Hindawi
Mobile Information Systems
Volume 2022, Article ID 2600712, 13 pages
https://doi.org/10.1155/2022/2600712

Research Article

Multi-Cloud Integration Security Framework Using Honeypots

Tahir Alyas,[1] **Khalid Alissa**,[2] **Mohammed Alqahtani**,[3] **Tauqeer Faiz**,[4] **Suleiman Ali Alsaif**,[5] **Nadia Tabassum**,[6] **and Hafiz Hasan Naqvi**[1]

[1]*Department of Computer Science, Lahore Garrison University, Lahore 54000, Pakistan*
[2]*Networks and Communications Department, College of Computer Science and Information Technology, Imam Abdulrahman Bin Faisal University, P.O. Box 1982, Dammam 31441, Saudi Arabia*
[3]*Department of Computer Information Systems, College of Computer Science and Information Technology, Imam Abdulrahman Bin Faisal University, P.O. Box 1982, Dammam 31441, Saudi Arabia*
[4]*School of IT, Skyline University College, Sharjah, UAE*
[5]*Computer Department, Deanship of Preparatory Year and Supporting Studies, Imam Abdulrahman Bin Faisal University, P.O. Box 1982, Dammam 31441, Saudi Arabia*
[6]*Department of Computer Science, Virtual University of Pakistan, Lahore 54000, Pakistan*

Correspondence should be addressed to Nadia Tabassum; nadiatabassum@vu.edu.pk

Received 22 April 2022; Revised 23 June 2022; Accepted 23 July 2022; Published 17 August 2022

Academic Editor: Marco Anisetti

This rapidly changing digital world is always sensitive to improving security and resilience to protect the inhabitants of this ecosystem in terms of data, processes, repositories, communication, and functions. The transformation of this digital ecosystem is heavily dependent on cloud computing, as it is becoming the global platform for individuals, corporates, and even governments. Therefore, the concerns related to security are now linked closely with cloud computing. In this paper, a multi-cloud security framework takes a view on the development of security mechanisms to provide a diversion to the attacker. The purpose is to gain more time to analyze the attack and mitigate the intrusion without compromises. This mechanism is designed using the honeypot technology that has been around for some time but has not been used in cloud computing and other technologies. The proposed framework provides modules related to managing the multi-cloud platform, the intrusion detection and prevention system, and honeypots. The results show significant improvement in the accuracy of detecting attacks. These results are generated in a two-phase scenario, and the first phase has been analyzed without the engagement of the honeypot module presented in the framework. The second phase has been executed with same parameters and conditions by engaging the honeypot module. It includes a comparison taxonomy of both results and an in-depth study of existing honeypots, as well as critical design elements for current honeypot research and outstanding concerns for future honeypots in IoT, multi-cloud contexts.

1. Introduction

The emergence of cloud computing has brought many new challenges from IT management to the development tier. The Internet development has brought about huge changes in cloud computing and has developed into an open, collaborative business model that looks for services and further diversifies other energy sources. Microservices gained importance to address the software-as-a-service (SaaS) segment, smarter and unstructured databases, incorporation of knowledge repositories, and more complex architectures to deal with the question of big data and the Internet of Things (IoT). One essential and integral segment of this progress is the security concerns related to data at rest, in transit, and the attacks and compromises on digital assets. The concept of security is not merely limited to dealing with virus attacks; it is now more sophisticated and multifaceted. In the digital world, the encryption regime requires continuous development as these methods are being compromised now and then. Similarly, cyber security, database security, and cloud security mechanisms are essential and shall be developed with cutting-edge technologies. Still, the dynamism in the digital ecosystem develops this concept to engage more intelligent and complex preventive measures to divert hacking attempts [1].

Cloud computing is a flourished domain, and multiple definitions are almost standardized as the properties and structure of cloud computing are standardized. Cloud computing provides computer services, from applications to processing and storage capacity, usually via the Internet and on pay-as-you-use price slabs. On this particular point, i.e., hiring the infrastructure instead of developing your own, businesses do not have to have their own IT infrastructure or data centers. Still, they can rent everything from applications to storage to cloud providers. One of the advantages of engaging in cloud computing is that the business and industry sector can reduce the capital expenditure involved in setting up the IT infrastructure and platforms. Instead, hiring the required services, infrastructure, and facilities is possible while paying only for the usage. On the other hand, cloud service providers (CSPs) can develop state-of-the-art infrastructure and extend the same facility to many customers/users, making it more profitable [2].

Cloud computing is the basis of many services. Cloud computing depicted that from primary storage, networking and dispensation abilities to "natural language processing, artificial intelligence, and standard office applications, cloud computing services" now offer a wide range of capabilities. People can now offer almost any service that does not entail the physical proximity of cloud-based IT hardware. This includes using services such as cloud backup in Gmail or smartphones and services permitting large corporations to keep all their data in the cloud and run all their solicitations. Netflix also depends on cloud services to run the benefits of video streaming and other occupational systems and has many other organizations [3].

Cloud computing has brought an entirely new structure for software developers and products. Developing multi-user, cloud-enabled applications is more lucrative than a standalone and single-user software entity. However, cloud computing has potential drawbacks, as it also brings new costs and risks to the companies that use it. The principle of cloud computing is that the place of the service and many particulars (such as the hardware or operating system running the service) have nothing to do with the user. With this in mind, a metaphor is rented from the old telecommunications network, where the public network (and later the Internet) is described as a cloud to show that justice does not matter [4].

Many companies are still worried about cloud service security, though there are few security breaches. In a team-managed internal system, you can fear many other problems that are more likely to occur than under the supervision of cloud engineers specializing in infrastructure protection, as shown in Figure 1. However, security concerns remain, especially for companies that transfer data among multiple cloud services. This has led to the growth of cloud security tools that can monitor cloud-to-cloud and cross-platform migration. These tools can indicate fraudulent use of cloud data, unauthorized downloads, and malicious software. However, it affects finances and performance: these tools can reduce cloud-based profitability by 5–10% and 5–15% [6].

IaaS refers to the essential information technologies leased, including computing servers, storage hardware, and network layers. However, half people said that the IaaS might not be protected sufficiently for the most critical data. This is stimulating for businesses who want to build modern IT infrastructure from scratch and want to do almost everything themselves. Still, this plan pushed the businesses to have the technical skills to coordinate services at this stage [7]. This service segment is the broadest and most detailed service structure consisting of computation hardware/infrastructure facilities. These services or the hardware is provided to the end-user in virtualization. Infrastructure-as-a-service providers are responsible for delivering the physical structures, e.g., processors, memory, servers, storage. The physical structure behind this virtualization is a data center containing multiple servers, computation capability, storage in hard drives, and memory provided to end-users to configure the respective virtual machines of their preferred configuration [8].

1.1. Emergence of Multi-Cloud. Cloud deployment models are developed and used in the best interest of the end-user even than the end-user always seeks for more benefits and better quality. As the market is filled with many CSPs and provision of many services and configurations, therefore, end-user has a split opinion, i.e., prefer specific services of a CSP but like the infrastructure of another CSP and vice versa.

Figure 2 shows multiple clouds that form an interconnection cloud system, creating a virtual multi-cloud environment. Each cloud participating in forming a cloud system consists of all basic cloud characteristics like VM (virtual machine), hypervisor, and specific service (memory, storage, network). The same is applicable on pricing slabs, service quality, and performance measures. Therefore, end-users prefer resources from multiple CSPs at one place to have the optimum benefits from each service. There are multiple reasons due to why many organizations are willing and investing in a multi-cloud environment. By assessing potential pitfalls, expectations, and negotiating positions that make it easy to switch from one cloud provider to another, connect the cloud's power and derive the most value from their partnership with any cloud service provider [9].

