ISSN: 2633-4828 Vol. 6 No.1, January, 2024

International Journal of Applied Engineering & Technology

COMPREHENSIVE SURVEY OF VARIOUS MACHINE LEARNING APPROACHES FOR INTRUSION DETECTION SYSTEM AND MALICIOUS URL IN CLOUD COMPUTING

Swetha T and Dr. Seshaiah Merikapudi
Departement of computer Science & Engineering, SJCIT Chickballapur,India
Swetha102reddy@gmail.com and merikapudi@gmail.com

ABSTRACT

Since cloud computing is altering the way that data is processed and stored, robust security measures are necessary to guard against new cyber threats. In this research, intrusion detection systems (IDS) in cloud computing environments are thoroughly examined. We identify several kinds of intrusions that target cloud resources' CIA (Confidentiality, Integrity, and Availability). The article discusses the advantages and disadvantages of many IDS kinds, such as network-based, host-based, VMM/hypervisor-based, and distributed IDS, by classifying and analyzing them. In addition, extant cloud-based intrusion detection systems are examined through the lens of intrusion detection methodologies, providing insights into hybrid, anomaly-based, and signature-based systems. To achieve complete security, the research emphasizes the need of an integrated strategy that combines several detection systems. Even if there have been improvements, problems like false alerts and effects on virtual machine performance still arise. A standardized framework for cloud security and further research cooperation are emphasized in the conclusion.

Keywords: Cloud Computing, Intrusion Detection System, Security, Confidentiality, Integrity, Availability, Soft Computing Techniques, Cyber Threats

INTRODUCTION

A malicious URL contains spyware or malware that is detrimental to the computer. This virus has the ability to unintentionally infiltrate user systems and get some legally-protected data from them [1]. Computer science's machine learning discipline applies statistical methods to enable machines to "learn" new skills based on existing ones without directly coding them. Different types of machine learning fall into one of three broad categories: supervised, semi-supervised, and unsupervised.

A Phishing is the effort by an individual or organization to get credit card numbers and other private information from unwary victims in order to commit identity theft, obtains financial gain, or engages in other fraudulent acts. In the present situation, the user login into his bank or secure mail account and provides details such his username, password, credit card number, etc. in order to access his private information online (via a money transfer or payment gateway) [2]. However, phishing tactics are often used by attackers to get this information (for example, a phishing website might gather the user's login credentials and send him to the genuine site). When a user logs in, there is no information that can't be immediately gotten from them.

A phishing website is defined as "any website that, without authorization, pretends to be someone else in order to deceive viewers into doing something that they would only trust a real representative of that other party to do [3]. Some of them contain graphics related to financial companies and ask for the viewer's personal credentials. Others boast about being able to use a third party to do things once they get their login credentials. So, a URL that directs a person to a phishing webpage is considered a phishing URL. Accordingly, our analysis is unaffected by the attack vector that disseminates phishing URLs [4].

Along with other forms of cybercrime, such as virus attacks and hacker attacks, phishing is a more recent phenomenon. Phishing sites, as seen in Figure 1, are part of a larger social engineering campaign that targets individuals by tricking them into handing over sensitive information such bank account, credit card, or certificate numbers. The goal of this assault is to illegally exploit this information. Businesses suffer serious losses in money, customer relationships, marketing campaigns, and credibility when phishing occurs. Due to fraud-related losses and employee time, phishing attempts may cost businesses a large amount of money every time they

happen [5]. Worse still, significant quantities of money may still be spent as a result of a brand's reputation and consumer trust declining.

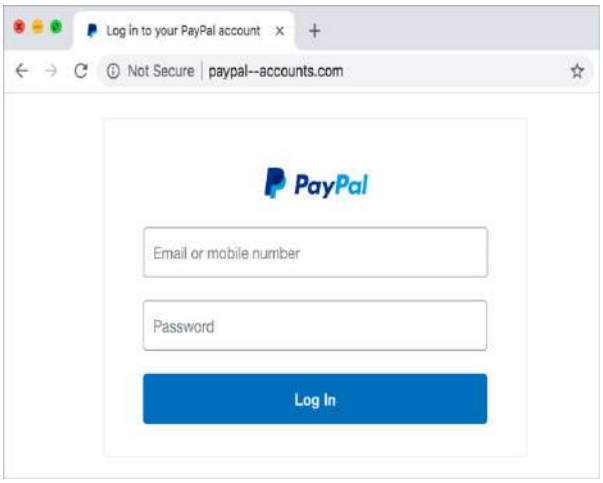

Figure 1: Image captured from a malicious website [6]

There are several ways to describe "phishing website," thus one should use extreme caution while defining the phrase since it is always changing. This description is one of these definitions. Phishing attacks usually include identity theft, fake websites, and phoned emails. Many financial institutions and their customers are the targets of phishing attempts [7]. However, there are other definitions of phishing websites that come from various angles. They discuss a few of these definitions below to help you better comprehend its characteristics and assault strategies.

Phishing websites are fraudulent websites designed to seem like the official websites by unscrupulous individuals. The majority of these websites have striking visual resemblance to deceive their victims. Certain types of these webpages have an identical replica of the actual ones. Phishing website victims risk disclosing to the operators of the website their credit card number, bank account information, password, and other private data [8]. It covers strategies like installing key loggers, taking screenshots, deceiving consumers via spam and email communications, and man-in-the-middle assaults. These widely used technologies have a number of shortcomings:

- An approach based on blacklists that have a low risk of false alarms;however, it fails to detect websites that aren't part of the blacklist database [9]. Because phishing websites are short-lived and the blacklist is not immediately put in place, its accuracy is low.
- Attackers may easily use technical tactics to circumvent heuristic feature identification, and heuristic-based anti-phishing solutions are prone to false and ineffective warnings.
- The approach based on similarity evaluation takes a lot of time. It is not appropriate to use the approach to identify due to the lengthy computation time required to display two pages on the client terminal, phishing websites [10]. Additionally, this approach's poor several factors affect accuracy, including text, images, and, and the method used to determine similarity. But this method—specifically, the picture similarity identification method—is still not flawless.

2. LITERATURE REVIEW

The technique proposed by Dhamija and Tygar (2005) makes advantage of a user's browser's "dynamic security skin" [11]. This method makes it difficult for phishers to impersonate a remote service, yet also allows for easy human verification, by using a shared secret photo. The user has to put in some effort, which is the method's biggest downside. This strategy can only be successful if the whole industry supports it, because the whole web

infrastructure (clients and servers) must be changed. Additionally, in cases when the user logs in from a public terminal, this strategy is insecure.

