International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 4s
DOI: https://doi.org/10.17762/ijritcc.v11i4s.6312
***Article Received**: 28 December 2022 **Revised**: 30 January 2023 **Accepted**: 06 February 2023*

Integrating the EGC, EF, and ECS Trio Approaches to Ensure Security and Load Balancing in the Cloud

Abin T Abraham[1], Dr. E. J. Thomson Fredrik[2]
[1]Research Scholar
Department of Computer Science
Karpagam Academy of Higher Education
Coimbatore
Email :abysid@gmail.com
[2]Professor
Department of Computer Applications
Karpagam Academy of Higher Education
Coimbatore
Email : thomson500@gmail.com

Abstract- According to data protection studies, "Distributed Denial-of-Service (DDoS)" threats have cost governments and businesses throughout the globe a large number of financial resources. Despite this, the existing practices fall short of the standards set by "Cloud Computing (CC)" monitoring technology. They ignore the "Intrusion Detection Systems (IDS)" techniques, which take advantage of the CC's multiple tenants and elasticity qualities, and also the hardware limitations. Attackers are finding increasing ways to effectively exploit them because of their rising complexity. DDoS assaults of this scale have never been observed online before 2018. As online services get more popular, so does the amount of DDoS assaults and malevolent hackers leading to terrible. Numerous IDS for DDoS are already in place to address this problem. One of the most challenging aspects of virtualization is establishing a "Trust Model (TM)" between the many "Virtual Machines (VMs)". The lack of a standard formulation for generating a TM would be the primary reason. As a consequence, the integrity of every VM might not have been recognized by an independent trust, which might lead to a decrease in trust value. In this research for TM creation, "Enhanced Graph Based Clustering (EGC)" is proposed, while "Enhanced Fuzzy (EF)" is used for detecting attacks, and the "Enhanced Cuckoo Search (ECS)" method is used to find the ideal "Load Balancing (LB)" distribution. By creating a new TM, the proposed (EGC-EF-ECS) system strengthens trust value. To expand the CC model's stability, it optimizes attacker recognition percentage and makes better use of resources by restricting each VM's processing, bandwidth, and storage requirements. The proposed EGC-EF-ECS outperformed the previously used BPA-SAB, and DCRI-RI approaches in terms of the "Intrusion-Detection-Rate (IDR)", "Load-Balancing-Efficiency (LBE)", and "Data-Accessing-Time (DAT)" evaluation metrics.

Keywords: Cloud Security, Load Balancing, Trust Model, Clustering, Fuzzy, Cuckoo Search

I. INTRODUCTION

CC had already rapidly advanced and contributed many notable results over the last several years, rendering itself a major achievement. It became a popular distributed network framework owing to its cost-effectiveness, terrific efficiency, on-demand accessibility, authorization, and many other elegant characteristics [1]. The CC-related technologies are considered to constitute the IT industry's "next-gen" framework. When compared to traditional platforms, it permits the customer to relocate personal applications and data programs to the web. Customers may deploy and run their customized applications with the help of the extensive framework provided by CC, which includes servers, memory, channels, and much other computing infrastructure [2].

The customer is responsible for managing and controlling the data, programs, and possibly other features they choose, rather than the CC infrastructure itself. In a business context, the creation of a prospective and privileged approach for the customers is guaranteed by the CC's information exchange system, which gives a flexible means of storing a huge quantity of data that might vary [3]. Remote storage of data has numerous benefits, but it comes with the risk such sensitive data may be compromised, disclosed, or indeed recreated [4]. Verifying customer credentials, malicious insiders or loopholes, spyware assaults, outer connection to the company's servers, a shortage of competent authorities, and other concerns are among the most critical [5].

Consequently, this must recognize the fact that privacy considerations seem to be highly essential in the CC frameworks since privacy is the highest-valued aspect of computing. Considering that the CC approach would have to maintain confidential user information across both "Cloud Users (CU)" as well as "Cloud Servers (CS)", access controls and authorization are important in CC [6]. Due to certain threats, the "Data Owners (DO)" need robust safeguards and

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 4s
DOI: https://doi.org/10.17762/ijritcc.v11i4s.6312
***Article Received**: 28 December 2022 **Revised**: 30 January 2023 **Accepted**: 06 February 2023*

efficient security procedures from the CS. Such confidentiality could be provided by using any number of effective methods and practical encryption methods [7]. Our purpose was to maintain the CC safe and eliminate those security flaws such that CS could be used with complete trust.

An LB would manage networking or application activity over a CS cluster. LB improves both the accessibility and reactivity of an application. An LB is located in the middle, in between CU and the CS, and processes all the applications and network information flowing in before distributing it to the many CSs in the backbone [8]. Throughspreading service requests across multiple CSs, an LB reduces the load on every given CS thus preventing each CS to turn into a bottleneck. As a result, the application's general accessibility and reactivity are improved. LB provides the most convenient method for developing an application's CS framework [9].

Whenever the application load increases, more CSs could be assigned to the pool of resources, as well as the LB would begin sending traffic to such extra CS immediately. Whenever users want to do computations more quickly and complete jobs more quickly, CUs need to leverage LB to divide the work among the resources available. It's possible that certain CS is being used at maximum capacity whereas others are idling or underused. Consequently, superior LB techniques can do more than just defend against these problems; they can also boost efficiency, keep the network stable, provide fault management, and make room for future changes [10].