The challenges of implementing multi-cloud integration are as follows:

(i) Data governance and compliance.

(ii) Multiple skillsets and vendors to manage.

(iii) Software development and delivery.

(iv) Data redundancy and security.

(v) Cost controls and cloud sprawl.

For several reasons, business organizations depend on multi-cloud infrastructure.

(i) For the avoidance of vendor lock-in.

(ii) Achieving broader technical and business goals includes using more price-competitive cloud services.

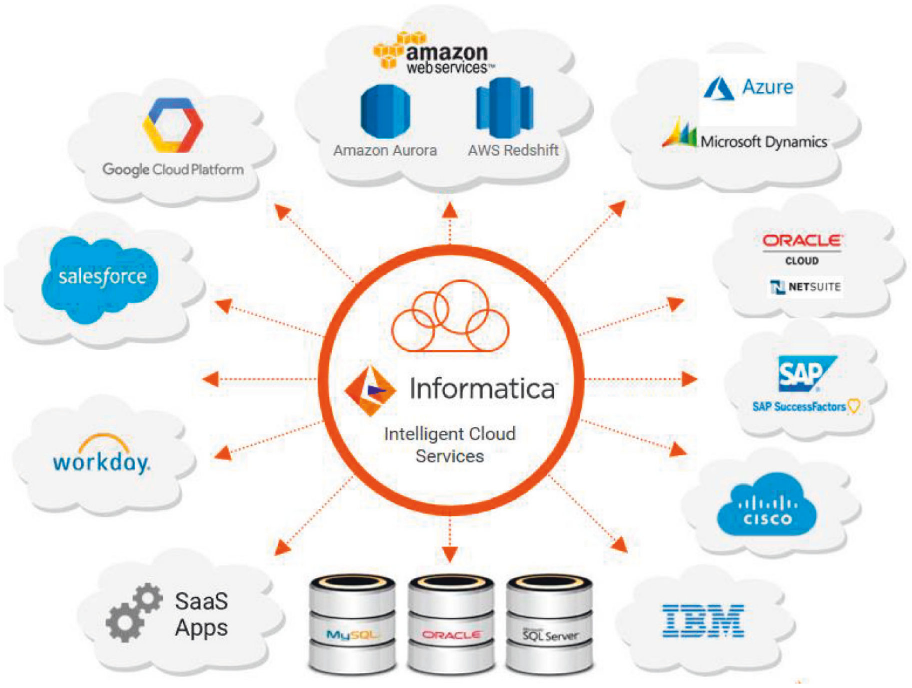

Figure 1: Multi-cloud integration mode [5]. However, security concerns remain, especially for companies that transfer data among multiple cloud services. This has led to the growth of cloud security tools that can monitor cloud-to-cloud and cross-platform migration. These tools can indicate fraudulent use of cloud data, unauthorized downloads, and malicious software. However, it affects finances and performance: these tools can reduce cloud-based profitability by 5–10% and 5–15% [6].

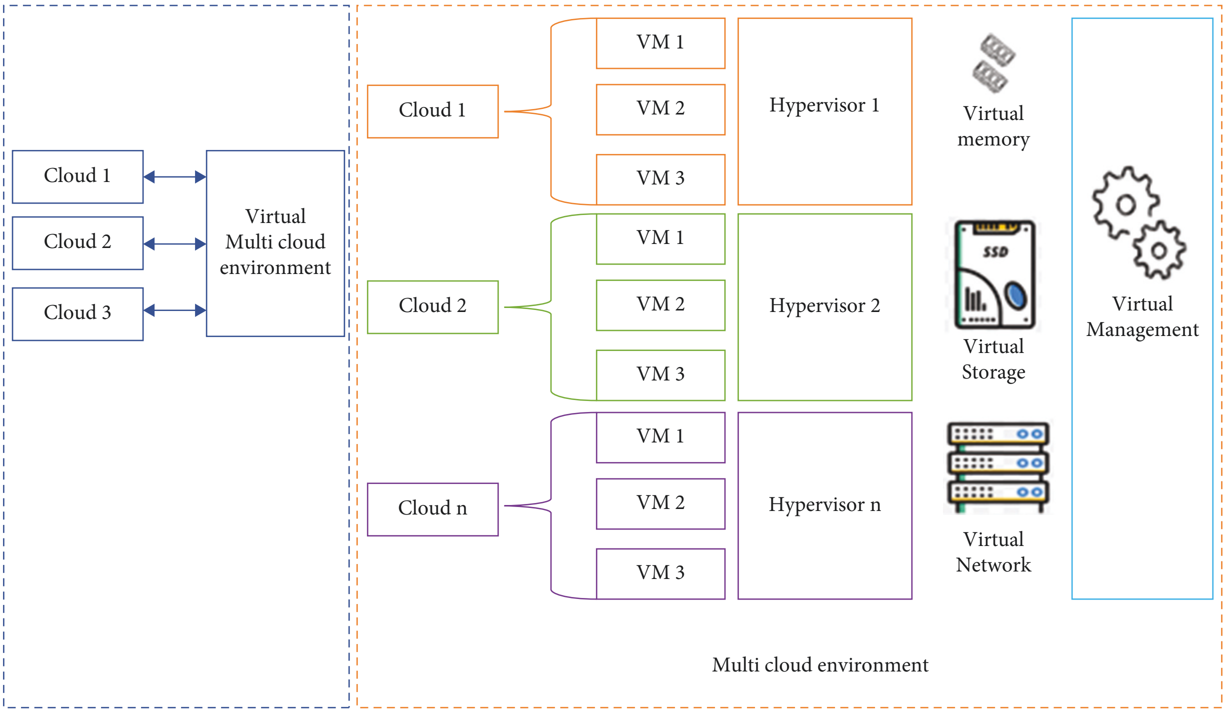

Figure 2: Multi-cloud design.

(iii) For minimal latency and optimal performance by data, sovereignty enables organizations to locate compute resources as close as possible to the end-user.

As demonstrated in Figure 3, a honeypot is a device built to run on either hardware or software with known vulnerabilities and attack surfaces to trap and entice in attackers.

2. Statement of the Problem

Cloud computing has brought an entirely new digital ecosystem to the IT community; its security is a complex challenge due to the nature of the cloud ecosystem. It provides infrastructure, platform, and software as services while maintaining the user data. The same segments require strict and carefully designed security measures in terms of security methods and mechanisms that may have low process costs and high accuracy. This paper provides a security mechanism to divert and capture unknown threats without compromising the performance and operations of the cloud platform. It is also essential to security day-to-day cloud usage for a single user or a corporate user alike.

3. Research Gap and Significance of the Study

The swift spread of cloud computing and the transformation of conventional digital systems to cloud platforms has brought many challenges related to data processing, distribution and management frameworks, and security. The fine line between security methods and mechanisms is essential to the research community. The security methods are being developed, deployed, and compromised, it is an ongoing process of having more advanced methods, and within due time, the same method has been compromised. On the other hand, it is also known that managing a threat requires time and knowing patterns to address the vulnerability. This is the gap that more mechanisms are required to gain more time and capture more threats to develop new security methods.

This research work contributes to developing the framework for a multi-cloud environment using honeypots to enhance the preventive measures against attacks. The proposed framework not only provides the conventional methodologies for intrusion detection, but also provides the dynamic range of algorithms to identify the threat level and accordingly behave. The honeypots are used in different scenarios, but multi-cloud intrusion prevention with a dynamic detection approach is not being presented.