Dhamija et al. (2006) Anti-Phishing Work Group database of 200 phishing attempts and discovered many reasons why individuals fall for these attacks, such as a lack of knowledge about computers and visual tricks used by scammers [12]. They also engaged 22 volunteers in a usability research. In order to determine if 20 distinct websites were genuine or fake, the participants were requested to examine them. The outcome demonstrated that there was little variation in age, sex, and computer usage. They also observed that visual cues such as padlocks, SSL (Secure Sockets Layer), and pop-up notifications about a website's incorrect signature were very ineffective and often ignored. They discovered that 40% of the times, participants were vulnerable to phishing assaults since 23% of them ignored security signs alerting them to the threat. The authors' study leads them to conclude that it is critical to reconsider security system design, especially with regard to usability.

Wu et al. (2006) introduced techniques that, by using sensitive information placement characteristics in HTML code mandate that web page designers adhere to certain guidelines while creating web pages [13]. It is difficult to convince everyone who creates websites to abide by the guidelines, nevertheless.

Liu et al. (2005) compared real and fake websites to find visual similarities that might be used to detect phishing attempts. Block level, layout, and style commonalities were found[14].

A visual approach to phishing detection was first proposed [15], and DOM is responsible for directing the visual similarity of web pages (Wood, 2005). This approach detects and reports phishing websites automatically, doing away with the requirement for heavy human effort. Web pages written in HTML are first divided into easily discernible block sections using their method. The next step is to compare the visible parts of the blocks to determine how similar the two web pages are visually. There are three metrics for this: similarities at the block level, layout, and style. Phishing websites if their visual similarity score is more than a certain threshold.

Using EMD-based visual similarity assessment, Fu et al. (2006) [16] developed a method for detecting phishing web pages. This technique evaluates online pages at the pixel level, in contrast to text-level methods that can only identify phishing websites that are "visually similar" to protected websites, not source code. Internet Explorer 8's toolbar phishing filter blocks user activity on recognized phishing sites. But the most popular and widely utilised ones depend on adding phishing sites to browser blacklists so that users cannot access them. One example is the new anti-phishing feature in Microsoft's Internet Explorer (IE8), which is based on a blacklist. The browser communicates with Microsoft servers to get lists of approved and prohibited sites to prevent phishing. It is also known that certain algorithms are used by Microsoft's approach in order to detect phishing indicators on websites (Sharif, 2005). Clearly, the company has not yet provided the public with any details on their anti-phishing strategies [17].

Based on its structural features, Chandrasekaran et al. (2006) recommended classifying phishing emails. Twenty-five characteristics were studied, including style indications (suspended, account, and security) and structural components (email subject line and body greeting). Out of 200 emails reviewed, 100 were phishing attempts and 100 were legitimate correspondence. Simulated annealing was the method utilised for feature selection. The chosen set of features was ranked according to their relevance using information gain (IG). It follows that a single class support vector machine was used to classify phishing emails in accordance with the selected attributes. Based on the results, it is possible to identify 95% of phishing emails using a simple [18].

Based on their phishing detection effectiveness utilizing previous data, many popular learning algorithms were compared, Fette et al. (2007) integrated Random Forests in their algorithm PILFER. The techniques, according to the authors, may also be used to identify phishing websites. A total of 6950 authentic emails and 860 phishing emails were examined. With a 0.1% false positive rate, the suggested strategy identified 96% of the phishing emails properly. From a phishing dataset gathered in 2002 and 2003, ten hand-selected characteristics were used

for training. The authors themselves have acknowledged that their implementation is not ideal and that further research in this area is necessary [19].

Six machine learning approaches were examined by Abu-Nimeh et al. (2007) in order to categorize phishing emails. They used 43 features (variables) and a phishing corpus consisting of 28,89 emails. The results demonstrated that a bag-of-words alone could provide high prediction accuracy as the spam detection method, and their feature set was only a collection of words. The problem is that textual components alone produce a lot of false positives since phishing emails seem so much like legitimate ones. Over 92% of phishing emails were identified by the classifiers evaluated [20].

The inconsistencies in online sites were studied by Pan and Ding (2006), namely the contradiction between the identity of a website and structural components and HTTP transactions [21].

Herzberg and Gbara (2004) proposed a method to visually verify appropriate certification while using standard certificates. This system would display a site-specific badge in the browser's trusted credentials area, verifying the authenticity of the certificate [22]. One such way to find attack instances that happen often is to look for misread URLs of prominent targets or attacks where one site hosts the text while another provides the images. A comprehensive analysis of the many techniques often used in phishing attacks is offered in Ollmann's 2004 book "The Phishing Guide" [23]. Starting with simple emails that tricked the receiver into responding with the attacker-needed information, this phenomenon has evolved into more complex methods of victim manipulation. As the level of sophistication of the phoney websites used to fool victims into inputting the information the attacker asks increases, victims are led to these sites using false advertisements and email links. For example, the counterfeit site may seem just like the actual firm. In addition, Ollmann provides a plethora of tools for checking the legitimacy of a website. Verifying the authenticity and signature of the certificate issued to the website is just as important as making sure the website is actually secure using SSL (secure Sockets Layer). If you pay more attention to the URL, you can usually find fake websites. Users may more quickly verify the authenticity of the website they are visiting if they are aware of the many ways in which attackers might alter the URL to seem legitimate.

In their White Paper titled "Know your enemy: Phishing," Watson et al. (2005) provide a detailed account of several actual phishing attacks that were recovered from honey nets in the UK and Germany [24]. "Honey nets" are open computer networks that collect data on different real-world risks for forensic analysis. Phishing attacks that leverage vulnerable web servers to host pre-made phishing sites are far more common, they found, unlike self-compiled servers. On occasion, a compromised server may house several aliases used by phishing websites. After being downloaded to the server, these websites often stay up for a few hours or even days.

Garera et al.'s research [25] focuses on examining the URL structures used in different phishing schemes. They discover that, often, it is easy to determine if a URL is associated with a phishing attempt without needing to be aware of the specifics of the associated website. It explains many characteristics that may be used to differentiate a legitimate URL from a phishing one. These characteristics are utilised to create an accurate and efficient logistic regression filter. Utilise a filter to evaluate the frequency of phishing on the Internet nowadays by thoroughly analyzing several million URLs [25].