Motivation and Problem Statement: Presently, the CC has unique cybersecurity and LB problems owing to its virtualized and multiple tenancies. Recent CC adoption initiatives have resulted in economic advantages for about 84% of enterprises. Nearly 66% of businesses have reported that by adopting this CC, they have reduced their energy consumption and resources, which is a significant step toward sustainable development. For CUs to be able to make full use of CC assets without risking the disclosure of personal data and confidential material to third parties like CS, a unique protection mechanism is required. Moreover, DOs could at whatever time inspect security criteria like system stability, allowing them absolute authority over the confidentiality and safety of its data storage. By improving the efficient utilization of existing hardware and software, appropriate LB could boost system efficiency while simultaneously reducing resource consumption. Further, it facilitates the advent of failing over, permits scalability, eliminates obstacles and over-resourcing, shortens processing times, and so on. However, CUs often look for further with less in terms of response times and budget. The implementation of standardized LB techniques with testing methods that might reduce resource expenditures while escalating performance is difficult due to the reality that companies are always coming up with novel approaches to the problem. Companies may become sustainable and also save funds by reducing their resource consumption with effective LB.

Paper Contribution: We establish a novel TM, identify threats, and develop an optimal LB mechanism for the CC setting is the primary contribution of this research.With the proposed EGC-EF-ECS architecture, we could improve resource usage, decrease reaction times, and promote user experience.Starting with the EGC method, the research framework builds a VM security framework for every hypervisor.The EGC method recommends integrating "ObjectiveSource (OS)", and "SubjectiveSource (SS)"to enhance the dependability of confidence ratings, and while doing so, this finds the relationship between the hypervisor and also the host VM.After VMs successfully gained confidence from the OS and SS, a further process is to compile data from all various sources into a single, conclusive "TrustScore (TS)".The EF classification was introduced to attack detection about the same period.ECS-spanning VMs may employ the best LB allocation technique, according to what the hypervisor needs.

Paper Organization: Section 2 provides a list of references of relevant literature, Section 3 provides a detailed explanation of the methodologies used in developing the proposed framework, Section 4 examines the findings acquired using the both proposed and existing frameworks also making comparisons between them, and Section 5 comes to the conclusion by discussing the article's potential applications moving forward.

II. RELATED WORKS

The researchers of [11] developed a concept wherein tokens contained in the initial "Transmission Control Protocol (TCP)" packets are employed to confirm the claimed identity of the CU. It is being leveraged to establish whether this CC architecture would resist DDoS assaults, identification counterfeiting, and CC fingerprints across a variety of contexts, including enterprise-grade servers, CC storage centers, and a CC spanning a university. For widely dispersed CC networking in universities, this method of initial packet authenticating utilizing tokens has been enhanced.

The Client-server paradigm had been presented in [12] by the researchers for validating the credibility of CSs acting as voluntary CSs inside a zero-trust CC network. At first, the software doesn't quite have much confidence mostly in CSs. To ensure that the highest-quality reliable CSs are used for certain activities, they devised a system depending on their behavior that may dynamically adjust the workload and lifespan of each CS. CSs with particularly lower TS scores are

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 4s
DOI: https://doi.org/10.17762/ijritcc.v11i4s.6312
***Article Received**: 28 December 2022 **Revised**: 30 January 2023 **Accepted**: 06 February 2023*

put on a block list or otherwise assigned fewer or even no duties as a result. Their research of the CS's whole lifespan led to the creation of this TS, which takes into account the CS's behavior, performance, and accessibility, amongst many other things.

Round-Robin is a method introduced by researchers in [13] for equitably distributing the running processes. 3 distinct scenarios have been used insisting CC analyzer modeling program evaluates the "Throttled LB (TLB)" techniques, and the results showed here that the TLB approach had the best performance. Both reaction speed and database server computation improved using the TLB technique. Throughout this assessment, the potential for a resource management evaluation was disregarded. When researching techniques, resource consumption is a common metric to analyze.

The researchers of [14] provide a comprehensive overview of nontraditional hybrid methods. Specifically, the research elucidates that such CC analytical technique has been the most often utilized method across multiple research mostly in the LB category. The outcomes of the research study don't specify whether a heterogeneous or homogeneous VMs setup was employed.

A "Balanced Throttled" technique was proposed by the researchers in [15]. The effectiveness of the suggested approach had been compared to that of the "Round-Robin" approach, the TLB approach, as well as the proactive VM supervision optimization method using the CC analyzer program. This research has improved the TLB method's reaction time. Nonetheless, the research fails to detail the exploratory settings that were employed.

III. METHODOLOGIES

3.1 BPA-SAB

To solve this issue, the "Binomial Protection based Authentication with a Stochastic Agent based Load Balancing (BPA-SAB)" has been developed and employed earlier [16]. To ensure users' privacy when using the CC, the BPA-SAB approach have being implemented. The CU sends its requirements to the CS initially. The suggested BPA-SAB method then executes the BPA to successfully recognize the intruder assaults, generating superior IDR outcomes. This authorization ensures that solely approved CUs may access the CS's content inside the CC, therefore boosting personal information PP. As a result, information kept in the CC is more secure. Next, the SAB is activated in sequence to carry out the LB required to get entry to the CC's contents. That's why the previously proposed BPA-SAB method, including its enhanced data PP as well as LB, is so useful for retrieving records on the CC.