4. Related Work

Cloud computing is a sort of computing that gives customers on-demand or pay-per-use access to a shared pool of computer resources. These services are mainly categorized into three different forms such as software-as-a-service (SaaS), platform-as-a-service (PaaS), and infrastructure-as-a-service (IaaS). CC has changed the overall picture of the IT world by offering its services with matchless features like virtualization, broad network access, resource pooling, on-demand self-service, easy maintenance, availability, rapid elasticity and scalability, economical, reliability, pay-as-you-go, security, and measured services. As a result, it has dramatically grabbed the attention of enterprises and urged them to adopt it rather than investing a huge amount of money on physical IT resources [10].

As the Internet becomes increasingly popular daily, security is shifting from being a secondary concern to taking center stage. The technologies are known as server honeypots, and client phishing sites are used in this study to propose an integrated architecture of malware gathering and processing. While client honeypots help us understand client side assaults, server honeypots help us understand server side threats [11]. Analyzing malware samples gathered from honeypots was the major objective of our investigation on honeypot technologies. To do this, malware samples from both client and server honeypots are analyzed [12].

Various researchers use intrusion detection systems and honeypot technologies to develop solutions and propose a mechanism to address multiple problems faced by intrusion detection systems. Similarly, a signature generator is proposed to secure the digital networks for honeypot technology. It is also notable that the exact mechanism collected information related to attacks and suspicious activities that helped develop resilience against vulnerabilities, especially against unknown worms and attacks. This work highlights the gap in developing more intrusion detection system mechanisms [13].

Honeypot integration with intrusion detection systems delivers good results. Another exciting mechanism was presenting a framework based on virtualization in cloud computing and developing nested virtual machines. For intrusion detection, they have placed honeypots on the nested virtual machines—this mesh of virtualization, honeypots, and interdependent virtual machines consisting of honeypots. Similarly, the intrusion detection system has incorporated IP tracing back mechanism [14].

The intrusion detection focuses on honeypots having IP trackback signatures that enhance conventional intrusion detection capacity. In a distributed network environment, the author performed a detailed experiment to analyze on different configurations. The results were encouraging and highlighted the honeypots as a better and more stable prevention system in collaboration with intrusion detection systems [15].

Analytical capability combined with efficient anomaly detection is a key target for research worldwide. [16] proposed the integration of honeypots in a distributed environment that is highly scalable and interactive in maintaining quick attack detection as well as performing analysis for better performance. The main objective of their work is to manage distributed service blocking by intruders. Their work has endorsed the idea that the attacks related to distributed denial of service (DDoS) can be prevented with the help of honeypots more successfully than the conventional intrusion detection systems [17].

As technology changes, open-source development and solutions have become essential and cost-effective. A significant contribution [18] discussed the honeypots in an

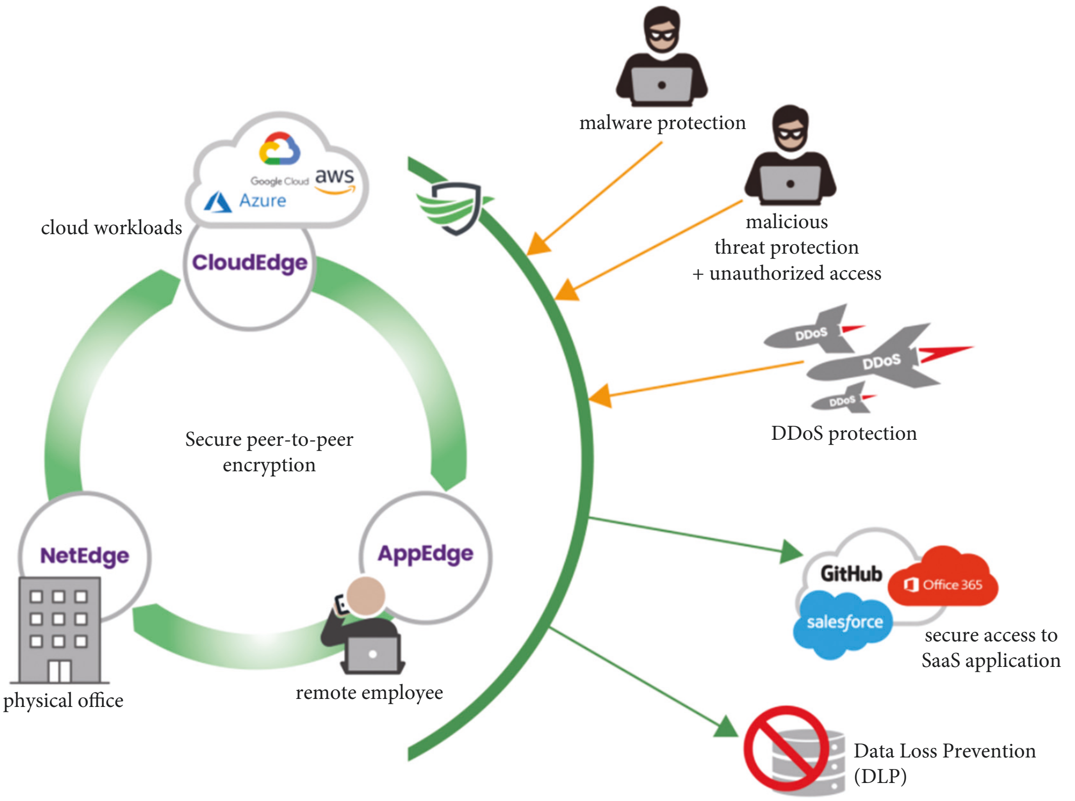

Figure 3: Multi-cloud threat scenario.

open-source environment. They have delivered SNORT, OSSEC, and honeypot while utilizing various machine learning techniques and algorithms for attack predictions and analytics. It is also notable that in a single configuration, only firewall, intrusion detection system, intrusion prevention system, or honeypots are not successful or not too much impactful. Therefore, it is recommended to engage multiple mechanisms to trade off the weaknesses and strengths [19].

A key issue is accurately describing the statistical characteristics of cyber attacks. In [20] article, we provide the first statistical approach for thoroughly examining data from cyber attacks that honeypots have detected. A new class of mathematical objects for characterizing cyber attacks, the stochastic cyber attack process, serves as the foundation for the framework, while emphasizing that the system may also be used to analyze high-interaction honeypot data, which provides deeper information on the assaults [21].

Specifically, the data volume has increased tremendously in scientific domains, making it difficult for traditional data management and processing methodologies to cater to this data. The computation demands for such heavy data volumes are becoming specialized. High-performance computing (HPC) and cloud computing are two segments aligned to provide solutions [22].

A security facility specifically designed to be examined, attacked, and hacked is known as a honeypot. Spotting and blocking unwanted accesses are often used to safeguard production systems. Additionally, it helps analyze how attackers behave, particularly while carrying out unidentified assaults [23].

The purpose of cyber deception is to deceive attackers by misrepresenting the network's health, falsifying their reconnaissance results, and diverting them away from their intended targets. Honeypots are decoy devices that may be placed within networks to capture attackers for surveillance reasons. Based on honeypot allocation, we suggest a two-phase deception strategy [24].

DDoS (distributed denial of service) assault flood targeted nodes with traffic from many sources. Assaults with a low rate of occurrence cause the network to degrade gracefully, but attacks with a high rate of occurrence cause the network to become functionally unstable. Previous responses to such assaults have reached a point of ineffectiveness. Survivable systems attempt to lessen the impact of these assaults. Increased reaction times and network congestion breakdowns plague the network. Furthermore, the Internet is fluid, and the problem of scripted responses to assaults has seen little attention [25].

5. Proposed Design and Implementation

Many researchers mentioned and exposed the effectiveness and suitability of honeypots in various configurations in cloud platforms. As the objective of this paper is to work on practical measures of prevention in the domain of security and resilience, a framework is proposed. The proposed framework is developed in a modular structure keeping the multi-cloud environment focused. It is also notable that the proposed framework mitigates the weaknesses of previously presented mechanisms for intrusion detection using honeypots as shown in Figure 4. Considering the adaptation of multi-cloud platforms by the corporate sector at a rapid pace is pointing toward a near-future eruption of security needs and an increase in cloud resilience demand.