Ma et al. [26] identify dangerous URLs using customizable lexical and host-based features. They find the common lexical and host-based features of malicious URLs using statistical approaches, thereby addressing an automated URL categorization problem. In order to detect potentially malicious URLs, we automatically extracted and analyzed hundreds of factors, these methods provide highly predictive models. With just a small percentage of false positives, the resultant classifiers identify a significant number of harmful websites from their URLs with an accuracy of 95–99% [26].

Whittaker et al. [27] describe a scalable ML classifier that updates Google's phishing blacklist. Their in-house classifier analyses URLs and content to scan millions of websites every day for phishing. Both user-supplied

websites and those retrieved by Gmail's spam filters are sorted using their technique. Despite the fact that many URL-based attributes are identical, they test our method using public machine learning algorithms. and datasets and provide a plethora of extra features. In contrast to their approach, the page does not employ any content-based or proprietary aspects.

A content-based approach called CANTINA that utilizes the Robust Hyperlinks algorithm and the TF-IDF algorithm for information retrieval is presented by Zhang et al. [28] to identify phishing websites. A weighted total of eight features—four related to content, three lexical, and one linked to WHOIS—is used to demonstrate that CANTINA can accurately identify around 95% of phishing sites. Their strategy is to lower the possibility of analyzing dangerous stuff on the user's device by avoiding downloading the real web pages. To this end, they assess solely the properties associated with URLs. Numerous research using machine learning are available in related fields, such phishing email detection.

Email headers, sender site WHOIS, email content, URL structures, etc. are among the ten criteria used by Fette et al. [29] to classify phishing emails from legitimate ham communications using Support Vector Machines (SVMs). Through the use of keyword-based feature groups from the email contents, we significantly enhance Fette et al.'s accuracy [29]. They maintain low false positive and negative rates while achieving 98% classification accuracy using various models.

According to Fette et al.'s [29] theory, the categorization of phishing emails could seem like a straightforward text classification issue, However, this becomes more convoluted when given that "phishing" emails are very indistinguishable to legitimate ones [30]. In light of the assumption, base the challenge of classifying phishing emails on the text classification problem from earlier research [31]. With the use of an online method called Confidence Weighted linear classifier and the "bag-of-words" representation provided by the email text alone, they are able to achieve 99% classification accuracy on publicly available benchmark data sets while maintaining a 1% threshold for both false positives and false negatives. In addition to methods based on machine learning (ML), there are several more strategies for detecting phishing emails. The URL blacklist feature, which is included in the majority of contemporary browsers, is perhaps the most popular anti-phishing technology [32] and [33]. Add-in toolbars and browser-based plug-ins are other widely used techniques.

Bhosale et al. (2020) [59]have used intrusion detection on KDD Container 99 using five distinct classification methods, with the help of a feature selection algorithm to determine which features were most important. Based on an analysis of the experimental results, it was shown that the Navies Bayes, CNN, SVM, ANN, and KNN models outperformed other statistical models when it came to intrusion classification accuracy.

Many distributed denial of service attacks target VANET services. In their 2020 study, Adhikary et al. [60] created a hybrid approach that utilises SVM kernel methods like Anova Dot and RBF Dot. Using a simulated real-time dataset, we compared our algorithm's training and testing accuracy to that of individual SVM kernel techniques. Results from the experiments showed that the hybrid algorithm outperformed the separate algorithms in terms of accuracy.

Aamir&Zaidi (2019) [61]have demonstrated many supervised machine learning algorithms for feature selection, including SVM, Random Forest, and KNN. In order to prove that the classifiers with smaller feature sets are accurate, a comparison investigation of their performance has beenconducted. We neglected to consider how parameter adjustment could have an impact.

Sharma et al. [62] to forecast harmful data or attacks based on kernel-based similarity measures. One way to illustrate this pattern of behaviour is by looking at a limited set of traffic features that were seen during a certain activity or period. The IDS was categorised using the modelling methods in this process. Both signature-based and behavior-based detection techniques were employed. The behavior-based approach was called anomaly detection, whereas the signature-based approach used a database of attack signatures. Identifying covert attacks is a breeze with this technique. After measuring the similarity measure of each event's behaviour, the data was

compared to the majority of kernel-based radius closest training processes. If a process's similarity measure was found to be within the range, its behaviour was examined further to determine if it was outside. This study demonstrated a 100% detection rate with a significantly reduced false positive rate. This learning algorithm type zeroed in on a certain kind of threat.

Lin et al. [63] suggested an intrusion detection system (IDS) distributed network-based semantic analysis framework. In order to aid the IDS in estimating the execution repercussions of control instructions, the framework integrated the system knowledge of the power grid's cyber and physical architecture. Packets in the SCADA network were subjected to semantic analysis by the distributed IDS. In order to predict the execution effect of control orders sent over the susceptible SCADA network, the chosen network intrusion detection system instances use the preexisting software for contingency analysis. With the help of trusted communication, distributed IDS instances were set up to detect compromise measurements or manipulated orders. The trustworthy detections of the malicious control commands were offered by the semantic analysis. The semantic analysis framework has successfully exploited specific system vulnerabilities on the IEEE 30-bus architecture. An attacker's system was in a vulnerable state when they used a control command that was purposefully tailored to cause harm. The time it took to conduct the network monitoring and activate the contingency analysis was measured using the semantic analysis. However, the efficiency is diminished by the network's performance.

Prabhjotkaur and Singh [64] implemented a centralised system for identifying and preventing Sybil attacks in WSN. Making two or more identical copies of a node was known as the Sybil node. In order to alter the centralised intrusion detection system's reliance on a misuse recognition algorithm in order to detect the malevolent cluster head responsible for the Sybil attack on the WSN. The inability to identify other routing assaults, such Hello Flood attacks, was the biggest drawback of this effort.

Spoof Guard [34] assesses a webpage's spoof potential by using its domain name, URL, link, and pictures. The plug-in runs a variety of tests, each of which yields a value between 0 and 1. The weighted average of the individual test scores makes up the final score. A user-generated rule-based approach to phishing attack detection has been attempted [35].

