International Journal of Computer Networks and Applications (IJCNA)
DOI: 10.22247/ijcna/2022/215916 Volume 9, Issue 5, September – October (2022)

RESEARCH ARTICLE

Hypervisor Attack Detection Using Advanced Encryption Standard (HADAES) Algorithm on Cloud Data

R. Mangalagowri
Computer Science Department, SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India.
mangalar@srmist.edu.in

Revathi Venkataraman
School of Computing, SRM Institute of Science and Technology, Kattankulathur, Chengalpattu, Tamil Nadu, India.
revathin@srmist.edu.in

Received: 24 July 2022 / Revised: 20 September 2022 / Accepted: 25 September 2022 / Published: 30 October 2022

Abstract – Cloud computing demonstrates excellent power to yield cost-efficient, easily manageable, flexible, and charged resources whenever required, over the Internet. Cloud computing, can make the potential of the hardware resources to increase huge through best and shared usage. The growth of the cloud computing concept has also resulted in security challenges, considering that there are resource sharing, and it is moderated with the help of a Hypervisor which can be the target of malicious guest Virtual Machines (VM) and remote intruders. The hypervisor itself is attacked by hackers. Since the hypervisor is attacked, the VMs under the hypervisor is also attacked by the attackers. Hence, to prevent the problems stated above, in this study, Enhanced Particle Swarm Optimization (EPSO) with Hypervisor Attack Detection using Advanced Encryption Standard (HADAES) algorithm is introduced with the intent of improving the cloud performance on the whole. This work contains important phases such as system model, optimal resource allocation, and hypervisor attack detection. The system model contains the data center model, migration request model, and energy model over the cloud computing environment. Resource allocation is done by using the EPSO algorithm which is used to select the optimal resources using local and global best values. Hypervisor attack detection is done by using HADAES algorithm. It is helpful to determine the hypervisor and VM attackers also it is focused to provide higher security for cloud data. From the test result, it is concluded that the proposed algorithm yields superior performance concerning improved reliability, throughput, and reduced energy consumption, cost complexity, and time complexity than the existing methods.

Index Terms – Cloud Computing, Hypervisor Attack Detection, Resource Allocation, Enhanced Particle Swarm Optimization (EPSO), Advanced Encryption Standard (AES) algorithm.

1. INTRODUCTION

Cloud computing is an innovative means of merging a suite of technologies for the implementation of a novel phenomenon generating a place for users to avail of shared and customizable resources over the internet on the fly. Pcs and similar devices may access sharing services, programs, and data as requested thanks to this Internet-based design. This system exhibits several features that are common with distributed systems, and therefore, cloud computing also makes use of the features associated with networking. Hence security forms the predominant challenge of this system since haring forms the basis of the services of cloud computing [1]. Cloud computing facilitates people to share; anything is its resources, services, and information. It enables the organization with a framework, which can be sufficiently elastic ensuring an efficient network for the enterprise. The best means of service provisioning using cloud computing has increased its fascination among an extensive range of businesses.

Cloud computing becoming immensely popular is getting an increasing number of organizations getting adapted to the cloud. The adoption of the cloud in magnanimous numbers leads to bigger risks and problems, which can pose security hazards to cloud computing [2]. One further type of this vulnerability is extended loss of access, which impacts the services and the resources of the cloud. The distributed denials of services are highly complex and go on to advance with a quick speed getting rid of the entire authentication and the countermeasures. Four groups of assaults on virtual machines (VMs) may be distinguished: VM-to-VM, VM-to-hypervisor, hypervisor-to-VM, and hypervisor-to-hypervisor. Therefore, the exclusive use of conventional network security mechanisms like firewalls, intrusion prevention systems (IPS), and intrusion detection systems (IDS) cannot handle intrusions. A hostile VM may utilize the hypervisor rather than the actual network to construct new communication channels. This emphasizes the need for innovative structures to safeguard network security without relying on networks.

The effectiveness of IDS depends on its capacity to strike a balance between the number of defenses and the number of false positives or detecting errors.