6. Proposed Framework

Figure 5 depicts the modular details of the proposed framework. It is visible that there are four main modules of the proposed framework, namely,

(i) Authenticity controller.

(ii) Honeypots.

(iii) Production system.

(iv) Cloud ecosystem.

As mentioned previously, various experiments have shown the failure of a single configuration of IDS and honeypots; therefore, in our proposed framework, the intrusion detection system is configured with honeypots and several of its components. Similar to this paper's objective, the framework can detect anomalies and capture unknown patterns by presenting a decoy system to the intruder. As mentioned in the figure below, a logical flow can also significantly contribute toward detecting, capturing, and flagging the anomaly. The complexity of the multi-cloud platform has also been addressed in the proposed framework as shown in Figure 5. It is also notable from the following diagram that the cloud ecosystem is incompletely encapsulated, while the multi-cloud management, intrusion detection, and honeypots are working in three tiers to ensure the prevention and protection from known/unknown threats.

The authenticity controller module is the first to interact with the external stimulus. This module aims to be the first tier of prevention by letting the authentic user into the system. The unknown new threats and attacks may dodge that; therefore, the optimum effort is to have a strict and vigilant entry into the system.

The analysis of influx packets identifies the behavior, pattern, properties, and level of anomaly. Suppose a usual request is generated but creates an exception in request time. The IDS will take it as an anomaly and monitor this kind of request, the behavioral anomaly. Similarly, if a packet generates a previously unavailable pattern, the authenticity controller will also start observing such patterns. The anomaly level is also a complex structure as the most critical factor in intrusion is evaluating the threat. In multi-cloud, it is always situational to identify the risk related to a threat.

The segmentation of honeypots used previously and familiar with the previously known attacks is available as honeywall that is continuously active. In case of a positive from the intrusion detection module, it will start working to provide or direct the request to suitable honeypots. The objective is to capture the unknown patterns and make the attack cluster if known patterns are available. honeywall performed well in system behavior in case of vulnerability. The honeywall provides the initial parameters to the honeypot framework and replica RACS to develop a relevant and suitable decoy to the current threat.

The proposed framework is catering multi-cloud platform, and the authenticity controller has the multi-cloud manager to identify the threats, anomalies, and attacks on certain services, nodes, cloud service providers, service models, etc. Authenticity controller extended the influx packets to the multi-cloud manager, which contains the meta properties of all member clouds and services. Instead of confirming the authenticity from the specific member cloud or service, the authenticity controller only uses the already available user directory to identify the incoming user request. Authenticity deployed by the cloud service provider is a separate process that shall be activated once the user reaches the production system. Three modules are available to develop the decoy for the intruder to get optimum time to evaluate and identify the threat type and select the correct preventive action. This module is constructed on multiple sub-modules representing processes and functions this module is supposed to perform. This module identifies and stores the threat signature and pattern analysis and maintains the log for unauthorized access.

It performs as an observer to keep track of system participants, the requests, and packets flagged by the intrusion detection module. Analyze the influx to declare the current packet and system-level status. In case of an anomaly, the event editor declared the system's intrusion state and started the flagged activities to stop or mitigate the intrusion without compromising digital assets and data in transit.

As the framework deals with multi-cloud environments and known and unknown threats, it is essentially required to have a dynamic rule base, which shall contain the rules against certain threats and mitigation. This module analyzed the packets to evaluate if there is any property or activity against the rules available in the database. It provides the results to the intrusion detection module and event editor module. It also caters the behavioral patterns by comparing the recent requests from a certain user to the past behavior/requests from the same user. In case of deviation beyond a certain threshold, the request/packet is reported to the event editor and the intrusion detection module for further action.

This module's purpose is not just to track all previously known threats and attacks on the system but also to keep the attack sequence to compare and identify the same sequence occurring in the multi-cloud environment at any node or by any ad hoc or authorized user. That helps in preventing the attack and taking action against the possible anomaly.

There can be thousands of attacks and attempts to compromise the security parameters in a multi-cloud environment. The intrusion detection and prevention systems, honeypots, and other mechanisms can mitigate the risk to a certain level. Collecting the maximum data related to unauthorized/unauthenticated access to the system and services

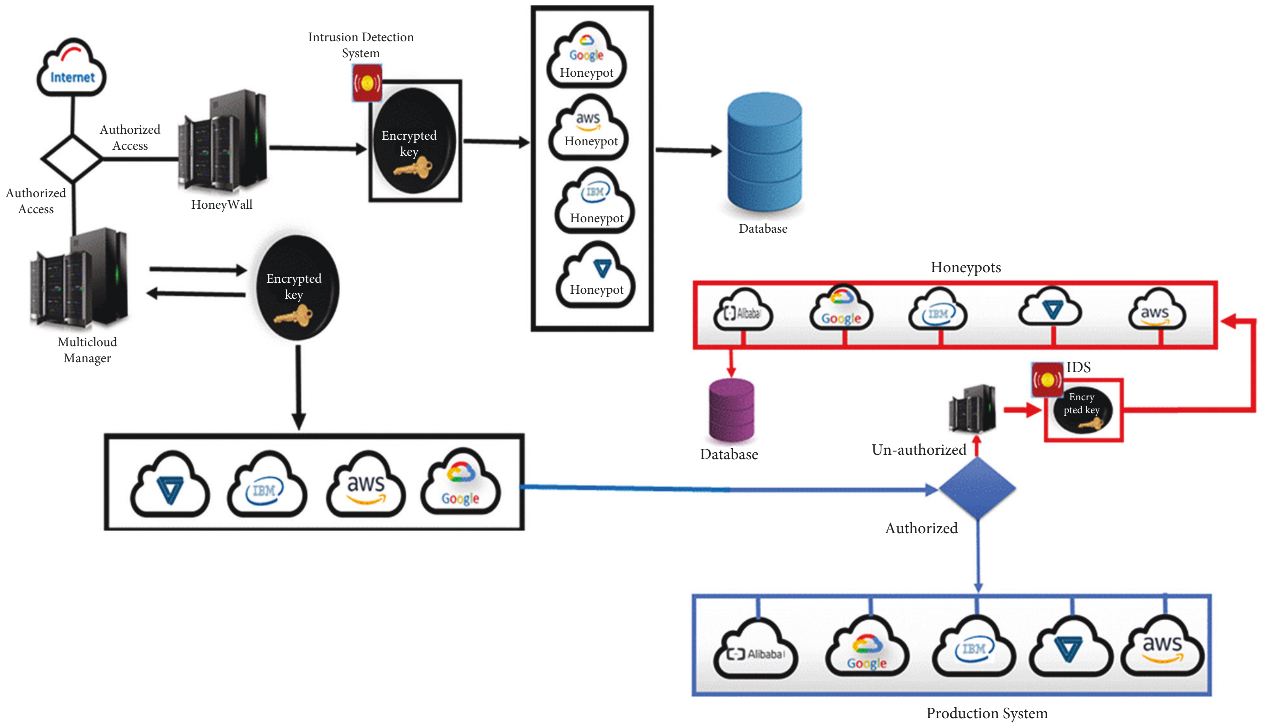

Figure 4: Proposed framework workflow.

