicious activities targeting computing and network resources [6]. From the objective perspective the Detection of intrusion is characterized as the process of recognizing and analyzing Different incidents that occur inside the system or network for potential intrusion and malicious response [7]. The objective of an intrusion detection system (IDS) is therefore discriminate efforts at intrusion and preparing for intrusion from ordinary scheme use.

This paper provides a full comparative analysis for all pervious and current intrusion detection techniques and algorithms studies which are focused in cloud computing environment.

3.1 Intrusion Detection System Evolutions

James P. Anderson was a unique computer safety figure He was Gunnery officer in the U.S. Navy in the early to mid-1950s, where he first became interested in cryptography. He joined the Burroughs Corporation as an Advanced System Technology manager by the late 1950s, where he quickly worked on multiprocessor development. He began a one-person computer security consulting company, James P. Anderson Company, in 1966, after a three-year stint at the computer market research company Auerbach and Associates, mainly serving the groups of defense and intelligence. It is arguable that no one had a deeper understanding of computer security than Anderson did in the late 1960s in the early 1970s, when he worked on the Ware DSB Task Force when he led what was called the Anderson Committee and wrote volumes I and II of the Anderson Report [8].

3.2 Intrusion Detection System Characteristics

Throughout the literature, different strategies and types are available for detecting intrusion behavior, and IDS systems Differentiate according to different parameters. By describing those criteria, we will clarify what types of IDSs you are likely to encounter and decide the correct field of application based on each type's advantages and disadvantages. First and foremost, it's possible to differentiate IDSs on the basis of the kinds of characteristics, Prediction performance, time performance, and Fault tolerance. In this case, IDSs may be categorized into network-based, host-based, Distributed-based and Hypervisor -based IDS types.

3.2.1 Prediction performance

Simple performance measurements such as forecast precision are not sufficient for intrusion detection. For instance, network intrusions typically constitute a very tiny proportion (e.g. 1%) of network traffic as a whole and trivial IDS that labels all network traffic as ordinary can attain 99% precision. IDS must fulfill two requirements in order to have excellent forecast efficiency: (1) it must be able to properly define intrusions and (2) it must not recognize legitimate intervention as an intrusion in a system setting. Typical IDS predictive performance assessment measures include detection rate and false alarm rate. The detection speed is portrayed because the proportion of the attacks properly detected and additionally the whole vary of attacks, whereas the proportion of traditional connections and also the total range of normal connections misclassified as attacks is the false alert rate. In practice, evaluating these two steps is very hard, as having worldwide understanding of all attacks is generally infeasible [9].

3.2.2 Time performance

An intrusion detection system's time output corresponds to the complete moment the IDS needs to identify an intrusion. This time involves time for processing and time for propagation. The processing time relies on the IDS processing speed, which is the rate at which audit events are processed by the IDS. If this rate is not large enough, then security incidents may not be processed in real time. The propagation time is the time it takes to propagate processed data to the security analyst. Both times should be as temporary as potential so as to permit ample time for the protection Associate in security analyst to retort to an attack before there has been much damage and to prevent an attacker from changing the audit details or modifying the IDS itself [9].

3.2.3 Fault tolerance

IDS should be reliable, robust and resistant to attacks on its own, and should be able to rapidly recover from effective attacks and continue to provide a safe service. This is particularly true for very big distributed DoS, overflow and various intentional attacks that can also shut down the computer system and hence IDS. This characteristic is very important for the proper functioning of IDSs, as most commercial IDSs run on operating systems and networks that are prone to assaults of multiple types. Furthermore, IDS should also be resistant to circumstances where an adversary may cause a large number of false or misleading alarms to be produced by the IDS. Such alarms can readily have an adverse effect on the system's accessibility, and these hurdles should be overcome rapidly by the IDS [9].

3.3 Cloud-Based Intrusion Detection System (CIDS) Types

In this section a study on types of CIDS is provided by gathering the latest field-relevant research. Our approach is to perform a general evaluation for all kinds of CIDS and apps settings to determine the advantages, limitations and assumptions for each CIDS type.

CIDS varieties will categorize into four classifications supported platform and input data. These are IDS deployed on a number or single machine (HIDS), IDS enforced on one or multiple network machine (NIDS), these varieties are shown in Figure one. IDS deployed on distributed system (DIDS), IDS deployed on a hypervisor system (HIDS).

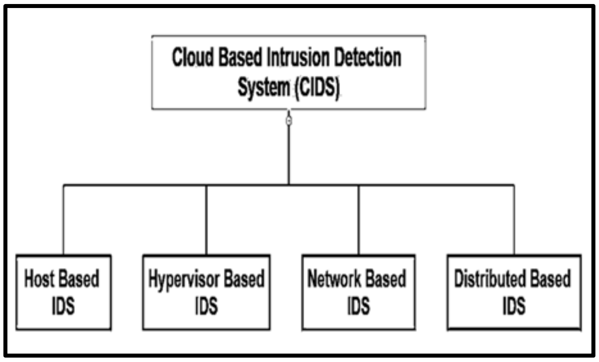


Figure 1: Cloud Based Intrusion Detection System (CIDS) Types

3.3.1 Host – based Intrusion Detection System (HIDS)

HIDS was the first form of intrusion detection to be created. HIDS tracks and analyzes single-host inner computing or system-level operations Such as: device configuration, program operation, wireless network traffic (for that host only) or network interface, system logs or audit logs, running user or server procedures, file access and modification, as shown in Figure 2 describing the HIDS. The capabilities of HIDS include integrity management, event correlation, log analysis, policy execution, rootkit identification, processor , memory, battery and hard disk use, and alert[10][11].

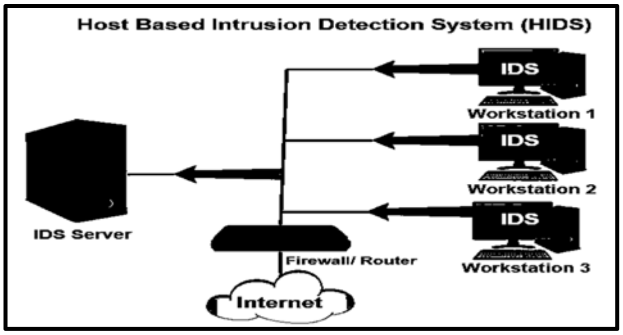


Figure 2: Host Based Intrusion Detection System (HIDS)

Strengths: HIDS tends to be additional precise and fewer false-positive than network-based IDS as a result of it analyzes log files and may so verify whether or not or not an attack has been fortunate. The host-based intrusion detection system wants the installation of programs or Agents on the theme to get the reports that stating whether or not any malicious activity has occurred.

Limitations: The issue with host-based systems is that associate degree inclination to be resource-intensive as a result of they use identical software resources that are mounted thereon and don't have an autonomous package like alternative IDS sorts [12].

Cloud Environment: Since HIDS can analyze encrypted traffic, thus, hypervisor, VM or host HIDS can be implemented to analyze system logs, user login information or access control policies and detect intrusion events.

3.3.2 Network – based Intrusion Detection System (NIDS)

To find suspect operations, NIDS is employed to trace and appraise network traffic on explicit network section. Figure three shows NIDS and therefore the steps it takes to find attacks. NIDS employed in packet level assessment for all network section schemes by checking information science, network transport and application protocol level operations and packet headers to find varied IP-based DOS attacks just like protocol SYN attack, packet attack fragment [13]. NIDS focuses a lot of on vulnerability abuse whereas HIDS focuses on privilege abuse.