3.2 DCRI-RI

By efficiently identifying prohibited users (i.e. attackers), the "Dynamic Certificateless Random Identity with Rank Indexing (DCRI-RI)" approach was introduced earlier that secures data transfer inside the CC [17]. Before the CUs may share the data stored mostly on the CC end, they must enroll the "Identity (ID)" in the CU part. The "user-id" and "password" for the CU are generated by the CS once enrollment has been completed. DCRI has been used to create the "Dynamic-Random key" that is utilized throughout the encryption operation. The source information is encrypted to increase speed, followed by the cipher-text forwarded to the CUs. The CU then decrypts the encrypted message to reveal the original message. The next step is a signature-key validation to ensure that only approved CUs have access to the information. The LBE would then be optimized by distributing the workload over many cloud nodes using a total weighted indexed RI method.

3.3 EGC-EF-ECS

The overall flow chart of the proposed EGC-EF-ECS model is shown in Figure 1.

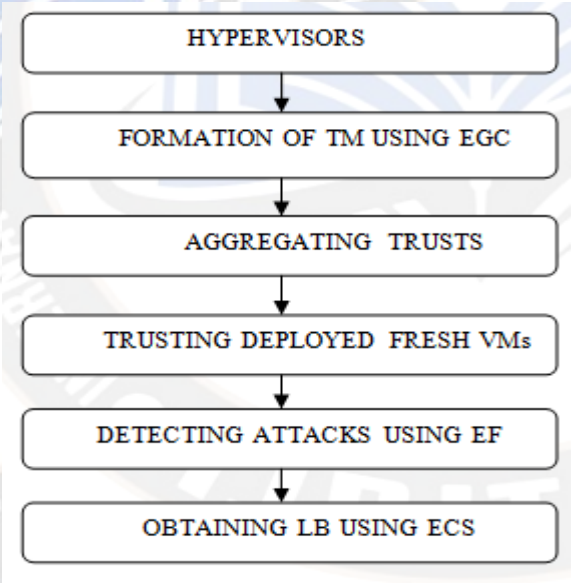

Figure 1:Proposed EGS-EF-ECS Methodology Flow

3.3.1 Formation of TM using EGC

(i) Model for CC

The VMs develops a CC model with the help of the hypervisor and its resources. Accordingly, a finite set of "Hypervisors [HY=(hy_1...hy_n)]", is considered, where each "Hypervisor [$hy_i \in$ HY]" hosts a set of "VMs [VM_i={vm_1,...,vm_l}, i.e. VM € Vm_i]". A "Client Set [CL=$c1_1$...cl_m]" has been the legitimate owner of each VM on "hy_i". Between both the CC machine as well as the VMs is

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 4s
DOI: https://doi.org/10.17762/ijritcc.v11i4s.6312
***Article Received**: 28 December 2022 **Revised**: 30 January 2023 **Accepted**: 06 February 2023*

often a "Software Agent" known as a "Hypervisor [$hy_i \in HY$]". The primary goal is to create a simulated pool of "Hardware Resources [$I = I_1, I_2, ...I_n$]" and also to schedule VM unprecedented admin rights so that numerous VMs may operate in parallel. This article aims to explore how the hypervisor could build trust well with guests' VMs. However, the existing system does not follow any specific representation for creating a TM. This research proposes an EGCfor the formation of TMs and tries to build a trustworthy network compared to the existing one.

(ii) VMs Trust Building

The proposed EGC algorithm automatically forms the TM based on the computation of weight values for each vertex of a hypervisor. This uses a connectivity based clustering tendencies validating metric to determine on its own the number of clusters being created. Consider the graph "G = (V, E)", where "V" represents "VMs", and "E" represents "Edges". A value of "l" indicates that there are "l" VMs in the set "G". A pair "(i,j)" represents an edge, where both "i" and "j" are VMs in "V" respectively. The VMs throughout this research are simulated mostly by "Natural Numbers (1 to *l*)".

Let's assume the matrix representing the connections between the VMs in the graph "G" is the adjacency-matrix"$A= [ad_{ij}]_{l\times l}$". The adjacency-matrix is a set of binary numbers, each of which represents the trust level among 2 VMs in terms of the underlying OS and SS. Therefore, "$[ad_{ij}= 1]$"while "VMs [i and j]" were contiguous, i.e., when "VM[i(VM)]" links to "VM[j(VM)]", or else "$ad_{ij}=0$".