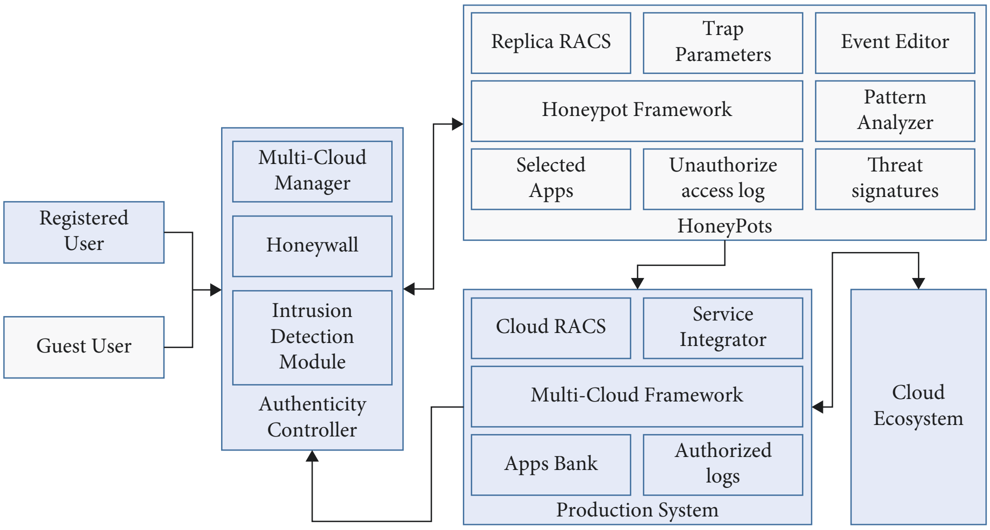

Figure 5: Modular structure of proposed framework.

is highly desirable to prepare the preventive measures and develop the honeypots according to the attacks. It provides the possible weakness or loophole in the security that can be addressed in time. As honeypots provide a decoy, a strong decoy is near to reality illusion for the intruder. The replica RACS, selected apps, and trap parameters all three modules are responsible for developing an active decoy. The purpose is to generate an alternate for the intruder that shall be similar to the real system. This module will provide the operating system, selected applications, and middleware to develop a decoy. These applications will keep changing according to the data provided by the event editor and unauthorized access logs. This module will provide the desired application and environment to the intruder to lure them into the honeypots.

As the objective or proposed framework is to provide a dynamic honeypot environment instead of a fixed decoy, trap parameter modules will collect the suitable parameters required for developing a honeypot. It will receive data from other modules to formulate a reliable honeypot environment for the intruder.

This module works as an administrator in the honeypot module; it receives and engages all other modules to develop dynamic honeypots, honeywall, and honeynets according to the changing queries and requests as observed by the other modules. The production system is the 3rd tier in the proposed framework to manage intrusion and honeypots. The more focus of this module is on the management of multi-cloud environment. The figure shows that the production system module has submodules and the cloud ecosystem. As mentioned earlier, these modules will use the meta properties of the constituent clouds and services instead of engaging the actual cloud platforms.

Similarly, here in the production system, the submodules use the meta properties-based directory of the multi-cloud constituents and service catalogue for processing and decision-making. In a multi-cloud environment, it is vital to have all the constituent cloud service providers, and the services, prices, subscriptions, and other details shall be recorded in a manner that will help in switching the service autonomously if required. This module aims to keep such information, which helps evaluate the existing multi-cloud configuration, and rank it compared to previously used configurations. This will help in developing new multi-cloud configurations.

The complexity of the multi-cloud platform is the engagement of services from different cloud service providers. The possibility of changing these services at any given time is a highly dynamic and difficult challenge. Suppose there are infrastructure services from a certain cloud provider and platform, services from another provider while the software services are engaged from a third service provider. In that case, it becomes a core requirement to develop integration among these services to provide a working environment to the end-user.

The multi-cloud platform may have same application service from different providers depending on the properties, service type, and performance. An application bank is provided which shall keep the record of application performance, usage time, and comparative applications from other cloud service providers that have been used previously. This will help provide users with the basic analytics related to the applications (typically used most extensively); therefore, if a better application by other cloud service providers in terms of performance, price, and ranking is available, the proposed framework will be able to provide a recommendation. As previously discussed that the usage pattern is critically important in the identification of an anomaly or a different behavior, therefore, all the authorized access record is maintained by this submodule to provide a usage pattern, to develop a normality threshold, and to develop the behavior of a specific user in terms of timing, request types, most requested services, etc.

The proposed framework provides three tiers of processes and functions to encapsulate the cloud ecosystem. The actual multi-cloud services, products, and structure are available in the cloud ecosystem. The constituent cloud service providers in this multi-cloud platform will have complete freedom to have their respective security and accessibility measures.

7. Simulations/Experiment and Results

Validating the proposed framework starts with identifying the most vulnerable and weak areas. Literature has provided multiple aspects related to cloud computing, multi-cloud platforms penetrating into dockers, containers, and SQL injection [26].

7.1. Simulation Setup. Targeted types of attacks are as follows:

(i) DOS: denial of service.

(ii) R2L: unauthorized access from a remote machine, e.g., guessing password.

(iii) U2R: unauthorized access to local superuser (root) privileges, e.g., various "buffer overflow" attacks.

(iv) Probing: surveillance and another probing, e.g., port scanning.

(v) Protocols for testing: HTTPS, HTTP, SMTP, DNS, POP3, BGP, IMAP, TSP, and SNMP.

Dataset used: KDD Cup 1999 dataset since 1999 has been the most wildly used dataset for evaluating anomaly detection methods. This dataset is prepared and built based on the data captured in DARPA′98 IDS evaluation program [27].

Dataset description: data files are as follows:

(i) kddcup.names: a list of features.

(ii) kddcup.data.gz: the full dataset

(iii) kddcup.data_10_percent.gz: a 10% subset.

(iv) kddcup.newtestdata_10_percent_unlabeled.gz

(v) kddcup.testdata.unlabeled.gz

(vi) kddcup.testdata.unlabeled_10_percent.gz

(vii) corrected.gz: test data with corrected labels.

(viii) training_attack_types: a list of intrusion types.

(ix) typo-correction.txt: a brief note on a typo in the dataset that has been corrected.

Consider all these factors and strategies proposed in handling security challenges due to the emergence of the multi-cloud environment. Intrusion detection systems are growing into more versatile and agile black boxes that may be designed to work with multi-cloud environments [28]. The upcoming multi-cloud computing security challenge can capture user behavior, login patterns, and routine anomalies. Intrusion detection and intrusion prevention are two subsets of these techniques and tools [29].

Univariate, multivariate, and time series models are all statistically based models. These models create two datasets that depict stochastic behavior based on network traffic activity. It is preferable to look into a little more information for comparison.

HAIL, RACS, and ICStore are three important candidates for cloud computing in general and multi-cloud settings, each with its own set of advantages and disadvantages. K.D introduced high availability and integrity layer (HAIL). It enables users to work with files across multiple servers without multiple protocols or updates. HAIL employs a proxy service to act as an identifier on behalf of the user. The proxy service/entity communicates with many cloud providers' servers and services across multiple cloud platforms. HAIL is also capable of encrypting files during aggregation to ensure file integrity.

Redundant array of cloud storage (RACS) also manages a multi-cloud environment at the storage level. The goal of RACS is to continue to locate the best cost-effective and secure resource for the end-user. It considers factors such as overhead costs, accessibility, and vendor performance. RACS uses a very identical scenario to RAID5 to administer the distributed file management system across numerous cloud services and service providers at the storage level. HAIL also uses a RAID-like scenario, although, as previously stated, the tradeoff is between multi-cloud range and versioning. The RAID5 engagement in RACS established availability, replication, and efficiency provisioning across various cloud platforms.