Strengths: NIDS prices less and quicker than HIDS as a result of there's no got to retain host-level device programming and it monitors period of time or close-time traffic [14]. Therefore, as they happen, NIDS will discover attacks. However, as a result of it doesn't measure the log theme, NIDS doesn't show whether or not or not such attacks are effective.

Limitations: The problem with NIDS is that it's lowest visibility inside the host machine, and there's no economical thanks to analyze encrypted network traffic for police work attack [15]. Hence, so far, several researches have advanced to make economical ways to sight attacks by NIDS. There are many network intrusion detection product, cherish Snort 16] and NetSTAT [17].

A real-time NIDS instrument. Network-based IDSs may not be able to track and analyze all traffic on wide, busy networks, so they can ignore attacks launched during peak traffic times.

Cloud Environment: In cloud platform, hypervisor or VM attacks are detected by placing NIDS on the cloud server that communicates with external network. That can't detect attacks within a virtual network created by the hypervisor.

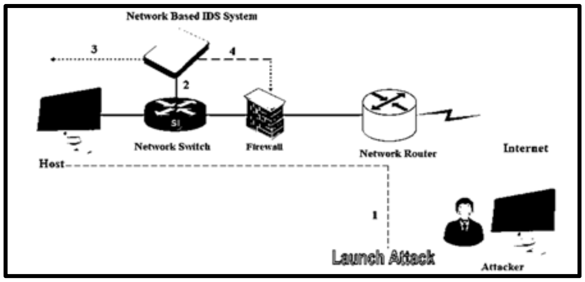


Figure 3: Network Based Intrusion Detection System (NIDS)

3.3.3 Distributed – based Intrusion Detection System (DIDS)

A distributed IDS (DIDS) consists of various network IDS, all of that communicate with one another, or with the core server which allows network monitoring [15]. As shown in figure 4, DIDS is intended to operate in a non-homogeneous setting, which suggests that DIDS is able to aggregating statistics from various sources to discover attacks on a community device inclusive of doorknob assaults and DDoS assaults. In the DIDS device, there are 3 factors which are IDS agent, verbal exchange element, and basic analytics server.

Strengths: DIDS has many blessings [18] [19]. It's going to be used for detection the recognized and so the unknown attacks since it takes the advantages of each NIDS and HIDS [20] [21].

Limitations: Singl e point of failure as for centralized DIDS and high connectivity and computing costs, the central repository can be overwhelmed and difficult to handle.

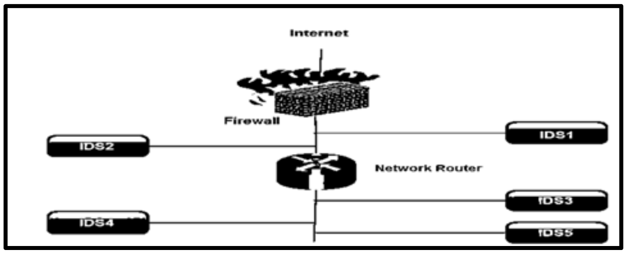


Figure 4: Distributed Based Intrusion Detection System (DIDS)

Cloud Environment: DIDS may be placed at either of two positions: on server process or on user machine.

3.3.4 Hypervisor – based Intrusion Detection System (HVIDS)

Hypervisor could be a platform running hypervisor-based VMs IDS that operates on the hypervisor layer. It permits users to observe and assess links between VMs or between hypervisors and VMs and also the virtual network supported hypervisors [22]. Depending on its specifications, it will maintain and implement numerous security ways for every VM. VMI-IDS are often shaped to run on the virtual machine monitor's privilege domain.

Strengths: The most necessary advantage of hypervisor-based IDS is that the quality of data. VMI-IDS noted a host's hardware and computer code conditions and events and provide a stronger perspective of the scheme than HIDS. Also it enables users to track and evaluate interactions between VMs, among hypervisor and VM and in the digital network primarily based on the hypervisor. Figure 5 demonstrates the architecture of the VMI-IDS. To detect any unusual activity, VMI-IDS observe programs operating in VM [23].

Limitations: All existing HVIDS techniques are detecting the intruder's attacks by deploying an IDS agent on the cloud Hypervisor, so the attacker can access all cloud VMs if the hypervisor is down.

Cloud Environment: HVIDS can be located at any of two positions: at hypervisor or at VM machines.

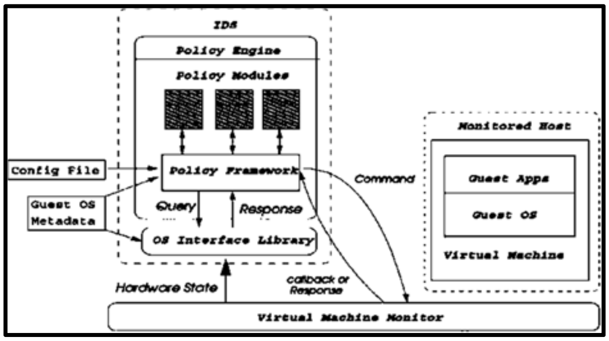
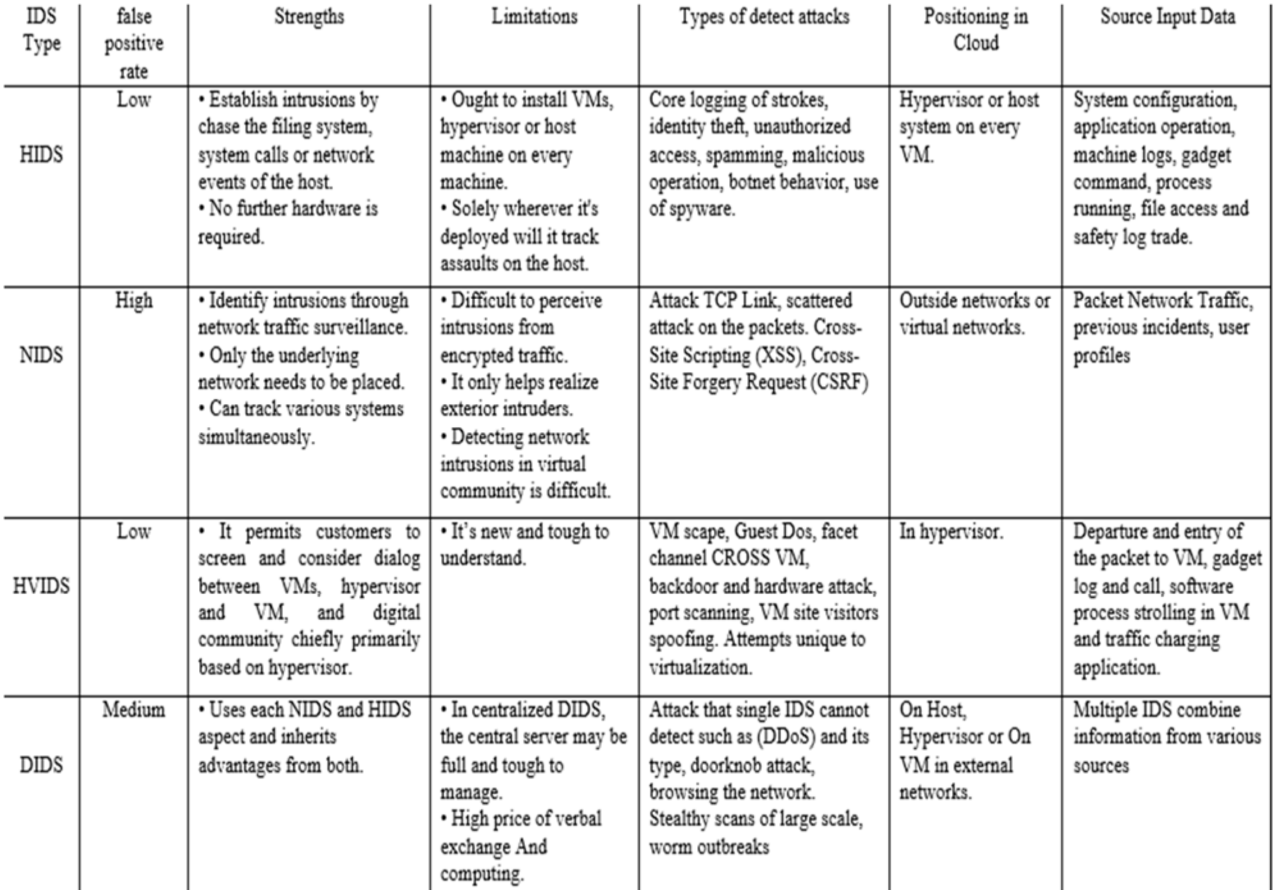