Assume "$WE = [we_{ij}]_{l\times l}$" represents the weighted matrix for edges of a graph "G" with edges that are weighted. The value of the "*weij* "entry in this matrix "WE" represents the edge's weight from node "VM [i(VM)]" to node "VM [j(VM)]". When "$we_{ij}= 0$", it means that neither edge connects "VMs [i and j]". By counting the number of neighboring samples, one could determine the degree of "VM [i(VM),(deg_i)]" in a weighted graph as per Equation (1). The adjacency of the VM is computed using the "Inter Quartile Range (IQR)".

$$deg_i = \sum_{j=1}^{l} ad_{ij}$$

Eq→1

The "Clustering Coefficient" has been a metric used to assess the degree to which a graph exhibits clustering behavior. It relies on studying cycles of 3 VMs revolving around a VM representing "VM i(VM)". Regarding unweighted graphs, the accompanying Equation (2) could be used as a measurement.

$$C_i = \frac{2\sum_{j=1}^{l-1}\sum_{k=j+1}^{l} ad_{ij}\, ad_{jk}\, ad_{ik}}{deg_i(deg_i - 1)}$$

Eq→2

The "Degree [deg_i]" value represents the overall count of neighbors for "VM i(VM)". The quantity of edges that might potentially appear here between vertices inside the VMs' local graph is determined by the denominator. To determine the degree to which "VM [i]" neighbors are linked to one another, we use this metric. To analyze a partition's propensity to cluster, it's also recommended that a strong validating metric be applied to this weighted "Clustering Coefficient". Equation (3) is taken into account for this objective.

$$dis_i = \sum_{j=1}^{l} ad_{ij}\, z_{ij}$$

Eq→3

The value of "z_{ij}" is assumed to be "1" since it is a binary-variable (When all of the VMs share identical SS as well as OS trust credentials). Whereas if "VMs [i and j]" are members of the identical cluster, else "0" (When the trusted credentials of the SS and OS vary from VM to VM). Take into consideration that "dis_i" is the actual population of neighboring VMs that are part of the same cluster as "VM [i]". The accompanying Equation (4) proposes a "Clustering Coefficient" for a "VM I generated from a partition "π" from a weighted graph.

$$Cc_i(\pi) = \frac{2\sum_{j=1}^{l-1}\sum_{k=j+1}^{l}(we_{ij}we_{jk}we_{ik})^{1/s}}{deg_i(deg_i - 1)} y_{ijk}$$

Eq→4

Whereas if " VMs [i, j,and k]" all belong to the identical cluster, then "y_{ijk}" does have the value"1", else it has the value "0".Following each round of expanding the graph VMs, a crossover procedure is performed to enhance the partition identified during the preceding stage. It does this by undoing the coarsening procedure, which would be repeated there until the initial graph form is recovered.

Throughout the coarsening stage, the graph's edges are matched one by one. It can be performed by using crossover by exchanging the current position of the VM in the

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 4s
DOI: https://doi.org/10.17762/ijritcc.v11i4s.6312
***Article Received**: 28 December 2022 **Revised**: 30 January 2023 **Accepted**: 06 February 2023*

trust model. The above procedure is expected to build a more optimistic cloud based environment between VMs using this proposed EGC.

(iii) Calculation of OS and SS Trust

The hypervisor determines the levels of OS trust through self-monitoring while generating the improved TM by using the EGC methodology. The hypervisor subsequently gathers suggestions based on the VMs' historical behavior in an attempt to enhance the integrity of the TS and uses the IQR statistical metric to detect any instances of anomalous consumption. This concept is known as "SS trust". As a result, the detection rate is enhanced in a dynamic CC setting. The suggestions aren't just dependent on DDoS associated information but on the VMs' entire behavior. It's developed from "ibf_{hyvm}" and "fbf_{hyvm}" functions of confidence, which are predicated on the OS as well as SS trust assumptions. Thereafter, freshly installed VMs are manually trusted, and trust aggregating is carried out.

3.3.2 Aggregating Trusts

For VMs which have already earned OS and SS confidence, the following process would be to compile data from many sources and provide a conclusive overall TS.It works on uncertainty through the distribution of probability. A VM with a problem ambiguity is considered.That VM's uncertainty could be shown by referring to the "Probability Distribution" represented by the division which comes before it.The overall TS of this step is increased by considering the new TM formulation using the EGC algorithm. The EGC TM algorithm developed as a graph increases the value of the TS. If one of the attackers lacks a TS, it becomes very easy to find a modified TS and is easily documented from a graph model.

3.3.3 TRUSTING FRESH DEPLOYED VMs

Freshly launched VMs within CC offer a significant challenge to the hypothesized trust process because of the need to establish trusting connections with them. Whenever a hypervisor develops confidence together in a fresh VM, this should approach other VMs including hypervisors to accept it. Throughout this scenario, the initial level of trust across all freshly launched VMs would've been determined to be a number halfway between complete confidence and complete distrust as "1/2 (i.e., $\gamma = ½$)". Consequently, following the requester's instructions to boost their self-assurance. The value of the VMs under management by the hypervisor "hy", the "W(vm)", is directly proportional to the hypervisor's level of trust in the VMs' privacy.

3.3.4 DETECTING ATTACKS USING EF

This study improved upon the standard fuzzy by integrating it along with "Neural Networks (NNs)" for classifying attacks, which may include either a solitary hidden VM layer or several layers. It isn't merely the weights of the connections between the hidden VM's inputs which make up its parameters, those weights remain fixed. Those VMs that are hidden could be generated at randomness and then left unchanged, either by getting predicted by chance while undergoing nonlinear alterations or by being adopted unchanged from their parents. Generally, the outcome weights in hidden VMs have learned alone in one cycle, which is effectively a generalized learning procedure. The incoming weights are typically chosen at randomness, whereas the outcome weights as well as hidden-layer variables have been computed rationally through the "Least Square" approach. The proposed EF classifier's attack identification conditional probability is determined by Equation (5).

$$\hat{t} = sign\big(f(x)\big) = sign(h(x)\beta)$$

Eq→5

The outcome of the weighted matrix between both the hidden-layer and the output-layer EF is denoted as "β", while "$\hat{t}$" has been the anticipated attacker labeling for incoming CUs data "x", the "h(x)" has been the hidden-layer outcome equal here to inputs to an EF's "x", and "x" has been the input to the CUs.