HAIL, as previously indicated, includes cryptography for more secure retrieval procedures and erasure-coded distributed storage. HAIL uses symmetric keys that must be kept secret by the user. In contrast, RACS and ICStore use RAID5 to distribute and engage in a multi-cloud environment across different servers or services. It is worth mentioning that these strategies are more focused on specific scenarios and services, revealing their limitations. We compared HAIL, ICStore, and RACS, and found that they are all storage-oriented. This is due to the nature of cloud computing, which is a distributed system. In the case of single cloud operations, each cloud service provider employs various technologies to assure storage security; however, this has been breached in several instances.

Although HAIL, RACS, and ICStore have given a multi-cloud solution, there is a gap for comprehensive intrusion detection or, more accurately, intrusion prevention modulation, as previously described. It is also worth noting that storage-related services are not the only ones subject to intrusion attempts. As previously stated, APIs, application-level activities, and hardware resources are all targets for intrusion. Because programs have a trust signature in their generic form, hackers can use them as a ruse to access the system as a trusted user. The most well-known examples of such intrusions are denial of service attacks, SQL injection assaults, and CAPTCHA breaking. As a result, it is highly desirable to provide.

In order to create a multi-cloud configuration, Snort employed a virtual segment made up of GCS, EC2, and Salesforce. Using our suggested methodology, the discovery and simulation of these factors have given insight into the fundamentals of intrusion and threats. The focus is on fundamental cloud activities rather than these parameters, which are valid for cloud and web apps and server-based environments. Using sniff, we have discovered such parameters confronting vulnerability at all times, especially in cloud computing.

The following parameters are selected after literature review and experimentation that most of the attacks are on conventional protocols, which is the first phase of intrusion detection. Suppose the protocol attacks and vulnerabilities are manageable with the help of the proposed framework. In that case, it is undoubtedly a great endeavor to evaluate other cloud segments to analyze vulnerabilities and risk of attacks to develop prevention and intrusion detection and provide the results for developing honeypots. The following parameters as shown in Table 1 have been selected to evaluate the proposed framework in this very first phase.

Table 1: Validation parameters.

1	HTTPS
2	HTTP
3	SMTP
4	DNS
5	POP3
6	BGP
7	IMAP
8	TSP
9	SNMP

Table 2: Dataset for phase-I.

Type	No. of attacks	Detected	Accuracy rate
HTTPS	2,955.00	2,863.00	96.89
HTTP	1,150.00	1,010.00	87.83
SMTP	995.00	895.00	89.95
DNS	587.00	480.00	81.77
POP3	490.00	485.00	98.98
BGP	175.00	163.00	93.14
IMAP	170.00	155.00	91.18
TSP	152.00	130.00	85.53
SNMP	155.00	127.00	81.94

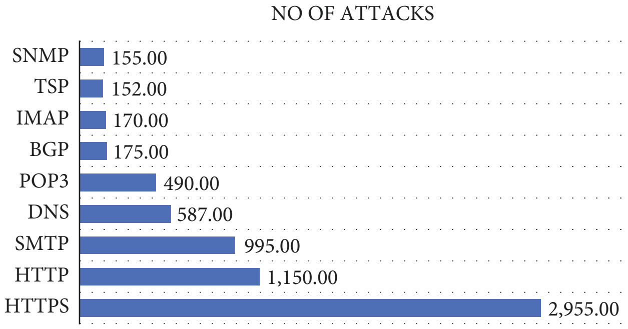

Figure 6: Number of attacks on protocols.

7.2. Validation Phase-I—Without Honeypots. The initial phase has applied two process cycles on all selected parameters. The first cycle is to calculate the total number of attacks in a 24-hour duration. It is visible in the data table that most of the threats were against HTTPS protocol, reaching 2995 while the lowest number among top five protocols is 490 in 24 hours, as shown in Table 2. It is also important to note that the phase-I for the proposed framework is executed without honeypots.

Similarly, the other protocols face a substantial number of attacks up to POP3 protocol, reaching almost 500. The ratio of attacks on the remaining protocols is comparatively lesser than the top five protocols ranging from 155 to 175 as shown in Figure 6. Although it is important to note that these results are based on a 24-hour cycle, it may change in a heavier figure if data are engaged for a longer instance. The second cycle in phase-I is activated to capture the number of

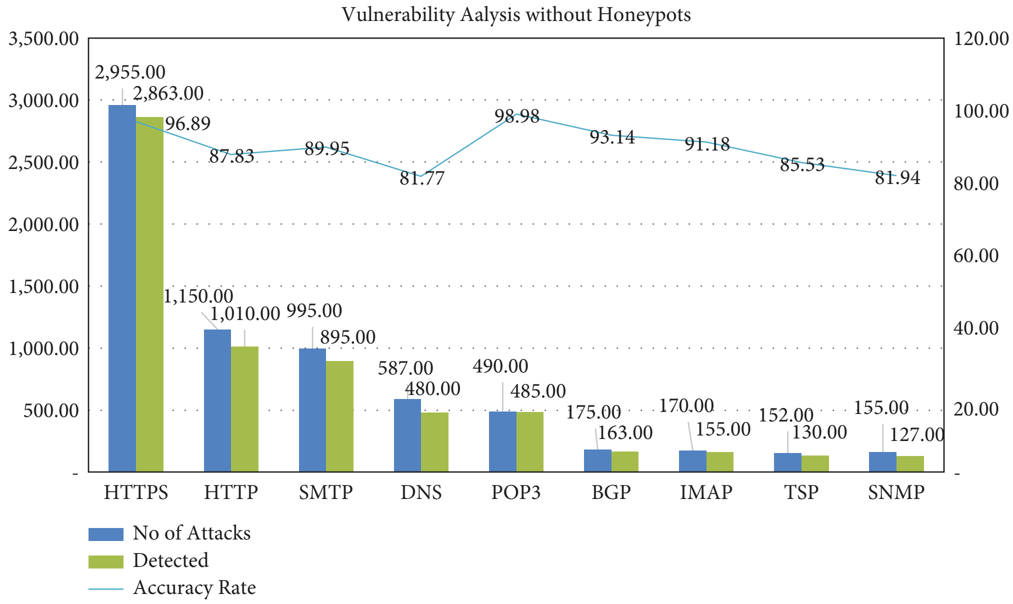

Figure 7: Vulnerability analysis without honeypots.

attacks detected without engaging honeypots on the same protocols. The result shows the most vital vulnerability detection is found in the protocol POP3, in which 485 attacks are detected successfully out of 490 attacks, resulting in a 98.98% success rate as shown in Figure 7. The DNS protocol is weakest, which faces 587 attacks and identifies 480 successfully, turning 81.77%, the lowest. Both cycles were executed and observed for another 24 hours, resulting in the same output with minor variation. The number of attacks and detection without the engagement of honeypots ranges from 81.77 to 98.98%. The simulation is performed with these protocols engaging different cloud platforms.

The top five protocols have 500+ attacks in a day, and these attacks are changing in nature and level with every passing day. Therefore, the vulnerability level is high and demands a carefully crafted framework to capture the missing attackers and prepare the system for unknown and high volume attacks.

7.3. Validation Phase-II—With Honeypots. The second validation phase includes the same protocols and parameters in similar tenure with honeypots as mentioned in the proposed framework. The dataset shows the number of attacks ranging from 148 to 2765. The maximum attacks as observed before are on HTTPS protocol, the previous accurate rate for this protocol without honeypots is 96.89, while after the engagement of honeypots, the accuracy is improved for this protocol at 98.99%. The weakest link in this scenario is 95.15 of the IMAP protocol that faced 165 attacks and detected 157 successfully. The status of this protocol without honeypots shows 91.18% accuracy, so in comparison, the engagement of honeypots has improved the detection rate as shown in Table 3.