Figure 5: Hypervisor based Intrusion Detection System (VIDS)



IDS Type	false positive rate	Strengths	Limitations	Types of detect attacks	Positioning in Cloud	Source Input Data
HIDS	Low	• Establish intrusions by chase the filing system, system calls or network events of the host. • No further hardware is required.	• Ought to install VMs, hypervisor or host machine on every machine. • Solely wherever it's deployed will it track assaults on the host.	Core logging of strokes, identity theft, unauthorized access, spamming, malicious operation, botnet behavior, use of spyware.	Hypervisor or host system on every VM.	System configuration, application operation, machine logs, gadget command, process running, file access and safety log trade.
NIDS	High	• Identify intrusions through network traffic surveillance. • Only the underlying network needs to be placed. • Can track various systems simultaneously.	• Difficult to perceive intrusions from encrypted traffic. • It only helps realize exterior intruders. • Detecting network intrusions in virtual community is difficult.	Attack TCP Link, scattered attack on the packets. Cross-Site Scripting (XSS), Cross-Site Forgery Request (CSRF)	Outside networks or virtual networks.	Packet Network Traffic, previous incidents, user profiles
HVIDS	Low	• It permits customers to screen and consider dialog between VMs, hypervisor and VM, and digital community chiefly primarily based on hypervisor.	• It's new and tough to understand.	VM scape, Guest Dos, facet channel CROSS VM, backdoor and hardware attack, port scanning, VM site visitors spoofing. Attempts unique to virtualization.	In hypervisor.	Departure and entry of the packet to VM, gadget log and call, software process strolling in VM and traffic charging application.
DIDS	Medium	• Uses each NIDS and HIDS aspect and inherits advantages from both.	• In centralized DIDS, the central server may be full and tough to manage. • High price of verbal exchange And computing.	Attack that single IDS cannot detect such as (DDoS) and its type, doorknob attack, browsing the network. Stealthy scans of large scale, worm outbreaks	On Host, Hypervisor or On VM in external networks.	Multiple IDS combine information from various sources

Table:3 Comparisons Of IDS Displays

3.4 Evaluation Criteria Of Intrusion-Detection Systems

Porras associated Valdes suggested the below three evaluation criteria to validate the effectiveness of an intrusion detection theme [24].

3.4.1 Accuracy

Accuracy deals with adequate assault identification and lack of false alarms. Inaccuracy happens when a scheme of intrusion detection symbols as anomalous or intrusive a lawful intervention in the setting [24].

3.4.2 Performance

The rate of handling of audit events is an output of the intrusion detection scheme. If the monitoring method for intrusion is weak in effectiveness it cannot be detected in real time [24].

3.4.3 Completeness

Completeness is the ownership of a scheme for detecting all assaults by intrusion. Incompleteness happens when an assault is not detected by the intrusion detection scheme. This metric is much harder to assess than the others because worldwide awareness of assaults or abuses of rights is impossible [24].

4. INTRUSION DETECTION MODELS AND FRAMEWORKS

4.1 Intrusion Detection Based On Binary Classification and K-Nearest Neighbors (K-NN)

Detection of intrusion was a significant countermeasure to safeguard computer infrastructure from malicious attacks. This framework recommends a two-step hybrid method based totally on binary class and k-NN technique as proven in Figure 6 to beautify detection efficiency and reduce bias in the direction of frequent attacks. Step 1 uses a couple of binary classifiers and one aggregation module to identify the best community connection instructions effectively and utilizes diverse binary classifiers. After step 1, the connections to unknown classes are sent to step 2 so that you can in addition examine their lessons by means of the K-NN algorithm. Which is primarily based on Step 1 outcomes and gives a superb addition to Step 1. The suggested technique achieves accurate results on the dataset NSL-KDD through blending the 2 steps [25] [26] [27].

Dataset: Just a few public datasets are available in the intrusion detection area to test the efficiency of IDSs models. The NSL-KDD dataset is a successful benchmark, improving the popular KDDCup99 dataset by solving some inherent problems that exist within it.

Evaluation Criteria: The efficiency of binary classification and k-nearest neighborhood (K-NN) intrusion detection models is evaluated by five commonly used measurements: precision, precision, detection rate (DR), F1 value, and false alarm rate (FAR).

Findings: The model proposed reduces biases against repeated types of attack. Furthermore, the technique used in the aggregation module increases the detection efficiency of the process being proposed. On the other hand, when we construct a distributed cloud IDS, it is not enough to using one classifier, we need a number of classifiers in such a way that classifier is located inside a separate network segment; if one classifier fails, better results will be obtained from other classifier.

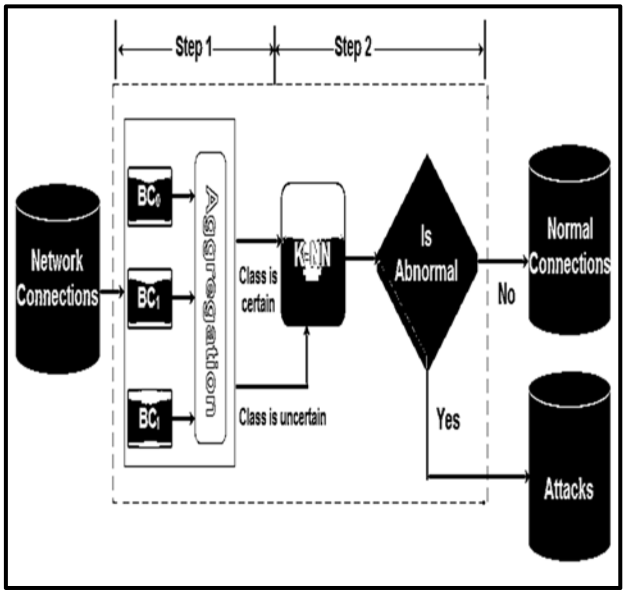


Figure 6: Intrusion Detection Approach Based on Binary Classification and k-NN

4.2 Intrusion Detection Based On Fast Learning Network and PSO

A established variant of the Fast Learning Network (FLN) based on particle swarm optimization (PSO) was proposed in this variant and called PSO-FLN. The model was applied to intrusion detection problem and validated based on the popular KDD99 dataset.

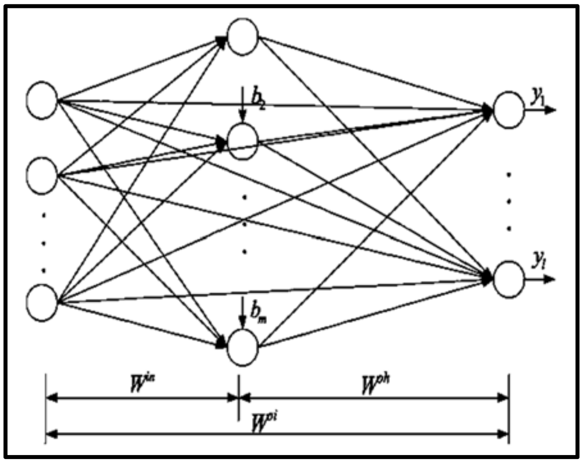


Figure 7: Fast Learning Network (FLN) [35].