3.3.5 OBTAINING LB USING ECS

To optimize attack detection and prevention under constrained resource limits, an ECS technique is developed for optimum load management.This includes a clear understanding of the traditional CS and the necessary changes to be made to it. Here ECS is the optimal load distribution detection strategy which is discussed as follows:

(i) General CS algorithm for optimal load detection

The Cuckoo is also an example of a Meta-heuristic approach, that offers several benefits over traditional approaches, including being easier to implement and requiring fewer tuning factors. Conversely, it possesses a poor generalization ability and is known to collapse to local optimum solutions quite frequently. The following are the three rules that make up the general CS:

Rule (i): There is only one egg laid once at a time by each cuckoo, and they all get dumped into different nests.

Rule (ii): The healthiest nests only with the highest-quality eggs would be passed on to the following generation.

Rule (iii): There is a certain number of nesting sites accessible, as well as the cuckoo's egg, has a certain chance of being found mostly by the host bird.

In the first of the aforementioned 3 rules, a randomized value is used to determine where the nest will be built.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 4s
DOI: https://doi.org/10.17762/ijritcc.v11i4s.6312
***Article Received**: 28 December 2022 **Revised**: 30 January 2023 **Accepted**: 06 February 2023*

However, with rule 1, those nests' placements are still similar and, occasionally, they won't be evenly spread inside a certain region. This leads to iterative computations and an increased likelihood of settling on a sub-optimal (locally optimum solution). With "Harmony Search(HS)", a novel enhancement process has been developed to address this problem.

(ii) ECS optimization process

It is widely acknowledged that the ECS method using HS has always been a meta-heuristic population-optimizing approach.

New initialization is conducted by using the following 3 new rules:

Rule (i): Consideration of memory

Rule (ii): Adjustment of the pitch.

Rule (iii): Selection based on random.

In this proposed work, the ECS algorithm is used to distribute loads optimally, taking into account the functional aspects of VMs such as CPU usage, memory consumption, network bandwidth utilization, and time-based aspects, namely execution time and task service time throughout addition to IQR and VM value and statistical measures. This is analogous and is derived from the mechanism by which musicians discover new features for the optimum distribution of loads through a cooperative improvisation method.

For given iterations in this research (100), the range of values for "IQR and VM" is equivalent to the range of values for "CPU", "Memory", and "Network Bandwidth" usage inside a VM. The configuration that exists at a specified instant is related to the "Solution Vector (IQR)", also referred to as improvising, while the "CPU", "Memory", and "Network Bandwidth" consumption of the VMs are related to the "Solution Vector (IQR)".

The "Harmony Memory (HM)" has been a matrix for optimal IQR vector optimization based mostly on basic IQR architecture with VM frequency of HS. "The HM Size (HMS)" has been the maximum number of concurrently operating IQR vectors including VMs. These are some of the main algorithmic variables that must be manually modified. Every row inside the memory-state of the "IQR matrix" represents a unique hypervisor solution, and every CU's data vector and last column inside the matrix represent the "IQR" and the "VM" value, respectively.

Improvisation will be the following phase in the initiating process. The unique system depends on the aforementioned "Consideration of memory", "Adjustment of pitch", and "Selection based on random" rules. Each IQR, as well as VM value, was determined separately, and only 2 out of the 3 rules would be applied to any given hypervisor. A significant "Harmony Search (HS)" parameter that requires human intervention is the "HM Consideration Ratio (HMCR)".

The "IQR" and "VM" values are probably taken straight from among the HM's optimized hypervisors since this is normal when considering HS memory. Every IQR value gets assigned a randomized number. The selected number would be taken into account from memory if it is lower than that of the HMCR. Otherwise, the "IQR", as well as the "VM" value for that dimension, would be selected at random from the whole range of potential values. Presumably, when HM is discovered, the invented value is chosen at random from those values already existent inside the HM.

The "Pitch Adjustment Rate (PAR)" has been configured upon initialization. In particular, it guides the amount of memory used for the actual pitch adjustments that were made. Maybe a second random measurement gets created that is deemed to be lower than the initial PAR. The accompanying Equation (6) was applied to make a rough adjustment to the modified IQR value:

$$x'_{new} = x_{new} + rand(\,).FW$$

Eq→6

In this case, "x'_{new}" has been the newly adjusted pitch value, "x_{new}" has been the previous IQR value picked by memory requirements, the "rand ()" has been a random number ranging from "-1" and "1", and "FW" has been the "Freight Width" variable. The comparison, that controls the greatest variation in pitch adjustments, has been replicated through word choice changes. Also, it's regarded as among the variables which must be set explicitly.