The number of attacks shown in the data again endorses the initial observation that the most vulnerable protocols are top five without changing the hierarchy, as shown in Figure 8. The SNMP protocol is the least attacked protocol. The proposed framework's engagement with honeypots significantly impacts detecting anomalies, attacks, and vulnerabilities on the selected parameters. The summary results show an overall improvement in attack detection capability in phase-II, using the proposed framework with honeypots. It shows the improvements in the accuracy of each protocol's vulnerability analysis. It is also interesting to note that the accuracy of POP3 protocol is decreased after the engagement of

Table 3: Dataset for phase-II.

Type	No. of attacks	Detected	Accuracy rate
HTTPS	2,765.00	2,737.00	98.99
HTTP	893.00	883.00	98.88
SMTP	551.00	544.00	98.73
DNS	289.00	285.00	98.62
POP3	249.00	242.00	97.19
BGP	171.00	163.00	95.32
IMAP	165.00	157.00	95.15
TSP	149.00	146.00	97.99
SNMP	148.00	146.00	98.65

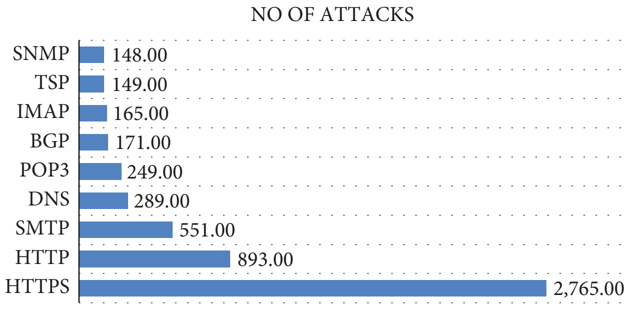

Figure 8: No. of attacks (with honeypot).

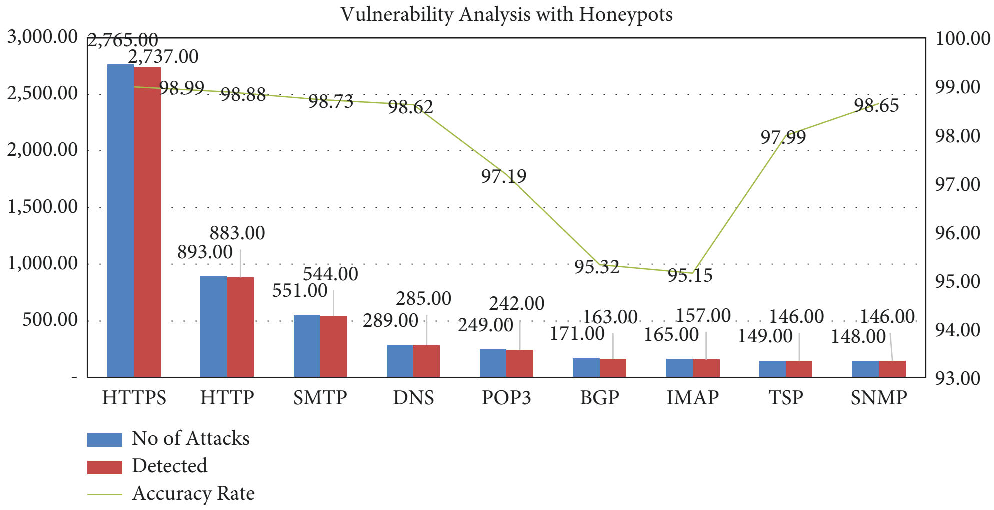

Figure 9: Vulnerability analysis with proposed framework.

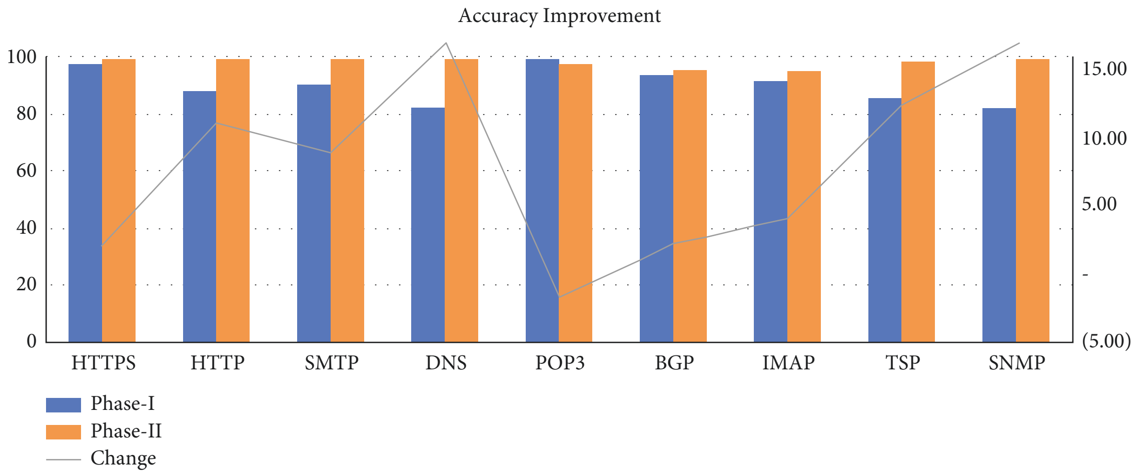

Figure 10: Proposed framework accuracy improvement.

Table 4: Proposed framework accuracy improvement.

Type	Phase-I	Phase-II	Change
HTTPS	96.88663	98.99	2.10
HTTP	87.82609	98.88	11.05
SMTP	89.94975	98.73	8.78
DNS	81.77172	98.62	16.85
POP3	98.97959	97.19	(1.79)
BGP	93.14286	95.32	2.18
IMAP	91.17647	95.15	3.97
TSP	85.52632	97.99	12.46
SNMP	81.93548	98.65	16.71

honeypots. In phase-I, the accuracy of this protocol is 98.98%, which was also the highest in the previous analysis. In comparison, in the current phase the accuracy of this specific protocol is 97.19% as shown in Figure 9.

7.4. Proposed Framework Results. The validation of the proposed framework shows a significant improvement in the analysis results with and without the engagement of honeypots. The highest difference is achieved on DNS attacks, where it showed 81.77 accuracies without honeypots but after the engagement of the proposed framework, the accuracy has increased on 98.62, which creates a difference of 16.85. The following table shows the accuracy improvement on each protocol.

Similar results depict the difference of 11.05 in the case of HTTP that was previously having an accuracy of 87.82%, while after the incorporation of the proposed framework, the accuracy result has improved to 98.88%. These results make it evident that engagement of honeypots or making a security mechanism positively impacts the system's resilience against known and unknown attacks, especially in the multi-cloud-like complex environment shown in Table 4.

The accuracy improvement graph presents the phase-I and phase-II results and the improvement due to the engagement of honeypots in intrusion detection and prevention with the help of the proposed framework, as shown in Figure 10.

Different intrusion detection algorithms are tested on the proposed framework, including SVM, LR, RF, GNB, and DT to evaluate the robustness of the framework with training and testing using KDD12 dataset. Intrusion detection based on anomaly and signature is both covered in the proposal. System can identify the threat and update its repository for newly identified threats. Therefore, the proposed framework is ideally suitable for a complex environment like multi-cloud.

8. Conclusion

The proposed model has taken a multi-cloud platform as the target environment and engaged the conception of honeypots to develop a security mechanism that not only prevents attackers but also captures the properties of the attack, maintains a database for the attack sequences, develops an understanding of the user behavior, and requests to detect an anomaly, dishonest behavior, and suspicious requests. With all these detections and preventive measures, honeypots collect trap parameters from the packets to develop a decoy to avoid attacks on the digital assets. A modular approach has been adopted to develop the framework containing modules and functional sub-modules. The framework can manage multi-cloud activities and the honeypot module, which can develop dynamic honeynets to divert intruders into a real-like environment.