4.2.1 Particle swarm optimization

Particle Swarm Optimization (PSO) is an evolutionary parallel computing method created by Mishra and Sengupta [28]. The protocol was created using the social behavior metaphor. The performance of the PSO algorithmic program is powerfully influenced by the enclosed standardization parameters, typically observed because the exploration, that describes the power of totally different regions within the downside house in an endeavor to spot an honest optimum, ideally the worldwide one. Exploitation depicts the capability to concentrate the search getting ready to a promising candidate answer so as to seek out the optimum expeditiously. The choice of algorithm parameters continues largely empirical despite latest studies efforts [29]. PSO algorithm's objective function used to assess its alternatives and is based on the resulting fitness values. Each particle, consisting of the candidate solution, saves its place and evaluates its fitness and speed [30]. In many apps, the PSO algorithm was used to fix many problems [31] [32].

4.2.2 Fast learning network

Fast Learning Network (FLN) is an input, secret and output layer parallel link between an SLFN and a 3-layer FNN [33]. FLN, an Artificial Neural Network, a Double Parallel Forward Neural Network (DPFNN), is shown below using an empirical approach, namely least-square methods as shown in Figure 7. Essentially the FLN is a DPFNN [34]. This defines a multilayer FNN parallel connection and a single FNN connection layer. As mentioned above, together with the internal data itself straight from the input nodes, the re-coded internal data from the concealed nodes is transmitted into the DFNN output nodes.

4.2.3 PSO based optimized FLN

FLN optimization based on PSO is based on the design of a particle representing one FLN weight candidate solution. The selection of both the weight values as well as the amount of neurons required in the hidden layer of achieving greater precision is one particular issue in performing the optimization. This implies a variable length through the solution depending on the amount of concealed neurons in FLN and the maximum amount of neurons in consideration when assigning a length to the particle in order to solve this issue. Tanging was used for the production of the hidden layer neurons for the activation feature.

Dataset: PSO-based intrusion detection should be trained on selected Dataset. To demonstrating the effectiveness of the intrusion detection system model based on PSO, we pick the highest dataset in terms of citation to the KD99 intrusion literature.

Evaluation Criteria: The efficiency of Fast Learning Network and PSO (PSO-FLN) intrusion detection models is evaluated BY Two key common performance assessment metrics for IDSs-accuracy (ACC) and false alarm rate (FAR).

Findings: The experimental study suggested a better performance of the proposed PSO-FLN model based on accuracy and the training running time compared to other intrusion detection models such as RNN and KNN. On the other hand, this PSO-FLN model was not tested simultaneously in detecting the various attacks.

4.3 Intrusion Detection Based On Recurrent Neural Networks (RNN-IDS)

Recurrent neural networks include input units, output units, and hidden units, and the most significant work is done by the hidden unit. The RNN model basically has a one-way flow of data from the entry units to the concealed units, and Figure 8 shows the synthesis of the one-way stream of data from the prior temporal concealment unit to the present timing hiding unit. We can look at hidden units as the entire network's storage that is reminiscent of end-to-end data. We can discover that when we unfold the RNN it embodies the deep learning. For supervised classification learning, an approach to RNNs can be used [35].

Recurrent neural networks have launched a directional loop capable of memorizing and applying the prior data to the present output, which is the vital distinction from traditional neural feed-forward networks (FNN). The previous output is also linked to a sequence's present output, and the nodes between the concealed layers are no longer connected; they have links instead. On the border of the hidden layer functions not only the yield of the original layer but also the output of the last concealed layer [35].

Dataset: The NSL-KDD dataset is commonly used in intrusion detection experiments, which not only effectively addresses the KDD Cup 1999 data set's inherent redundant records issues, but also makes the number of records acceptable in the training set and test set.

Evaluation Criteria: For this model the most important intrusion detection performance metric (Accuracy, AC) is used to calculate the RNN-IDS model performance. For performance measures the detection rate and false positive rate as well as the accuracy are used.

Findings: The RNN-IDS model can effectively improve intrusion detection accuracy as well as detection rate, but this model spends more time for training phase which can be eliminated by using GPU acceleration.

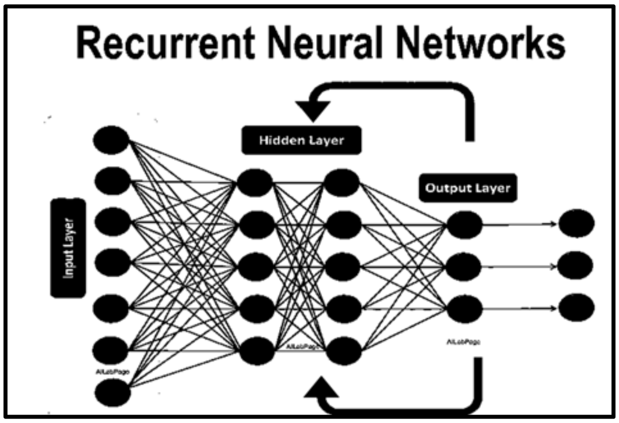


Figure 8: Recurrent Neural Networks (RNNs) [35].

4.4 Intrusion Detection Based On Bayesian Networks

The Bayesian Networks (BN) is a graphic probabilistic model that embodies the relationship between cause and effect. It is by definition a Directed Acyclic Graph (DAG) consisting of a collection of nodes and directed edges where a node represents an interest variable and an edge represents the causality relationship between two nodes. The causality relation is carried by a node-specific Conditional probability table (CPT). It can be applied to system-level dependency graphs due to BN's ability to model cause-and-effect relationships. The rationale is that there is an affinity between BN and system-level dependency charts: in system-level dependency graphs, the dependency relationships between system objects indicate the causalities of BN infection. Figure 9 demonstrates the general design of the scheme: first we conduct system call auditing on each individual host and then collect traces of the system call to a central assessment machine for offline instance graph-based BN building and zero-day route identification [36].

Dataset: To check the IDS model based on the Bayesian Networks, the KDD99 is used as a dataset for intrusion detection evaluation. It consists of a large number of network traffic operation involving both natural and malicious connections.

Evaluation Criteria: For the Bayesian Networks based intrusion detection model, the most important performance metrics are True negative (TN), True positive (TP), False positive (FP) and False negative (FN).

Findings: The planned model of the Bayesian Networks is economical in discovering the attacks within the test dataset and is exceptionally correct in detecting all major attacks reported in the agency dataset. Additionally, realistic implementation of the planned IDS program supported the Bayesian Networks may be wont to train and sight attacks in live network traffic.

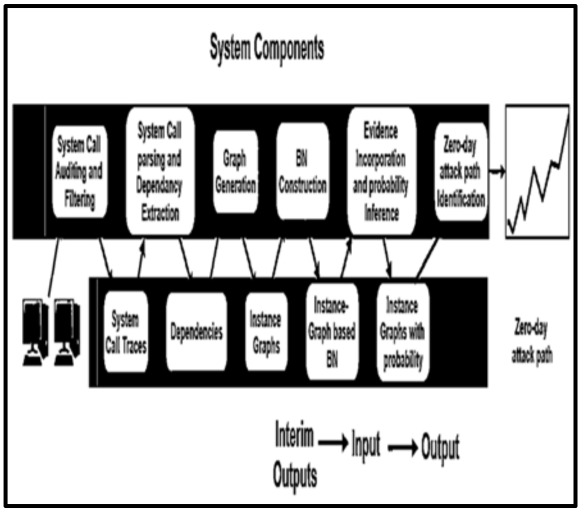

System Components
System Call Auditing and Filtering
System Call parsing and Dependancy Extraction
Graph Generation
BN Construction
Evidence Incorporation and probability Inference
Zero-day attack path Identification
System Call Traces
Dependencies
Instance Graphs
Instance-Graph based BN
Instance Graphs with probability
Zero-day attack path
Interim Outputs
Input
Output

Figure 9: Bayesian Networks Based IDS [36].