As once the improvized IQR value has already been confirmed, the memory gets updated to reflect the way of comparing the fresh IQR value against the existing improvized value and the IQR value vector already stored within the memory. which contains the least fitness. Considering the freshly improvised solution is more desirable in terms of IQR fitness, it will replace the vector that has the weakest IQR score. This process of innovation and improvement is repeated until some termination requirement is met or the maximal number of possible repetitions has been achieved.

To establish the "HMCR", the PAR chooses a continuous, self-learning "Parameter Set List (PSL)" as follows:

- To begin, the PSL is initialized by randomly populating all that with values for the HMCR as well as the PAR. By a uniform distribution with HMCR values between "0.9" and "1.0", we calculate the IQR's HMCR effectiveness.
- Next, we generate the PAR values from a normal distribution, yielding values in the range "[0.0, 1.0]" again for PAR.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 4s
DOI: https://doi.org/10.17762/ijritcc.v11i4s.6312
***Article Received**: 28 December 2022 **Revised**: 30 January 2023 **Accepted**: 06 February 2023*

- The procedure will be repeated unless all elements in the PSL have been used.
- As a result, the FW factor has been calculated so that the HM hypervisor's higher IQR value may be used more effectively.
- The Equation (7) that follows shows that the FW factor decreases linearly with the iteration count:

$$FW(i) = \begin{cases} FW_{max} - \frac{FW_{max} - FW_{min}}{MI} 2i \text{ if } i < \frac{MI}{2} \\ FW_{min} \text{ elsewhere} \end{cases}$$

Eq→7

Wherein "i" has been the iteration range, "MI" has been the total amount of iterations, and "FW_{max}" and "FW_{min}" have become the highest and lowest values of FW used in the optimum selection of the IQR value. The probability of detecting the optimal load sharing "b(vm)" is calculated as a sum of its values using Equation (6).

IV. Results And Discussions

The proposed EGC-EF-ECS combination approach is evaluated experimentally under the Java Platform by use of the Cloudsim simulation. During the execution of the implementation, the Cloudsim simulation makes use of the database collected through Amazon's ECC. The proposed EGC-EF-ECS method effectively completes IDS operations on the Amazon ECC database, hence enhancing PP alongside optimal LB. Within a time constraint of 100 epochs, the optimal ranges for the frequency of CUs as well as CUS data used in research are 100–500 and 200–1000, respectively. By evaluating the proposed EGC-EF-ECS method in comparison to the existing BPA-SAB and DCRI-RI methods, its functionality is validated. Experiment assessments are carried out using the LBE, IDR, and DAT criteria.

(i) Evaluation of LBE's Efficiency

LBE has been considered as the proper identification of authorized CUs out of the total number of CUs inside the CC in particular to provide the necessary services to such CUs. Calculating the LBE using the following Equation (8):

$$LBE = \frac{Numberofauthorizedusersaredetected}{Totalnumberofcloudusersincloud} * 100$$

Eq→8

LBE is represented as a percentage in Equation (8). Authorized CU detection allows numerous CSs to share the workload more fairly. The method's CC transmission benefits are enhanced in situations where the LBE becomes considerable.

Table 1: LBE's Efficiency

CUs	BPA-SAB	DCRI-RI	EGC-EF-ECS
100	96	98	99
200	94	96	98
300	92	95	97
400	90	94	96
500	88	92	95

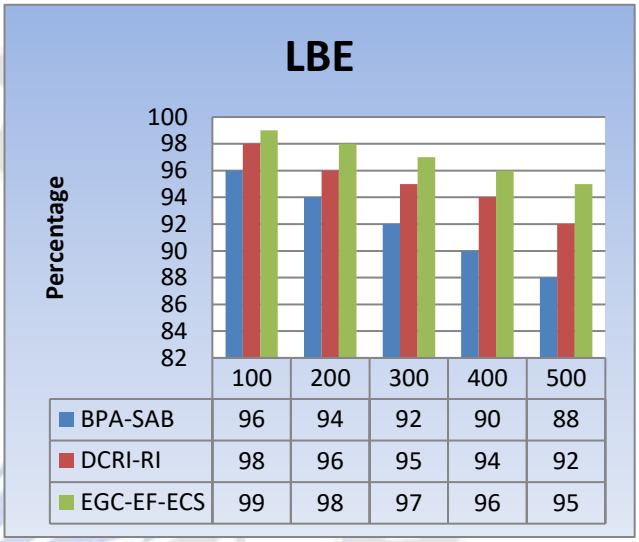

Figure 2: LBE's Efficiency

Research findings of LBE regarding the total number of CUs are shown in Table 1 and Figure 2. The frequency of CUs has been put into consideration for modeling operations as being from the limit of 100-500. LBE comparative is carried out between the proposed EGC-EF-ECS technique and the two state-of-the-art methods, BPA-SAB, and DCRI-RI. All these approaches progressively adjust the LBE to account for the changing CU count. About a large number of CUs, the proposed EGC-EF-ECS approach improves the LBE more so than the current methods BPA-SAB, and DCRI-RI.