According to [7]'s findings, supervised algorithms generally exhibit higher classification accuracy when applied to data that has known threats (the first scenario). The decision tree algorithm has produced the best results out of all of these, with a 95% true positive rate and a 1% false positive rate. The k-nearest neighbour algorithm comes after the next two best algorithms, which are the SVM and MLP. Nonetheless, in the event that test data contains undiscovered attacks, supervised algorithms' detection rate significantly drops. Since there is no discernible difference in accuracy between seen and unseen strikes, the unsupervised approaches perform better in this situation

The foundation of anomaly detection systems is building a model of user behaviour that isregarded as typical. This is accomplished by combining statistical and machine learning techniques to investigate system calls and processes or network traffic. The anomaly detection method is more effective at detecting new attacks than any other.Deviant conduct is categorised as an infringement. On the other hand, typical behaviour in a big, dynamic system is not well defined and varies over time. This frequently leads to a significant number of false positives, or false alarms.

Cloud security has been enhanced by a new technology called intrusion detection. This method was developed recently using machine learning algorithms, which are very useful for monitoring and securing systems. In this paper, we present a method for cloud security intrusion detection using RF and graphic visualisation. Subsequently, features engineering is done with the first, and intrusion detection and prediction are done with the second. We lowered the feature count to two prior to the model's training. The RF classifier outperforms DNN, DT, and SVM in terms of accuracy when it comes to predicting and classifying the type of attack, as demonstrated by the results obtained.

as “in press” [5]. Capitalize only the first word in a paper title, except for proper nouns and element symbols.

Sl no	Authors	Paper Title	Technique used	Conclusion
1	Rachna Dhamija, J.D. Tygar	Phishing protection: Dynamic security skins	Share secret images via remote server using HTML location attributes.	Remote server identification is easy to verify. Phisher spoofing difficult.
2	Min Wu	Fighting User Interface Phishing	25 style identifiers, structural qualities, Apply SVM classifier	All website developers are hard to convince to obey the regulations.
3	Madhusudhananchand rasheran, Krishnan narayanan.	Phishing email detection using structure.	URL-structured extraction	Provide 95% accuracy.
4	SujataGarera, NielsProvos, Monica Chew	Phishing detection and measurement framework	Lexical and host-based URL characteristics	logistic regression filter
5	J. Ma, L.K. Saul, S. Savage, G.M. Voelker	Mastering the art of detecting malicious websites fromquestionable URLs, going beyond blacklist technologies	Scalable machine learning classifier	95.99% accurate prediction
6	C. Whittaker, B. Ryner, M. Nazif	Scalable, automated phishing page classification	Uses 8 characteristics 4 content, 3 lexical, 1 WHOIS	Page content to identify phishing sites

We have researched many methods for identifying rogue URLs. We looked at the numerous techniques applied in the different systems now in use., including features based on web page content, features based on hosts, features extracted from URL structures, and hybrid features that combine multiple features, based on the results of our survey. Thus, hybrid features outperform dispersed features in terms of effectiveness. The system may be trained with ease using a machine learning technique, potentially yielding real good results.

3. INTRUSIONS IN CLOUD

Intrusions seek to access a system or network by breaking its critical infrastructure (CIA). The most common types of cloud-based CIA breaches are as follows: [37]

A. Hypervisor or Virtual Machine Attacks

An attacker might successfully take control of the virtual machines by breaking into the hypervisor.SubVir [38], BLUEPILL [39], and DKSM [40] the most prevalent virtual layer assaults that enable hackers to get control of the host using a hypervisor. Before developers are aware of such exploits [41], cybercriminals use virtual computers' zero-day vulnerabilities to compromise the hypervisor or VMs and access them. Numerous virtual server-based websites suffered harm as because HyperVM zero-day vulnerability was exploited [42].

B. User-to-root Attacks

Attackers may get root access by first gaining access to a valid user's account by password sniffing, and then by taking advantage of security holes. One way to create root shells is to use buffer overflows that occur in processes

running at the root level. An attacker in a cloud environment may compromise valid user instances and then escalate their privileges to root on the host or virtual machines. The security of systems hosted in the cloud is jeopardized by this attack [43].

C. Insider attack

Authorized users who attempt to acquire and abuse officially granted or unassigned rights are considered attackers. The attack is linked to trust since insiders might provide information to enemies. For example, Amazon Elastic Compute Cloud (EC2) was the victim of an internal denial-of-service attack. This hack exposes cloud users' personal information.

D. Port Scanning

In order to execute attacks against the services that are operating on open ports, threat actors may learn about filtered, open, and closed ports by scanning them. Attackers may use port scanning in a cloud environment to find open ports and target the services that are utilizing them [44]. This attack might result in cloud integrity and confidentiality being lost.

E. Backdoor Channel Attacks

Making use of this covert attack, hackers may get remote access to the compromised workstations and jeopardize the privacy of user data. Backdoor channels may be used by hackers in order to launch a denial-of-service attack by acquiring the victim's resources and turning them into zombies. Cloud users' privacy and ease of access are under attack.

F Denial of Service (DoS) attack

By sending a lot of network packets, the attacker takes use of zombies to overload the available resources. As a result, genuine users can't utilise the services provided by the Internet. A denial-of-service (DoS) assault occurs when an attacker in a cloud environment sends zombies' plethora of demands to get access virtual machines (VMs), preventing them from being accessible to authorized users [45]. The availability of cloud resources is the attack's aim.

4. CLOUD-BASED INTRUSION DETECTION SYSTEM

Current Intrusion Detection and Prevention Systems (ID/PS) lack performance and security in a cloud computing environment due to architectural differences between the cloud computing paradigm and other computing approaches, such as Grid computing. The need for an effective system for the safe provisioning of cloud resources is driven by the users' fast increasing demand for these services, since hackers have the potential to disrupt the cloud's security and harm user data stored there. Instead of deploying a typical IDPS, The need of creating an IDPS tailored to the characteristics of the cloud has been brought to light by Patel et al. [53]. Four novel approaches were proposed by the writers for this purpose: topics such as autonomous computing, risk assessment, fuzzy logic, and ontology. The capacity of cloud resources to self-manage on demand is known as autonomous computing. Fuzzy logic operates using degrees, or 0 and 1, between false and true. Rather of using precise numbers, a result is reached by a probabilistic technique. Fuzzy logic is used by the risk management to help conduct vulnerability assessments, handle false positives and determine risk severity so that the right course of action may be taken. The representation of knowledge as a collection of ideas is known as ontology.