4.5 Intrusion Detection Based On Data Mining and Machine Learning

Due to the fast growth of the methods used to assault these networks, concerns about network security have risen in latest years. Detecting congestion from intrusion efforts is becoming harder, according to his growth, as the methods used in these assaults try to use network packets comparable in features to those coming from ordinary flows, making traditional network security methods very weak against such assaults. Thus, more complicated methods are being created to safeguard these networks from complicated assaults, such as using computer teaching to differentiate ordinary data streams from attack packets [37] [38].

Machine learning is the research area which seeks to give machines the capacity to acquire Knowledge from the outside world without any natural contact. Knowledge obtained through a certain method of machine learning may differ from one collection of outputs, from the outside world to another. In addition, information obtained from a given collection of variables may also differ from one method of computer modeling to another, depending on the distinct methods used to obtain such understanding. One of the primary areas of machine learning is data mining, where information sets are entries from the outside world, gathered from the domain needed to extract knowledge [39].

Data mining techniques of a day now play a crucial role in detecting intrusion systems. As IDS is now becoming an important part of our protection infrastructure, data mining techniques can be employed to obtain insightful knowledge of mechanisms for intrusion prevention. These can assist in identifying new vulnerabilities as well as intrusions, identify past unknown trends of attacker behaviors and provide intrusion detection decision support. Various data mining methods, such as classification and clustering, prove useful and often commonly used by analyzing network data, to obtain knowledge about intrusions [40].

Dataset: CIDDS-001(Coburg Intrusion Detection Dataset) is a modified flow-based label dataset created by M. Ringet.al. The dataset contains practical normal and attack traffic that enables critical network intrusion detection systems to be benchmarked in a cloud environment.

Evaluation Criteria: The proposed IDS based data mining and machine learning are evaluated using the following performance metrics (Accuracy, False Positive Rate (FPR) and Running time).

Findings: With an average FPR of 0.21 percent, the overall accuracy of the proposed IDS reached 97 percent for all attack groups. The proposed IDS also achieved a reasonable running time of 6.23s for detecting the attacks in the data set being evaluated.

4.6 Intrusion Detection Based On Unsupervised Networks

The current Network Intrusion Detection Systems (NIDSs) rely on either dedicated signatures of previously seen attacks or marked traffic data sets to discover the network attacks for costly and complex user profiling. While these two strategies are basically the opposite, they share one common downside: they require an inner agent's knowledge, either as signatures or as profiles of ordinary activities. In this section we present the UNIDS model, an unattended detection scheme for intrusion on the network that can detect unidentified network assaults without using any signatures, marked traffic, or training.

With regard to normal-operation traffic we demonstrate that this hypothesis can always be checked through the use of traffic aggregation, their unsupervised detection consists of identifying outliers, i.e. in-stances that differ considerably from the majority [41]. UNIDS depends on solid Sub-Space Clustering (SSC) methods, Density-based Clustering and Evidence Accumulation (EA) methods for blindly extracting traffic-composing instances [42] [43].

UNIDS operates in three successive steps, analyzing packets captured in fixed-length adjacent time slots. Figure 10 shows this system's modular, high-level description. The first step is to detect an anomalous time slot that will perform the clustering assessment. Captured packets are first aggregated into traffic flows with multi-resolution. Different time series are then constructed on top of these flows, and any generic algorithm of change detection based on evaluation of the time series is lastly used to flag an anomalous change. The second step requires all the flows flagged as anomalous in the time slot as input. Using a robust multi-clustering algorithm, based on a mixture of Sub-Space Clustering (SSC) [44]. Density-based Clustering and Evidence Accumulation Clustering (EAC) methods, outward flows are recognized at this phase.

Dataset: The performance of the Unsupervised Network Intrusion Detection System (UNIDS) is evaluated to detect network attacks in the well-known and commonly used dataset of KDD99 network attacks.

Evaluation Criteria: The following output metrics (detection accuracy, False Positive Rate (FPR) and True Positive Rate (TPR) are used to determine UNIDS.

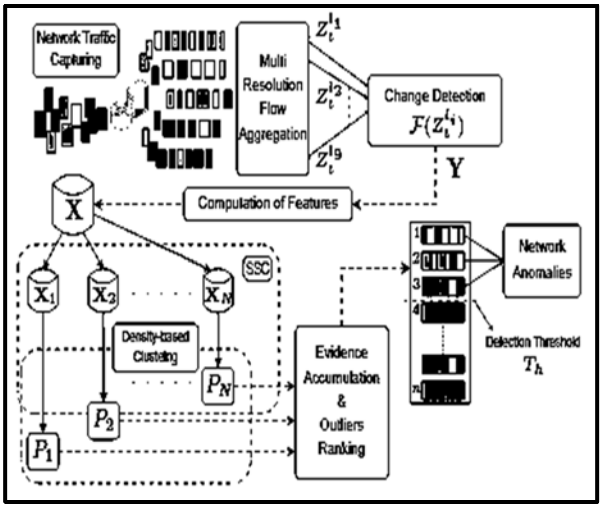


Figure 10: Unsupervised Network Intrusion Detection System (UNIDS)

Findings: The experimental findings are tested on UNIDS 'effectiveness in detecting real single source-destination and distributed network attacks from various networks in real traffic traces, but with high False Positive Rate (FPR).

5. INTRUSION DETECTION METHODS AND ALGORITHMS

5.1 Current Methods for Intrusion Detection

5.1.1 Signature - based intrusion detection system

Signature based detection is the method of matching patterns to the observed events in order to identify potential intrusions. It is also called Misuse-based Identification or Knowledge-based. The signature-based detection system can achieve a high degree of accuracy and less false positives when detecting intrusions [45].

The authors deployed the network-based IDS in separate cloud positions in the study done by Mazzariello, Canonico, and Bifulco. In determining the efficacy of the IDS two results were shown by considering two scenarios. First, they inferred enhanced controller load, and the IDS recognized the likelihood of the attack. Second, closing the virtual machine to IDS deployment resulted in an increase in Processor load [46].

Strengths: An effective approach for detecting known attacks while identifying intruders with a high degree of accuracy and a low false positive rate. Also one of the driving factors for using signature-based detection is its ease of updating and preserving preconfigured guidelines. Such signatures are made up of several elements which identify the traffic.

Limitations: It is an effective method for detecting known attacks but fails to detect zero-day attacks or known attack variants, also it has a high missed alarm rate and it needs maintaining a large database of signatures.

Cloud Environment: The signature-based intrusion observation technique are often accustomed detect external intrusions either at the face of the cloud, or at the rear finish of the cloud to observe the intrusions. This can't be accustomed track Cloud-based zero-day attacks like ancient network.

5.1.2 Anomaly - based intrusion detection system.

Detection of abnormalities includes detecting incidents that tend to be anomalous as per normal system behavior.

The anomaly-based approach involves gathering information on legitimate users 'actions over a amount of your time so applying applied mathematics tests to the discovered behavior to see whether or not or not that behavior is legitimate [47]. It detects known attacks in this technique, as well as the zero-day attacks. Anomaly detection methods are used for detecting various rates of unknown attacks. A wide vary of ways together with data processing and applied mathematics analysis are studied as alternative ways to deal with anomaly detection problems.