(ii) Evaluation of IDR's Efficiency

The percentage of CUs that are correctly identified as intrusions, that is, CUs that are not authorized, to all CUs is known as the IDR. According to Equation (9) provided, we may calculate the IDR.

$$IDR = \frac{No.ofusers - No.ofintrusionscorrectlydiscovered}{No.ofusers} * 100$$

Eq→9

As seen in Equation (9), the IDR is generally expressed in percentage form. Better outcomes for encrypted CC transmitting data could be achieved whenever IDR efficiency seems to be large.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 4s
DOI: https://doi.org/10.17762/ijritcc.v11i4s.6312
***Article Received**: 28 December 2022 **Revised**: 30 January 2023 **Accepted**: 06 February 2023*

Table 2: IDR's Efficiency

CUs	BPA-SAB	DCRI-RI	EGC-EF-ECS
100	96	98	99
200	93	95	97
300	91	93	95
400	88	91	93
500	85	89	91

Research findings of IDR regarding the total number of CUs are shown in Table 2 and Figure 3. The frequency of CUs has been put into consideration for modeling operations as being from the limit of 100-500. IDR comparative is carried out between the proposed EGC-EF-ECS technique and the two state-of-the-art methods BPA-SAB, and DCRI-RI. All these approaches progressively adjust the LBE to account for the changing CU count. About a large number of CUs, the proposed EGC-EF-ECS approach improves the IDR more so than the current methods BPA-SAB, and DCRI-RI.

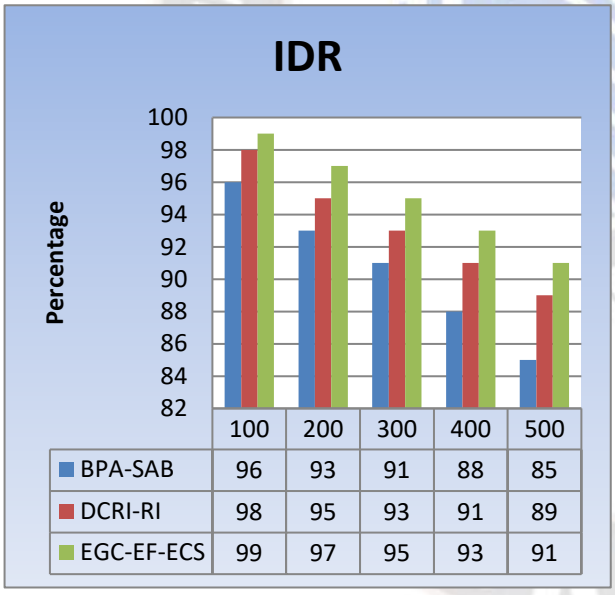

Figure 3: IDR's Efficiency

(iii) Evaluation of DAT's Efficiency

The DAT has been the sum of all time spent throughout the CC retrieving data. According to the Equation (10) provided, this DAT is calculated:

$$DAT = Number\ of\ data * time\ (accessingdatafromcloudserver)$$

Eq→10

Calculating a DAT using Equation (10) generates the results in "milliseconds (ms)". Whenever data access times are reduced, the method improves the efficiency of CC transmission of data.

Table 3: DAT's Efficiency

CUs DATA	BPA-SAB	DCRI-RI	EGC-EF-ECS
200	20	10	3
400	35	18	5
600	50	31	8
800	65	43	14
1000	80	66	27

Research findings of IDR regarding the total number of CUs are shown in Table 3 and Figure 4. The frequency of CUs data has been put into consideration for modeling operations as being from the limit of 200-1000. IDR comparative is carried out between the proposed EGC-EF-ECS technique and the two state-of-the-art methods BPA-SAB, and DCRI-RI. All these approaches progressively adjust the DAT to account for the changing CU's data volume. About a large number of CUs data, the proposed EGC-EF-ECS approach performs the DAT in minimal time so to the current methods BPA-SAB, and DCRI-RI.

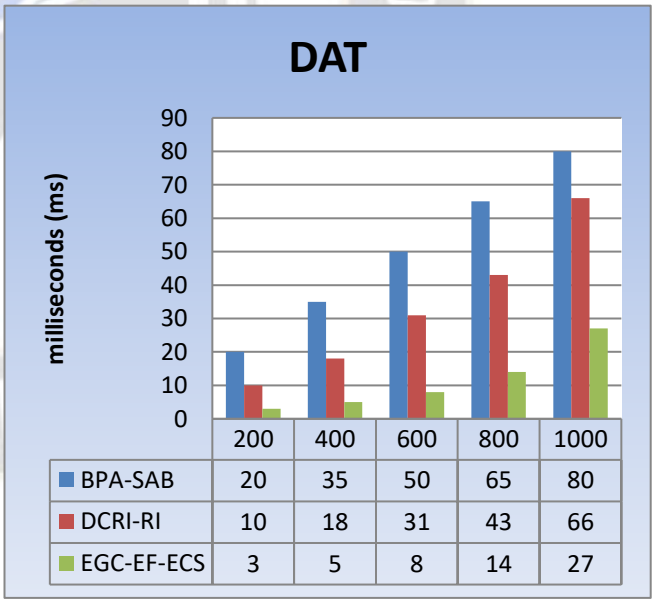

Figure 4: DAT's Efficiency

V. CONCLUSION

Our research aims at ways to improve the CC platform's intrusion identification capabilities against DDoS attacks that use VMs. The EGC method is used to overcome the TM formulating conflict. To begin, under the CC setting,