Aspects include the detection technique, IDS's placement inside the network, and its settings affect how successful it is [46]. IDSs may be put in the cloud at several points, include at the very periphery of a network, on a host, within a virtual machine, or scattered across the whole cloud. IDS detection techniques might be hybrid, anomaly-based, or signature-based. In order to make signature-based or anomaly-based intrusion detection systems (IDS) more successful, it is possible to use soft computing technologies such as fuzzy logic, ANN, SVM, association rules, and GA [47].

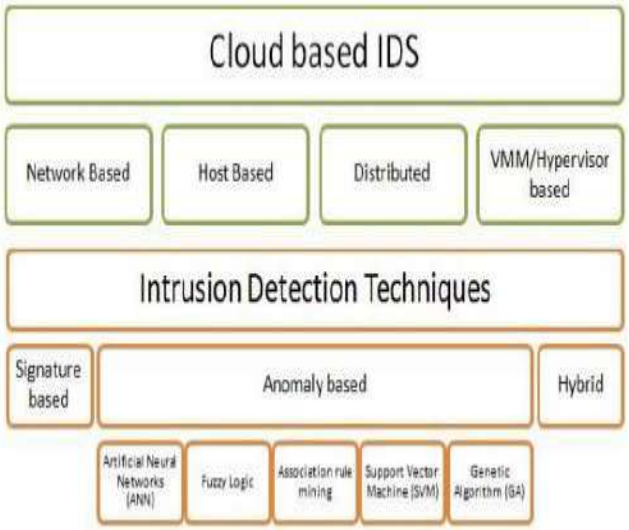

Figure2: Cloud IDS types (green), IDS detection methods (red orange) [48]

4.1. Cloud-Based IDS types

IDS that are cloud-based fall into four categories. In Fig. 2, these kinds are shown (in green). Each of them will be discussed in the next subsections.

i. Network Based IDS

Network-based intrusion detection systems (NIDS) record and analyze all network traffic for port scanning and DOS assaults. Intrusion detection systems (NIDS) check IP and transport layer headers of recorded network packets for breaches. It detects intrusions using anomalies and signatures. Network intrusion detection systems (NIDS) gather network packets and correlate them with attack signatures or match users' actual behaviour in real-time to their pre-identified profiles. A few strategically placed NIDSs may protect many hosts inside the network against intruders. Run in stealth mode, NIDS may conceal its position from potential attackers. If traffic is encrypted, the NIDS cannot analyze it [49]. In a cloud context, Cloud servers with network intrusion detection systems (NIDS) installed may detect attacks on hypervisors and virtual machines (VMs) via their connections to the outside world. The onus for NIDS installation in the cloud rests on the cloud service provider.

ii. Host Based IDS

A host-based intrusion detection system (HIDS) analyses data from one host to detect malicious activity. System logs or operating system audit trails may include this data. If HIDS detects any changes in the system's or program's behaviour after analyzing the data, it will notify the network management that the system is under attack. One way to improve HIDS's detecting capabilities is to find the characteristics that make it work better. Additional storage is required for data analysis, but [50]. Installing HIDS on hosts, virtual machines (VMs), or hypervisors allows them to examine system logs, user login credentials, or access control rules in order to detect intrusion events in cloud computing networks. The responsibility for monitoring hypervisor-installed HIDS at a virtual machine lies with the cloud user, not the cloud provider. Although HIDS can analyze encrypted communications, it may also be turned off and is vulnerable to DoS attacks. Software integrity is often protected using HIDS.

iii. VMM/Hypervisor Based IDS

The role of the hypervisor is to offer a platform for interactions between virtual machines. Security solutions that rely on hypervisors are often implemented at the hypervisor layer. Finding anomalies in the available data becomes much easier with its help. It all comes down to communication, which happens on many levels: within

the virtual network that is based on the hypervisor, and between the hypervisor and the virtual machines themselves [51].

iv. Distributed IDS

Several IDSs, including NIDS and HIDS, are placed across a sizable network to monitor traffic for suspicious activity and form a Distributed Intrusion Detection System (DIDS). A centralized server or the participating IDSs may interact with one another. The two functional components of each of these distinct IDSs are the correlation manager and the detection component. As it monitors the system or network, a common format is used to submit detection results to the correlation manager. The correlation manager uses several IDS to provide attack-relevant alerts. DIDS detects known and unknown threats using anomaly- and signature-based detection methods in the analysis phase. When it comes to clouds, DIDS may be found either at on the server doing the processing or on the host machine [52].

5. ANALYSING CURRENT INTRUSION DETECTION SYSTEMS IN THE CLOUD (CIDS)

We will divide CIDS into three types based on their intrusion detection mechanisms in this section. An anomaly-based, hybrid, and signature-based approach are the three main types. We have looked into and examined systems from every category to determine whether or not they adhere to cloud security standards.

a. Signature Based IDS

IDS that collaborate to thwart DoS and DDoS assaults has been suggested and simulated by C. C. Lo et al. All four of its components work together to accomplish a common goal. In order to detect intrusions, the first one record and analyses network traffic. If a packet matches the rules in the block table, it is instantly discarded. If it does not, it is sent to the alert clustering component, which then decides how serious of an alarm the suspicious packet is. In the third section, intrusion packets are stopped and other IDSs are notified. The proposed intrusion detection system will protect the system against assaults that target a single point of failure. But since it finds intrusions using signature-based detection methodologies, it is unable to identify unknown assaults.

b. Anomaly Based IDS

Using the concepts of autonomic computing, A. Patel et al. have presented an intrusion prevention system based on autonomic agents [54]. The process of detection is anomaly based. In order to spot suspicious events, autonomous sensors keep an eye on system activity such as file access, system calls, and alterations as well as network traffic. The system offers self-management features that need little human interaction, and the agents may be reconfigured at runtime without having to be restarted.

c. Hybrid IDS

A multithreaded Network Intrusion Detection System (NIDS) has been proposed by Ms. Parag K. Shelke et al. [58] as a means of countering Cross Site Scripting (XXS) and Distributed Denial of Service (DDoS) attacks. Each of the three components of the proposed NIDS serves a unique purpose: A shared queue receives UDP, TCP, ICMP, and IP packets from the capture module for analysis. The analysis and processing module evaluates incoming data packets using the knowledge base and pre-set criteria. The NIDS performance is improved by the shared queue's multi-threaded processes. Efficient matching and evaluation help identify harmful packets and provide warnings. The third-party service that is keeping tabs on everything quickly informs the user about the details of the assault and sends a report to the service provider for consultation. Despite being a new technique, the implementation specifics are lacking to validate the notion.