Strengths: It has the advantage of characteristic threats that haven't been known before. The key component for victimization this technique with success is to form rules in such some way that the warning rate for unknown and well-known attacks could also be reduced.

Limitations: Despite of the ability of this technique to detecting the zero-day attacks, but one downside is the high false alarm rate and the poor precision.

Cloud Environment: There are vast numbers of events occurring in Cloud (network or device level), making it difficult to track or regulate them using anomaly detection.

5.2 Software Computing Based Intrusion Detection Techniques

The ability to handle ambiguous and partially accurate data through machine programming techniques makes them appealing to use when detecting intrusions. To improve the accuracy and efficaciousness of signature or anomaly-based IDS detection, bound package computing techniques corresponding to the factitious Neural Network (ANN), formal logic, and Support Vector Machine (SVM), etc. are used.

Using machine computing methods is preferable rather than using the conventional Cloud system IDS. That technique, however, has some strengths and disadvantages which affect IDS performance.

5.2.1 Artificial neural network intrusion detection technique

Techniques victimization ANN to explain intrusions are meant to generalize and establish data as either intrusive or incomplete. ANN IDS can be used with Strategies for Multi-Layer Perceptron (MLP), Back Propagation (BP) or Multi-Layer Feed Forward (MLFF). A Gradiega Ibarra, Ledesma, and Garcia approach combined the utilization of organization map (SOM) with MLP to assess intrusion rates and located that Kyrgyzstani monetary unit provides high detection rates compared to ANN. in an exceedingly three-layer neural network, Cannady used a signature-based identification theme to classify intrusions. To gauge intrusions, he used a 9 network feature vector consisting of the supply port, raw data, destination port, information length, supply science address, ICMP string, ICMP sort, and destination science address [48].

Strengths: ANN-based IDS provides a necessary answer for unstructured network knowledge.

Limitations: The performance of this intrusion detection system depends on the amount of hidden layers, also because the ANN coaching cycle. Nevertheless, it will need additional coaching samples and time to effectively learn ANN.

Cloud Environment: Using ANN-based IDS alone cannot be a good resolution for police investigation cloud intrusion, as a result of it wants a framework for speedy intrusion detection.

5.2.2 Genetic algorithm intrusion detection systems

Using genetic algorithms within the growth of IDS helps to mix totally different network characteristics to determine the simplest potential parameters for rising accuracy and optimizing potency. In packet analysis, Gong, Zulkernine, and Abolmaesumi enforced seven network options, particularly frequency, protocol, source IP, destination information processing, supply port, port of destination, and name of attack. The authors were able to identify and assess high-precision intrusions of the network by using confidence-building feature frameworks.

The author suggested a solution combining genetic and fuzzy algorithms with signatures and anomalies to classify assaults. Fuzzy logic allows quantitative parameters to be considered, while genetic algorithms determine the best matching parameters that the fuzzy logic will provide.

Strengths: Genetic Algorithm (GA) works on a population of possible solutions using the survival principle of the most fit to produce better and better approximations to the solution of the problem GA is trying to solve.

Limitations: The genetic algorithms begin processing by selecting a random chromosome population in the first place. The chromosome consists of a finite number of chromosomes, which is predefined in each application.

Cloud Environment: The selection of suitable intrusion detection parameters (network features) will improve the performance of the IDS underlying them. Genetic algorithmic rule (GA) based mostly IDS is also used for this in Cloud.

5.2.3 Fuzzy logic intrusion detection system

Fuzzy logic offers a high degree of flexibility to outline intrusion issues. It helps handle defective intrusions. Tillapart, Thumthawatworn, and Santiprabhob urged a Fuzzy IDS to forestall intrusions to the network equivalent to Ping of Death, SYN, UDP floods, email bombs, port scanning, and FTP countersign guess. Chavan, Shah, Dave, and Mukherjee have applied each mathematical logic and ANN to develop the Evolving Fuzzy Neural Network (EFUNN) that enforced unattended and supervised learning. Their analysis found that using EFUNN at high precision levels with fewer input results than using ANN alone [48].

Strengths: This gives the ambiguous question of intrusion detection some flexibility and may be accustomed quickly detect unknown attacks in Cloud exploitation formal logic with ANN.

Limitations: Cannot be accustomed discover network intrusions in real time, as a result of the training amount is extremely long.

Cloud Environment: In Cloud, rules of association may be wont to produce new signatures. Variations of noted attacks may be detected in real time, mistreatment these fresh created signatures.

5.2.4 Support Vector Machine (SVM) intrusion detection systems

SVM detective work intrusions with small samples of data whose size doesn't have an effect on the result's accuracy. Examination SVM with ANN, Chen, Su, and Shen it had been determined that the degree of false-positive with SVM was a lot of correct since the parameters set with SVM were borderline. A limitation on SVM is that it will solely take a look at binary knowledge. Li and Liu have instructed an alternate good network intrusion and turning away theme employing a configurable firewall and SNORT tool to attenuate alarm rates and increase the accuracy of the intrusion detection system [49].

Strengths: Support Vector Machines (SVM), due to its powerful generalization character and the ability to resolve the curse of dimensionality, has become one of the popular machine learning algorithms used to detect intrusion.

Limitations: Support Vector Machines (SVM) handles all data functions equally. Many features are redundant or less critical in the real intrusion detection datasets.

Cloud Environment: In Cloud, if restricted sample information are provided for intrusion detection, the utilization of SVM is Associate in nursing economical answer than ANN; as a results of data measurements don't have a sway on the accuracy of SVM-based IDS.

5.2.5 Hybrid intrusion detection systems

These methods, given the great ability of anomaly detection systems to detect unknown or zero-day vulnerabilities, suffer from a major weakness, namely their high false alarm rate. That is induced mainly for two reasons. The first is the lack of a training data collection covering all the appropriate areas and the second is that irregular activity is not necessarily an intrusion signal.

Some researchers have suggested the concept of hybrid detection to address these problems and retain the advantages of detecting misuse. Hybrid IDS incorporates the benefits of the above methods, two or more. Krishna and Quadir

Launched a fresh DDoS detection system, which enforced design supported the Hidden Markoff model and therefore the twin communications protocol mechanism. Five packets double apply the 3-way handshaking protocol, and use a SYN to preserve a log [50]. The aim of the double communications protocol technique is to make sure that associate identification match exists before an association is terminated.

The author [51] notes that the Andrei Markov model assists in detective work any suspicious behavior once applied to wireless device networks. No link is left 0.5 open as a result of the shopper cannot reciprocate an identical pattern, associated an assault are often half-tracked back to its creator (Modi and Abdul, 2014). Vissers recommended the Cloud Trace Back (CTB) answer as a process for internet services by detective work edge routers. In reverse, SOA is enforced to trace the precise origin of a distributed denial of service attack. A Cloud Trace back Mark (CTM) is inserted into the header of a web file. All programs are then taken through the CTB prohibiting any direct assault. To discover it, the user patron needs the reconstruction of the request to attack [52] [53].

Distinct variation using both the covariance strategy and the entropy-based scheme is proposed which provide host-and network-level in-depth detection [54]. A table Displaying the traditional techniques of intrusion detection discussed and outlined in the works of (Modi et al., 2013; Kacha et al., 2013; Dewal et al., 2016). Is shown in Table 2

.