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 4s
DOI: https://doi.org/10.17762/ijritcc.v11i4s.6312
***Article Received**: 28 December 2022 **Revised**: 30 January 2023 **Accepted**: 06 February 2023*

the SS, as well as OS TS levels, have been obtained by calculating weights inside the EGC method. Then, it combines the TS from the host OS with the SS to establish trustworthy connections between both the hypervisor as well as the guest VMs. After that, we insist on using EF classification to recognize DDoS attacks inside a CC setting. Lastly, we suggest ECS as an optimum localization load allocation technique that may increase the detection mechanism while staying within a certain resource. Based on experimental data, it is clear that the proposed EGC-EF-ECS approach outperforms the state-of-the-art BPA-SAB and DCRI-RI approaches regarding the IDR, LBE, and DAT parameters. In the future, the proposed approach can be implemented in any real-time large-scale organization to analyze its efficiency.

REFERENCES:

[1] T. Alam, “Cloud computing and its role in the information technology”, IAIC Transactions on Sustainable Digital Innovation (ITSDI), vol. 1, no. 2, pp. 108–115, 2020.

[2] J. Muda, S. Tumsa, A. Tuni, and D. P. Sharma, “Cloud-enabled E-governance framework for citizen-centric services”, Journal of Computer and Communications, vol. 8, no. 7, pp. 63–78, 2020.

[3] A. A. Khan, A. A. Laghari, S. Awan, and A. K. Jumani, “Fourth industrial revolution application: network forensics cloud security issues”, Security Issues and Privacy Concerns in Industry 4.0 Applications, pp. 15–33, 2021.

[4] Sun. X, Liu. P, Singhal. A, "Toward Cyber resiliency in the Context of Cloud Computing [Resilient Security]". IEEE Secur. Priv. 2018, 16, 71–75.

[5] X. Lu, Z. Pan, and H. Xian, “An efficient and secure data sharing scheme for mobile devices in cloud computing”, Journal of Cloud Computing, vol. 9, no. 1, pp. 60–13, 2020.

[6] J. Li, H. Yan, and Y. Zhang, “Identity-based privacy-preserving remote data integrity checking for cloud storage”, IEEE Systems Journal, vol. 15, no. 1, pp. 577–585, 2021.

[7] H.-Y. Lin and Y.-M. Hung, “An improved proxy Re-encryption scheme for IoT-based data outsourcing services in clouds”, Sensors, vol. 21, no. 1, p. 67, 2020.

[8] Abraham, A.T., Thomson Fredrik, E.J. (2021). "Analysis of Task Scheduling Algorithms in Cloud Computing", Turkish Online Journal of Qualitative Inquiry (TOJQI), 12(2), 722–742.

[9] M. Ghobaei-Arani, A. Shahidinejad, "An efficient resource provisioning approach for analyzing cloud workloads: a metaheuristic-based clustering approach", J. Supercomput., 77 (2021), pp. 711-750.

[10] A.S. Voros, et al."The SMART4ALL High Performance Computing Infrastructure: Sharing high-end hardware resources via cloud-based microservices", 2021 31st International Conference on Field-Programmable Logic and Applications (FPL) (2021), pp. 384-385.

[11] D’Silva. D, Ambawade D.D. "Building a zero trust architecture using Kubernetes", In Proceedings of the 2021 6th international conference for convergence in technology (i2ct), Maharashtra, India, 2–4 April 2021; pp. 1–8.

[12] Albuali. A, Mengistu. T, Che. D, "ZTIMM: A zero-trust-based identity management model for volunteer cloud computing", In Proceedings of the International Conference on Cloud Computing, Honolulu, HI, USA, 18–20 September 2020; Springer: Cham, Switzerland, 2020; pp. 287–294.

[13] T. Adityasaisrinivas, K. Govinda, S. S. Manivannan, and E. Swetha, “Analysis of load balancing algorithms using cloud analyst”, International Journal of Recent Technology and Engineering, vol. 6, pp. 684–687, 2019.

[14] S. K. Mishra, B. Sahoo, and P. P. Parida, “Load balancing in cloud computing: a big picture”, Journal of King Saud University - Computer and Information Sciences, vol. 32, no. 2, pp. 149–158, 2020.

[15] S. Y. Mohamed, M. H. N. Taha, H. N. Elmahdy, and H. Harb, “A proposed load balancing algorithm over cloud computing (balanced throttled)”, International Journal of Recent Technology and Engineering, vol. 10, no. 2, pp. 28–33, 2021.

[16] Abraham, A.T., Thomson Fredrik, E.J. (2022). Ensuring the Security and Load Balancing in the Cloud Environment by BPA-SAB Method. In: Hu, YC., Tiwari, S., Trivedi, M.C., Mishra, K.K. (eds) Ambient Communications and Computer Systems. Lecture Notes in Networks and Systems, vol 356. Springer, Singapore. https://doi.org/10.1007/978-981-16-7952-0_37.

[17] Abraham, A.T., Thomson Fredrik, E.J. (2022). Ensuring the security and balancing the load in the cloud computing by DCRI-RI hybrid method. International Journal of Health Sciences, 6(S2), 9776–9793. https://doi.org/10.53730/ijhs.v6nS2.7559.