6. CONCLUSION

The present study has offered a thorough analysis of intrusion detection systems (IDS) in relation to cloud computing, highlighting the vital role that security plays in guaranteeing the viability of cloud models. We emphasized the need for strong security measures by outlining many forms of intrusions that endanger cloud systems' Confidentiality, Integrity, and Availability (CIA). Our analysis of several IDS types—there are several strategies used to combat potential attacks, including network-based, host-based, VMM/hypervisor-based, and distributed IDS. By analyzing existing cloud-based intrusion detection systems, we can see the strengths and

shortcomings of signature-based, anomaly-based, and hybrid detection methodologies, offering insightful information to both academics and practitioners [56]. Even if the security picture has clearly improved due to developments in IDS approaches, difficulties still exist. One potential path towards a more complete security framework is the integration of detection methods, such as anomaly-based, signature-based, and soft computing approaches. Specifically, soft computing approaches provide a chance to improve IDS's efficacy and flexibility in dynamic cloud settings. Even with these developments, certain problems persist, including resource-intensive IDS implementations, false alarms from anomaly-based approaches, and performance deterioration in virtual machines [57]. Collaboration between academics and business is necessary to address these issues in order to improve current strategies and provide novel solutions. Going ahead, a top focus should be the creation of a standardized architecture for cloud security that takes into consideration the particular difficulties presented by the cloud environment. Future studies should concentrate on improving current intrusion detection system (IDS) techniques, decreasing false positives, and minimizing performance effects in order to attain a delicate balance between security and system efficiency [58]. In conclusion, even though the area of intrusion detection for the cloud has advanced significantly, further study and cooperation is needed to strengthen cloud computing's security posture and counter new threats. We can only create robust and flexible security architecture for the changing cloud computing environment by means of ongoing research and innovation.Important issues with intrusion detection systems (IDS) have been addressed by research in an effort to increase the security of real-time applications built on WSN and MANET. Starting with the creation of the Multi-Agent Intrusion Detection System (MAIDS), the three-stage approach has demonstrated notable gains in detection accuracy and packet transmission ratio when compared to conventional Network Intrusion Detection Systems (NIDS). Subsequently, the ARMA-IDS was unveiled; it was intended to enhance detection precision in scenarios involving critical infrastructure by employing fuzzy rule generation and feedback loops, with a particular emphasis on real-time datasets like KDD and SCADA.

The novel strategy presented in the previous part, which takes use of the LWCSO-PKM algorithm, has proved successful in improving characteristics and classifying attack types in SCADA systems. Our approach outperforms existing methods and increases detection probability by updating databases. The next stage, according to the research, is to include optimisation algorithms and intrusion detection systems (IDS) into network protocols. These will be essential for future real-time industrial application implementations. The conclusion emphasises the importance of machine learning algorithms in intrusion detection systems (IDS) and draws attention to their capacity to offer reliable defences against dynamic threats, like those that contain malicious URLs. This work establishes the foundation for a more complex and resilient intrusion detection landscape in light of recent advances in machine learning.

REFERENCES

[1] Attou, H., Mohy-eddine, M., Guezzaz, A., Benkirane, S., Azrour, M., Alabdultif, A., &Almusallam, N. (2023). Towards an intelligent intrusion detection system to detect malicious activities in cloud computing. *Applied Sciences*, *13*(17), 9588.

[2] Meryem, A., &Ouahidi, B. E. (2020). Hybrid intrusion detection system using machine learning. *Network Security*, *2020*(5), 8-19.

[3] Vinayakumar, R., Soman, K. P., &Poornachandran, P. (2018). Detecting malicious domain names using deep learning approaches at scale. *Journal of Intelligent & Fuzzy Systems*, *34*(3), 1355-1367.

[4] Arunkumar, M., & Ashok Kumar, K. (2022). Malicious attack detection approach in cloud computing using machine learning techniques. *Soft Computing*, *26*(23), 13097-13107.

[5] Sharon, A., Mohanraj, P., Abraham, T. E., Sundan, B., &Thangasamy, A. (2022, February). An intelligent intrusion detection system using hybrid deep learning approaches in cloud environment. In *International Conference on Computer, Communication, and Signal Processing* (pp. 281-298). Cham: Springer International Publishing.

[6] Aldallal, A., & Alisa, F. (2021). Effective intrusion detection system to secure data in cloud using machine learning. *Symmetry*, *13*(12), 2306.

[7] Kim, A., Park, M., & Lee, D. H. (2020). AI-IDS: Application of deep learning to real-time Web intrusion detection. *IEEE Access*, *8*, 70245-70261.

[8] Alshammari, A., &Aldribi, A. (2021). Apply machine learning techniques to detect malicious network traffic in cloud computing. *Journal of Big Data*, *8*(1), 1-24.

[9] Selvapandian, D., & Santhosh, R. (2021). Deep learning approach for intrusion detection in IoT-multi cloud environment. *Automated Software Engineering*, *28*, 1-17.

[10] Afzal, S., Asim, M., Javed, A. R., Beg, M. O., & Baker, T. (2021). Urldeepdetect: A deep learning approach for detecting malicious urls using semantic vector models. *Journal of Network and Systems Management*, *29*, 1-27.

[11] Ferrag, M. A., Maglaras, L., Moschoyiannis, S., &Janicke, H. (2020). Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study. *Journal of Information Security and Applications*, *50*, 102419.

[12] Vu, L., Nguyen, Q. U., Nguyen, D. N., Hoang, D. T., &Dutkiewicz, E. (2022). Deep generative learning models for cloud intrusion detection systems. *IEEE Transactions on Cybernetics*, *53*(1), 565-577.

[13] Bingu, R., &Jothilakshmi, S. (2023). Design of intrusion detection system using ensemble learning technique in cloud computing environment. *International Journal of Advanced Computer Science and Applications*, *14*(5).

[14] Asharf, J., Moustafa, N., Khurshid, H., Debie, E., Haider, W., &Wahab, A. (2020). A review of intrusion detection systems using machine and deep learning in internet of things: Challenges, solutions and future directions. *Electronics*, *9*(7), 1177.