Table 2: Summary of traditional IDS techniques

No	IDS Technique	Pros	Cons
1	Signature-based IDS	1) High accuracy with relevance police work best-known attacks. 2) Offers low information processing costs. 3) Straightforward to watch and forestall attacks owing to thorough log files.	1) Are unable to discover intelligent intrusions. 2) The detection of recent attacks within the information. 3) Massive traffic limits examination of any packet that permits unattended packets to meet up with.
2	Anomaly-based IDS	1) The warning rate is higher for non-identified assaults. 2) New threats are simple to observe while not upgrading the information. 3) The system teaches for itself. It bit by bit learns concerning the network, and builds a profile. 4) The upper the amount of accuracy used.	1) A network remains in unmanaged condition during profile development and is therefore vulnerable to attack. 2) It is untraceable when malicious operations assume ordinary traffic characteristics. 3) Collected conduct and characteristics determine the detection precision. 4) Low accuracy when unidentified attacks are detected.
3	ANN based IDS	1) Sensible detection of unstructured network packets. 2) Classification potency obtained by the addition of specific hidden layers.	1) The training stage requires a lot of time. 2) Has less versatility. 3) Further samples of knowledge are required to provide effective training.
4	Genetic algorithm IDS	1) Provides the most effective detectable characteristics. 2) Has improved potency.	1) Terribly rough. 2) Its usage contrasts particularly with a general pattern.

5	Fuzzy logic IDS	1) Improved skillfulness in addressing unpredictable issues.	1) Gives low precision values compared to ANN.
6	SVM based IDS	1) Justifiably classifies intrusions, even with bottom sample details. 2) Ability to manage an outsized array of tasks.	1) Classifies solely completely different options, thus, the options should be pre-processed before they're enforced.
7	Hybrid Techniques	1) Effective as it combines several methods to classify rules correctly. 2) Detect newly brand attacks	1) Its cost of computing is high. 2) Negative effect on precision as well as efficiency.

6. RESULTS AND FINDINGS

The paper discusses progressive intrusion detection ways and offers a comprehensive description of the assorted approaches and algorithms for detection. The definition of techniques for intrusion detection is described evolutionarily in order to provide an in-depth view of the different approaches.

Research results reflect conclusions in depth after going through various IDS methods, and point out the pros and cons of each methodology and method. The paper concluded that the current IDS techniques and algorithms are not capable of detecting unknown attacks with minimal false warning rates on the cloud platform.

The main contribution for this review paper is to recommend building a multi-phase anomaly intrusion detection technique to detect zero-day, fast-spreading and complex network attacks on the cloud platform with the least amount of false alarm rates.

7. CONCLUSION AND FUTURE WORK

A review of existing intrusion detection techniques, algorithms and models in cloud computing has been provided in this paper. The analysis affords a cloud-based totally IDS taxonomy primarily based at the detection type, detection technique, and approach for regular profile development, IDS architecture, and detection time. Based on the review open issues were found in cloud-based IDS, and suggestions were made for future study. The research indicates that the cloud-based IDS has done a variety of research work.

The cloud IDS research work covers different detection methods, scope, architectural structure, and time detection. The open research problems in cloud-based IDS, however, are the existence of high false alarm due to highly dynamic environment with dynamically added and removed nodes, so IDS is needed to possess the capability that will cope with the dynamic cloud climate. The distributed cloud nature also makes it susceptible to distributed attack; however, it does not validate most of the existing approach to detecting distributed attacks in IDS using collaborative IDSs.

Cloud computing systems are easy targets for intruders and present new risks and challenges to an enterprise due to its application and operating models, underlying technology and distributed design. Specifically, some kind of sharing is inherent in cloud computing, and cannot be prevented. It in effect blurts the common distinction between private resources and public resources. In future work we tend to advocate a fresh accommodative framework of intrusion detection system, Hypervisor-based Cloud Intrusion Detection System (HVCIDS), to beat variety of the troubles of typical cloud-primarily based mostly intrusion detection systems, like the single point of failure, high false alarm and the detection time issue especially in the era of big data.

REFERENCES

[1] Elmasry, Hesham, Khedr, Ayman E., Nasr, Mona, (2019) An adaptive technique for cost reduction in cloud data center environment, international journal of Grid and Utility Computing Vol. 10 (5), 448-464.

[2] Khedr, Ayman E., Kholeif, Sherief, Hessen, Shrouk, (2015) Adoption of cloud computing framework in higher education to enhance educational process, International Journal of Innovative Research in Computer Science & Technology (IJIRCST), Vol. 3 (2).

[3] Khedr, Ayman E., Idrees, Amira M., (2017), Enhanced e-Learning System for e-Courses Based on Cloud Computing., Journal of Computers, Vol 12 (1), 10-19.

[4] Hegazy, Abdelfatah, Khedr, Ayman E., Geddawy, Yasser, (2015), An adaptive framework for applying cloud computing in virtual learning environment at education a case study of "AASTMT", International Conference on Communication, Management and Information Technology (ICCMIT 2015), Procedia Computer Science Vol. 65, 450-458.

[5] Khedr, Ayman E., Nasr, Mona, Elmasry, Hesham, (2015) new balancing technique for green cloud computing and environmental Sustainability, International Journal of Advanced Research 3 (9), 201-215.

[6] Kruegel, C., Valeur, F., and Vigna, G. (2004). Intrusion Detection and Correlation: Challenges and Solutions. Springer-Verlag Telos.

[7] Ashok, K., and Venugopalan, R. (2017). Intrusion detection systems: a review. International Journal of Advanced Research in Computer Science, Vol. 8(8), 1-16.

[8] Anderson, J. (1972). Computer security technology planning study. Technical Report, ESDTR-73-51, United States Air Force, Electronic Systems Division.

[9] Debar, H., Dacier, M., and Wespi, A. (1999). Towards a Taxonomy of Intrusion Detection Systems. Computer Networks, Vol. 31(8), 805-822.

[10] Vokorokos, L., and Baláž, A. (2010). Host-based intrusion detection system, in Intelligent Engineering Systems (INES). IEEE 14th International Conference on Intelligent Engineering Systems.

[11] Chauhan, P., and Chandra, N. (2013). A Review on Hybrid Intrusion Detection System using Artificial Immune System Approaches. International Journal of Computer Applications, Vol. 68(20), 1-6.

[12] Marteau, F. (2018). Sequence covering for efficient host-based intrusion detection. IEEE Transactions on Information Forensics and Security, Vol. 14(4), 994 – 1006.

[13] Modi, C., Patel, D., Patel, H., Borisaniya, B., Patel, A., and Rajarajan, M. (2013). A survey of intrusion detection techniques in Cloud. Journal of Network and Computer Applications, Vol. 36(1), 42-57.

[14] Yu, K., Wu, M., Wong, W. (2008). Protocol-based classification for intrusion detection. In WSEAS Transactions on Computer Research Journal, Vol. 3(3), 135-14.

[15] Gupta, R., Singh, S., Verma, S., and Singhal, S. (2017). Intrusion detection system using snort. International Research Journal of Engineering and Technology (IRJET), Vol. 4(4), 1-5.

[16] Vigna, G., and Kemmerer, R. (1998). NetSTAT: A network-based intrusion detection approach. In Proceedings. 14th Annual Computer Security Applications IEEE Conference.

[17] Patel, A., Taghavi, M., Bakhtiyari, k., and Júnior, J. (2013). An intrusion detection and prevention system in cloud computing: A systematic review. Journal of network and computer applications, Vol. 36(1), 25-41.

[18] Kshirsagar, D., Sawant, S., Wadje, R., and Gayal, P. (2017). Distributed Intrusion Detection System for TCP Flood Attack. In Proceeding of International Conference on Intelligent Communication, Control and Devices, Vol. 479, 951-958.