The validation process for the proposed framework is split into two main phases; one phase is the vulnerability attacks and detection of such attacks without the engagement of the honeypots module in the proposed framework. The results provided by phase-I are the threshold value for phase-II. In the second phase, the same parameters were used under the same environment with the honeypot module in the proposed framework. The results for the same parameters show significant improvement in detection accuracy rate. The difference between phase-II and I is the key indicator to expose the potential of honeypots in a carefully crafted mechanism for intrusion detection and prevention in the multi-cloud environment for a secure and resilient system.

Data Availability

The data used in this paper can be requested from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest regarding the publication of this work.

Authors' Contributions

All authors contributed.

References

[1] P. Priyadarshinee, R. D. Raut, M. K. Jha, and B. B. Gardas, "Understanding and predicting the determinants of cloud computing adoption: a two staged hybrid SEM - neural networks approach," *Computers in Human Behavior*, vol. 76, pp. 341–362, 2017.

[2] N. Tabassum, A. Ditta, T. Alyas et al., "Prediction of cloud ranking in a hyperconverged cloud ecosystem using machine learning," *Computers, Materials & Continua*, vol. 67, no. 3, pp. 3129–3141, 2021.

[3] A. E. Gonzalez and E. Arzuaga, "HerdMonitor: monitoring live migrating containers in cloud environments," in *Proceedings of the 2020 IEEE International Conference on Big Data*, pp. 2180–2189, Atlanta, GA, USA, December 2020.

[4] T. Alyas, N. Tabassum, M. Waseem Iqbal, A. S. Alshahrani, A. Alghamdi, and S. Khuram Shahzad, "Resource based automatic calibration system (rbacs) using kubernetes framework," *Intelligent Automation & Soft Computing*, vol. 35, no. 1, pp. 1165–1179, 2023.

[5] https://www.informatica.com/in/products/cloud-integration/connectivity/getting-started.html.

[6] L. Heilig, E. Lalla-Ruiz, and S. Voß, "Modeling and solving cloud service purchasing in multi-cloud environments," *Expert Systems with Applications*, vol. 147, Article ID 113165, 2019.

[7] M. B. Erdem, A. Kiraz, H. Eski, Ö. Çiftçi, and C. Kubat, "A conceptual framework for cloud-based integration of Virtual laboratories as a multi-agent system approach," *Computers & Industrial Engineering*, vol. 102, pp. 452–457, 2016.

[8] M. Alaluna, E. Vial, N. Neves, and F. M. V. Ramos, "Secure multi-cloud network virtualization," *Computer Networks*, vol. 161, pp. 45–60, 2019.

[9] M. Lattuada, E. Barbierato, E. Gianniti, and D. Ardagna, "Optimal resource allocation of cloud-based spark applications," *IEEE Transactions on Cloud Computing*, vol. 10, no. 2, pp. 1301–1316, 2022.

[10] J. Nazir, M. Waseem Iqbal, T. Alyas et al., "Load balancing framework for cross-region tasks in cloud computing," *Computers, Materials & Continua*, vol. 70, no. 1, pp. 1479–1490, 2022.

[11] M. I. Sarwar, M. W. Iqbal, T. Alyas et al., "Data vaults for blockchain-empowered accounting information systems," *IEEE Access*, vol. 9, Article ID 117306, 2021.

[12] T. Alyas, I. Javed, A. Namoun, A. Tufail, S. Alshmrany, and N. Tabassum, "Live migration of virtual machines using a mamdani fuzzy inference system," *Computers, Materials & Continua*, vol. 71, no. 2, pp. 3019–3033, 2022.

[13] K. B. Sundharakumar, S. Dhivya, S. Mohanavalli, and R. V. Chander, "Cloud based fuzzy healthcare system," *Procedia Computer Science*, vol. 50, pp. 143–148, 2015.

[14] S. Malik, N. Tabassum, M. Saleem, T. Alyas, M. Hamid, and U. Farooq, "Cloud-iot integration: cloud service framework for m2m communication," *Intelligent Automation & Soft Computing*, vol. 31, no. 1, pp. 471–480, 2022.

[15] D. Baig, T. Alyas, M. Hamid et al., "Bit rate reduction in cloud gaming using object detection technique," *Computers, Materials & Continua*, vol. 68, no. 3, pp. 3653–3669, 2021.

[16] M. Shifrin, R. Mitrany, E. Biton, and O. Gurewitz, "VM scaling and load balancing via cost optimal MDP solution," *IEEE Transactions on Cloud Computing*, vol. 7161, p. 1, 2020.

[17] R. Tahir, A. Raza, M. Naqvi, F. Zaffar, and M. Caesar, "An anomaly detection fabric for clouds based on collaborative VM communities," in *Proceedings of the IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing*, pp. 431–441, Madrid, Spain, May 2017.

[18] A. Praseed and P. S. Thilagam, "DDoS attacks at the application layer: challenges and research perspectives for safeguarding web applications," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 1, pp. 661–685, 2019.

[19] N. Tabassum, M. Khan, S. Abbas, T. Alyas, A. Athar, and A. Khan, "Intelligent reliability management in hyper- convergence cloud infrastructure using fuzzy inference system," *EAI Endorsed Transactions*, vol. 6, pp. 1–12, 2020.

[20] A. Mondal and R. T. Goswami, "Enhanced Honeypot cryptographic scheme and privacy preservation for an effective prediction in cloud security," *Microprocessors and Microsystems*, vol. 81, Article ID 103719, 2021.

[21] I. A. Valdovinos, J. A. Pérez-Díaz, K. K. R. Choo, and J. F. Botero, "Emerging DDoS attack detection and mitigation strategies in software-defined networks: taxonomy, challenges and future directions," *Journal of Network and Computer Applications*, vol. 187, Article ID 103093, 2021.

[22] N. Tabassum, T. Alyas, M. Hamid, M. Saleem, and S. Malik, "Hyper-Convergence storage framework for EcoCloud correlates," *Computers, Materials & Continua*, vol. 70, no. 1, pp. 1573–1584, 2022.

[23] A. K. Soliman, C. Salama, and H. K. Mohamed, "Detecting DNS reflection amplification DDoS attack originating from the cloud," in *Proceedings of the 13th International Conference on Computer Engineering and Systems (ICCES)*, pp. 145–150, Cairo, Egypt, December 2019.

[24] W. Zhang, B. Zhang, Y. Zhou, H. He, and Z. Ding, "An IoT honeynet based on multiport honeypots for capturing IoT attacks," *IEEE Internet of Things Journal*, vol. 7, no. 5, pp. 3991–3999, 2020.

[25] L. Shi, Y. Li, T. Liu, J. Liu, B. Shan, and H. Chen, "Dynamic distributed honeypot based on blockchain," *IEEE Access*, vol. 7, Article ID 72234, 2019.

[26] M. A. Khan, S. Saqib, T. Alyas et al., "Effective demand forecasting model using business intelligence empowered with machine learning," *IEEE Access*, vol. 8, pp. 116013–116023, 2020.

[27] "Intrusion Detection System Using Machine Learning Algorithms," 2022, https://www.geeksforgeeks.org/intrusion-detection-system-using-machine-learning-algorithms/?cv=1.

[28] A. A. Khan, M. Zakarya, I. U. Rahman, R. Khan, and R. Buyya, "HeporCloud: an energy and performance efficient resource orchestrator for hybrid heterogeneous cloud computing environments," *Journal of Network and Computer Applications*, vol. 173, Article ID 102869, 2021.

[29] A. Nasir, T. Alyas, M. Asif, and M. N. Akhtar, "Reliability management framework and recommender system for hyper-converged infrastructured data centers," in *Proceedings of the 3rd International Conference on Computing, Mathematics and Engineering Technologies*, pp. 1–6, Sukkur, Pakistan, January 2020.