[15] Uğurlu, M., &Doğru, İ. A. (2019, September). A survey on deep learning based intrusion detection system. In *2019 4th International Conference on Computer Science and Engineering (UBMK)* (pp. 223-228). IEEE.

[16] Bhingarkar, S., Revathi, S. T., Kolli, C. S., &Mewada, H. K. (2023). An effective optimization enabled deep learning based Malicious behaviour detection in cloud computing. *International Journal of Intelligent Robotics and Applications*, *7*(3), 575-588.

[17] Tian, Z., Luo, C., Qiu, J., Du, X., &Guizani, M. (2019). A distributed deep learning system for web attack detection on edge devices. *IEEE Transactions on Industrial Informatics*, *16*(3), 1963-1971.

[18] Muna, A. H., Moustafa, N., &Sitnikova, E. (2018). Identification of malicious activities in industrial internet of things based on deep learning models. *Journal of information security and applications*, *41*, 1-11.

[19] Mani, S., Sundan, B., Thangasamy, A., & Govindaraj, L. (2022). A new intrusion detection and prevention system using a hybrid deep neural network in cloud environment. In *Computer Networks, Big Data and IoT: Proceedings of ICCBI 2021* (pp. 981-994). Singapore: Springer Nature Singapore.

[20] Alani, M. M., &Tawfik, H. (2022). PhishNot: a cloud-based machine-learning approach to phishing URL detection. *Computer Networks*, *218*, 109407.

[21] Srilatha, D., &Thillaiarasu, N. (2023). Implementation of Intrusion detection and prevention with Deep Learning in Cloud Computing. *Journal of Information Technology Management*, *15*(Special Issue), 1-18.

[22] Lee, S. W., Mohammadi, M., Rashidi, S., Rahmani, A. M., Masdari, M., &Hosseinzadeh, M. (2021). Towards secure intrusion detection systems using deep learning techniques: Comprehensive analysis and review. *Journal of Network and Computer Applications*, *187*, 103111.

[23] Al-Ghuwairi, A. R., Sharrab, Y., Al-Fraihat, D., AlElaimat, M., Alsarhan, A., &Algarni, A. (2023). Intrusion detection in cloud computing based on time series anomalies utilizing machine learning. *Journal of Cloud Computing*, *12*(1), 127.

[24] Liu, H., & Lang, B. (2019). Machine learning and deep learning methods for intrusion detection systems: A survey. *applied sciences*, *9*(20), 4396.

[25] Saba, T., Rehman, A., Sadad, T., Kolivand, H., &Bahaj, S. A. (2022). Anomaly-based intrusion detection system for IoT networks through deep learning model. *Computers and Electrical Engineering*, *99*, 107810.

[26] Kassem, A. K. (2021). *Intelligent system using machine learning techniques for security assessment and cyber intrusion detection* (Doctoral dissertation, Universitéd'Angers).

[27] Chen, W., Zeng, Y., &Qiu, M. (2019, December). Using adversarial examples to bypass deep learning based url detection system. In *2019 IEEE International Conference on Smart Cloud (SmartCloud)* (pp. 128-130). IEEE.

[28] Liu, Z., Xu, B., Cheng, B., Hu, X., &Darbandi, M. (2022). Intrusion detection systems in the cloud computing: A comprehensive and deep literature review. *Concurrency and Computation: Practice and Experience*, *34*(4), e6646.

[29] Kim, J., Shin, Y., & Choi, E. (2019). An intrusion detection model based on a convolutional neural network. *Journal of Multimedia Information System*, *6*(4), 165-172.

[30] Ali, R., Ali, A., Iqbal, F., Hussain, M., &Ullah, F. (2022). Deep learning methods for malware and intrusion detection: A systematic literature review. *Security and Communication Networks*, *2022*.

[31] Sokolov, S. A., Iliev, T. B., &Stoyanov, I. S. (2019, May). Analysis of cybersecurity threats in cloud applications using deep learning techniques. In *2019 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)* (pp. 441-446). IEEE.

[32] Mighan, S. N., &Kahani, M. (2021). A novel scalable intrusion detection system based on deep learning. *International Journal of Information Security*, *20*, 387-403.

[33] Krishna, A., Lal, A., Mathewkutty, A. J., Jacob, D. S., & Hari, M. (2020, July). Intrusion detection and prevention system using deep learning. In *2020 International Conference on Electronics and Sustainable Communication Systems (ICESC)* (pp. 273-278). IEEE.

[34] Guezzaz, A., Asimi, A., Asimi, Y., Azrour, M., &Benkirane, S. (2021). A distributed intrusion detection approach based on machine leaning techniques for a cloud security. In *Intelligent Systems in Big Data, Semantic Web and Machine Learning* (pp. 85-94). Cham: Springer International Publishing.

[35] Kumaar, M. A., Samiayya, D., Vincent, P. D. R., Srinivasan, K., Chang, C. Y., & Ganesh, H. (2021). A hybrid framework for intrusion detection in healthcare systems using deep learning. *Frontiers in Public Health*, *9*.

[36] Luo, C., Su, S., Sun, Y., Tan, Q., Han, M., & Tian, Z. (2020). A Convolution-Based System for Malicious URLs Detection. *Computers, Materials & Continua*, *62*(1).

[37] Kanimozhi, V., & Jacob, T. P. (2021). Artificial Intelligence outflanks all other machine learning classifiers in Network Intrusion Detection System on the realistic cyber dataset CSE-CIC-IDS2018 using cloud computing. *ICT Express*, *7*(3), 366-370.

[38] Aslan, Ö., Ozkan-Okay, M., & Gupta, D. (2021). Intelligent behavior-based malware detection system on cloud computing environment. *IEEE Access*, *9*, 83252-83271.

[39] Wang, Y. C., Houng, Y. C., Chen, H. X., & Tseng, S. M. (2023). Network Anomaly Intrusion Detection Based on Deep Learning Approach. *Sensors*, *23*(4), 2171.

[40] Shareena, J., Ramdas, A., & AP, H. (2021). Intrusion detection system for iot botnet attacks using deep learning. *SN Computer Science*, *2*(3), 1-8.

[41] Zhong, M., Zhou, Y., & Chen, G. (2021). Sequential model based intrusion detection system for IoT servers using deep learning methods. *Sensors*, *21*(4), 1113.