[19] Göcs, L., and Johanyák, C. (2015). Survey on intrusion detection systems. In 7th International Scientific and Expert Conference TEAM Technique, Education, Agriculture & Management, Belgrade, 1-11.

[20] Khan, A., and Herrmann, P. (2017). A trust based distributed intrusion detection mechanism for internet of things. IEEE 31st International Conference on Advanced Information Networking and Applications (AINA), 1-8.

[21] Mishra, P., Pilli, E., Varadharajan, V., and Tupakula, U. (2017). Intrusion detection techniques in cloud environment: A survey. Journal of Network and Computer Applications, Vol. 77(C), 18-47.

[22] Hebbal, Y., Laniepce, S., and Menaud, J. (2015). Virtual machine introspection: Techniques and application. In Availability, Reliability and Security (ARES), 2015 10th International Conference on IEEE, 1-10.

[23] Phillip, A., Porras and Alfonso V. (1998). Live traffic analysis of tcp/ip gateways. In Proceedings of the 1998 ISOC Symposium on Network and Distributed System Security (NDSS'98), San Diego, CA, Internet Society.

[24] De la Hoz, E., Ortega, J., and Martínez-Álvarez, A. (2014). Feature selection by

multi-objective optimization: Application to network anomaly detection by hierarchical self-organizing maps. Knowledge-Based Systems., Vol. 71, 322-338.

[25] Ji, S., Jeong, B., Choi, S., and Jeong, D. (2016). A multi-level intrusion detection method for abnormal network behaviors. Journal of Network and Computer Applications., Vol. 62, 9-17.

[26] Bhuyan, M., Bhattacharyya, D., and Kalita, J. (2014). Network anomaly detection: Methods, systems and tools. IEEE Communications Surveys & Tutorials, Vol. 16(1), 303-336.

[27] Mishra, V., and Sengupta, A. (2014). MO-PSE: Adaptive multi-objective particle swarm optimization based design space exploration in architectural synthesis for application specific processor design. Advances-in-engineering-software., Vol. 67, 111-124.

[28] Trelea, I. (2003). The particle swarm optimization algorithm: Convergence analysis and parameter selection. Information Processing Letters., Vol. 85(6), 317-325.

[29] Blondin, J. (2009). Particle Swarm Optimization: A Tutorial. [Online]. Available: http://cs.armstrong.edu/saad/csci8100/pso_tutorial.pdf.

[30] Sengupta, A., Bhadauria, S., and Mohanty, S. (2017). TL-HLS: Methodology for low cost hardware Trojan security aware scheduling with optimal loop unrolling factor during high level synthesis. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems., Vol. 36(4), 660-673.

[31] Sengupta, A., and Mishra, V. (2014). Expert systems with applications automated exploration of data path and unrolling factor during power_ Performance tradeoff in architectural synthesis using multi-dimensional PSO algorithm. Expert System Application, Vol. 41(10), 4691-4703.

[32] Sahu, S., Sarangi, K., and Jena, S. (2014). A detail analysis on intrusion detection datasets. In Proc. IEEE International Advance Computing Conference (IACC), 1348-1353.

[33] Olusola, A., Oladele, A., and Abosede, D. (2016). Analysis of KDD'99 intrusion detection dataset for selection of relevance features. In Proc. World Congress on Engineering and Computer Science, Vol. I, 16-23.

[34] Martens, J., and Sutskever, I. (2011). Learning recurrent neural networks with hessian-free optimization. In Proc. 28th International Conference on International Conference on Machine Learning, Bellevue, WA, USA, 1033-1040.

[35] Spirtes, P., Richardson, T., and Meek, C. (1995). Learning U2R Bayesian networks with discrete variables from data. In Proceedings of (15) 6.66% the First International Conference on Knowledge Discovery and Data Mining, 294-299.

[36] Acemoglu, D., Malekian, A., and Ozdaglar, A. (2016). Network security and contagion. Journal of Economic Theory, Vol. 166, 536-585.

[37] Yu, D. Jin, Y. Zhang, Y. and Zheng, X. (2018), “A survey on security issues in services communication of Micro services-enabled fog applications”, Concurrency and Computation: Practice and Experience, cpe4436.

[38] Storey, V., and Song, I. (2017). Big data technologies and Management: What conceptual modeling can do. Data & Knowledge Engineering, Vol. 108, 50-67.

[39] Witten, I., Frank, E., Hall, M., and Pal, C. (2016). Data Mining: Practical machine learning tools and techniques. Morgan Kaufmann.

[40] Androulidakis, G., Chatzigiannakis, V., and Papavassiliou, S. (2009). Network Anomaly Detection and Classification via Opportunistic Sampling. In IEEE Network, Vol. 23(1), 6-12.

[41] Parsons, L., Haque, E., and Liu, H. (2004). Subspace Clustering for High Dimensional Data: A Review. In ACM SIGKDD Explorations Newsletter, Vol. 6(1), 90-105.

[42] Ester, M., Kriegel, P., Sander, J., and Xu, X. (1996). A Density-based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In Proc. ACM SIGKDD.

[43] Fred, A., and Jain, K. (2005). Combining Multiple Clustering Using Evidence Accumulation. In IEEE Trans. Pattern Anal. And Machine Int., Vol. 27(6), 835-850.

[44] Gander, M., Felderer, M., Katt, B., Tolbaru, A., Breu, R., and Moschitti, A. (2012). Anomaly detection in the cloud: Detecting security incidents via machine learning. In International Workshop on Eternal Systems, 103- 116.

[45] Kene, S., and Deepti, P. (2015). A Review on Intrusion Detection Techniques for Cloud Computing and Security Challenges. 2nd International Conference on Electronics and Communication Systems (ICECS), Coimbatore, Vol. 2, 227-231.

[46] Nascimento, G., and Correia, M. (2011). Anomaly-based intrusion detection in software as a service. In2011 IEEE/IFIP 41st International Conference on Dependable Systems and Networks Workshops, 19-24.

[47] Modi, K., and Abdul, Q. (2014). Detection and Prevention of DDoS Attacks on the Cloud using Double-TCP Mechanism and HMM-Based Architecture. International Journal of Cloud Computing and Services Science, Vol. 3(2), 113-120.

[48] Kacha, C., Shevade, K., and Raghuwanshi, K. (2013). Improved Snort Intrusion Detection System using Modified Pattern Matching Technique. International Journal of Emerging Technology and Advanced Engineering, Vol. 3(7), 81-88.

[49] Parwani, D., Dutta, A., Shukla, P. (2015). Various Techniques of DDoS Attacks Detection and Prevention at Cloud: A Survey. Oriental Journal of Computer Science & Technology, Vol. 8(2), 110-120.

[50] Dewal, P., Narula, G., and Jain, V. (2016). A Survey of Intrusion Detection Systems and Secure Routing Protocols in Wireless Sensor Networks. International Journal for Research in Emerging Science and Technology, Vol. 3(1), 1-7.

[51] Chawla, I., Luthra, P., and Kaur, D. (2015). DDoS Attacks in Cloud and Mitigation Techniques. International Journal of Innovative Science, Engineering & Technology, Vol. 2(7), 596-600.

[52] Reddy, V., Rao, K., and Lakshmi, P. (2012). Efficient Detection of DDoS Attacks by Entropy Variation. IOSR Journal of Computer Engineering, Vol. 7(1), 13-18.

[53] Girma, A., Moses, G., Li, J., Lui, C., and Abayomi, K. (2015). Analysis of DDoS Attacks and an Introduction of a Hybrid Statistical Model to Detect DDoS Attacks on Cloud Computing Environment. 12th International Conference on Information Technology-New Generations, Las Vegas, 212-217.