[42] Cui, B., He, S., Yao, X., & Shi, P. (2018). Malicious URL detection with feature extraction based on machine learning. *International Journal of High Performance Computing and Networking*, *12*(2), 166-178.

[43] Sasikumar, S. (2021). Network intrusion detection and deduce system. *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, *12*(9), 404-410.

[44] AkshayKumaar, M., Samiayya, D., Vincent, P. M., Srinivasan, K., Chang, C. Y., & Ganesh, H. (2022). A Hybrid Framework for Intrusion Detection in Healthcare Systems Using Deep Learning. *Frontiers in Public Health*, *9*, 824898.

[45] Sethi, K., Kumar, R., Mohanty, D., &Bera, P. (2020). Robust adaptive cloud intrusion detection system using advanced deep reinforcement learning. In *Security, Privacy, and Applied Cryptography Engineering: 10th International Conference, SPACE 2020, Kolkata, India, December 17–21, 2020, Proceedings 10* (pp. 66-85). Springer International Publishing.

[46] Kolli, C. S., Ranjan, N. M., Talapula, D. K., Gawali, V. S., & Biswas, S. S. (2022). Multiverse fractional calculus based hybrid deep learning and fusion approach for detecting malicious behavior in cloud computing environment. *Multiagent and Grid Systems*, *18*(3-4), 193-217.

[47] Yan, X., Xu, Y., Cui, B., Zhang, S., Guo, T., & Li, C. (2020). Learning URL embedding for malicious website detection. *IEEE Transactions on Industrial Informatics*, *16*(10), 6673-6681.

[48] Parra, G. D. L. T., Rad, P., Choo, K. K. R., & Beebe, N. (2020). Detecting Internet of Things attacks using distributed deep learning. *Journal of Network and Computer Applications*, *163*, 102662.

[49] Veeraiah, D., Mohanty, R., Kundu, S., Dhabliya, D., Tiwari, M., Jamal, S. S., &Halifa, A. (2022). Detection of malicious cloud bandwidth consumption in cloud computing using machine learning techniques. *Computational Intelligence and Neuroscience*, *2022*.

[50] Jha, P., & Sharma, A. (2021, January). Framework to analyze malicious behaviour in cloud environment using machine learning techniques. In *2021 International Conference on Computer Communication and Informatics (ICCCI)* (pp. 1-12). IEEE.

[51] Ahmad, Z., Shahid Khan, A., Wai Shiang, C., Abdullah, J., & Ahmad, F. (2021). Network intrusion detection system: A systematic study of machine learning and deep learning approaches. *Transactions on Emerging Telecommunications Technologies*, *32*(1), e4150.

[52] Awotunde, J. B., Abiodun, K. M., Adeniyi, E. A., Folorunso, S. O., &Jimoh, R. G. (2021, November). A deep learning-based intrusion detection technique for a secured IoMT system. In *International Conference on Informatics and Intelligent Applications* (pp. 50-62). Cham: Springer International Publishing.

[53] Fontaine, J., Kappler, C., Shahid, A., &Poorter, E. D. (2020). Log-based intrusion detection for cloud web applications using machine learning. In *Advances on P2P, Parallel, Grid, Cloud and Internet Computing: Proceedings of the 14th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC-2019) 14* (pp. 197-210). Springer International Publishing.

[54] Shah, H., Shah, D., Jadav, N. K., Gupta, R., Tanwar, S., Alfarraj, O., ... & Marina, V. (2023). Deep learning-based malicious smart contract and intrusion detection system for IoT environment. *Mathematics*, *11*(2), 418.

[55] Lansky, J., Ali, S., Mohammadi, M., Majeed, M. K., Karim, S. H. T., Rashidi, S., ... &Rahmani, A. M. (2021). Deep learning-based intrusion detection systems: a systematic review. *IEEE Access*, *9*, 101574-101599.

[56] Yang, W., Zuo, W., & Cui, B. (2019). Detecting malicious URLs via a keyword-based convolutional gated-recurrent-unit neural network. *IEEE access*, *7*, 29891-29900.

[57] Rizvi, S., Scanlon, M., McGibney, J., & Sheppard, J. (2022, November). Deep learning based network intrusion detection system for resource-constrained environments. In *International Conference on Digital Forensics and Cyber Crime* (pp. 355-367). Cham: Springer Nature Switzerland.

[58] Diwan, T. D., Choubey, S., Hota, H. S., Goyal, S. B., Jamal, S. S., Shukla, P. K., & Tiwari, B. (2021). Feature entropy estimation (FEE) for malicious IoT traffic and detection using machine learning. *Mobile Information Systems*, *2021*, 1-13.

[59] Bhosale, KS, Nenova, M &Iliev, G 2020, 'Intrusion Detection in Communication Networks Using Different Classifiers', In TechnoSocietal 2018, Springer, pp. 19-28.

[60] Adhikary, K, Bhushan, S, Kumar, S &Dutta, K 2020, 'Hybrid Algorithm to Detect DDoS Attacks in VANETs', Wireless Personal Communications, pp.1-22.

[61] Aamir, M &Zaidi, SMA 2019, 'DDoS attack detection with feature engineering and machine learning: the framework and performance evaluation', International Journal of Information Security,vol.18, no.6, pp.761-785.

[62] A. Sharma, AK. Pujari and KK. Paliwal, Intrusion Detection using Text Processing Techniques with a Kernel based Similarity Measure, Computers &Security, 26 (7-8) (2007) 488-495.

[63] H. Lin, A. Slagell, Z. PW. Kalbarczyk Sauer and RK. Iyer, Semantic Security Analysis of Scada Networks to Detect Malicious Control Commands in Power Grids, Proceedings of the first ACM Workshop on Smart Energy Grid Security, (2013), 29-34.

[64] AC. Prabhjotkaur and S. Singh, Review Paper of Detection and Prevention of Sybil Attack in WSN using Centralizedids, International Journal of Engineering Science, 8399, (2016).

[65] Hanaa Attou, AzidineGuezzaz_, Said Benkirane, MouradeAzrour, and Yousef Farhaoui, Cloud-Based Intrusion Detection Approach UsinMachine Learning Techniques

[66] Mahdi Zamani zamani@cs.unm.edu Department of Computer Science University of New Mexico, Machine Learning Techniques for Intrusion Detection