Virtualization technology isolates the operating systems and the hardware on which they are run employing VMs and an underlying hypervisor. This separation isolation facilitates not just hardware resource sharing, but also the uncomplicated change of VMs between various physical hosts [3]. VM migration can be beneficial in terms of user mobility, load balancing, dealing with failures, and system administration. In the past years, researchers have primarily focused on the improvement of migration performance as studied in recent work [4]. The security aspects have been highlighted very less. But, the implementation of cloud computing is struck due to the missing security and trust assurance for the virtualization technology. Therefore, improvement of security and reliability must be focused more.

As the hypervisor is the virtualization manager it is highly susceptible to attacks. The hypervisor is the primary element of the virtualized system which is accountable for emphasizing the separation between virtual machines and resource control of the hardware [5]. The hypervisor forms the software that allows several guest virtual machines to execute at the same time on the same server. The hypervisor again constitutes the single point of failure. Therefore, the hypervisor also has to be taken care of for risks of compromise painstakingly [6].

The primary challenge that this technical work-study is hypervisor attack detection in cloud computing. Many different research efforts and techniques have been presented, however, security is not ensured considerably. The available techniques have a setback with attacks and security factors. To get rid of the problems stated above, in this technical work, Enhanced Particle Swarm Optimization (EPSO) with Hypervisor Attack Detection using Advanced Encryption Standard (HADAES) algorithm is presented with the intent of improving the overall performance in the cloud. The chief contribution made by this study involves the construction of a system model, resource allocation, and hypervisor attack detection. The proposed technique helps achieve improved security by applying efficient algorithms for the cloud environment. The other sections of the study are structured as given: Section 2 presents an outline of a few literary works in hypervisor attack detection and resource allocation. The proposed technique for the EPSO+HADAES algorithm is explained in Section 3. Section 4 explains the experimental results and performance analysis. Lastly, Section 5 provides the conclusions.

2. RELATED WORK

In [7], Cao e al (2021) a DDoS anomaly detection methodology that analyses user commonalities to find DDoS attack origins. The main objective of the method is to locate DDoS by cheaply observing how similarly active individuals interact with current users. The suggested model, which incorporates the crucial actions below, achieves this purpose. An example user set is created first. The queries of the live customers are again monitored to determine how closely every living customer resembles the sample individuals. Finally, discovered individuals will be marked as abnormal ones if the variance of resemblance surpasses the set criteria.

In [8], Velliangiriet al (2020) concentrated on creating a deep learning-based classifier that could predict DDoS attacks. Users' service requests are collected and categorized as log data. To reduce the classifier's straining time, a few key characteristics are selected from the log file using the Bhattacharya distance metric. Elephant Herd Optimization (EHO) is modified with the Taylor series to create the Taylor-Elephant Herd Optimization-based Deep Belief Network (TEHO-DBN), and the algorithm created in this way is used to train the DBN to identify DDoS attacks. The simulated findings demonstrate that the suggested TEHO-based DBN classifier demonstrates superior effectiveness, producing an improved efficiency of 0.830 when compared to state-of-the-art techniques while taking assessment criteria into account.

In [9], Varadharajan et al (2014) centered on the privacy services that a cloud provider may provide to its customers (tenants) as a component of its infrastructure as a defense against these assaults. The main contribution is a safety paradigm that offers adequate safety as a service that a cloud provider may deliver to its tenants and the customers of its tenants. The safety-as-a-service approach makes it flexible for the tenants to add additional security features that best meet their security requirements in addition to providing basic security to the provider to protect its cloud architecture. In this study, the design of the security framework is described and how the architecture defends against various kinds of attacks is discussed. It is realized by the security framework and the research work studies the analysis and the results of performance analysis. Additionally, the decision by a renter to opt-in for extra security services might help the cloud provider create a structure for billing the occupants for these extra security services.

In [10], Nikolaiet al (2014) presented the infrastructure and technique to make the best use of Utilizing hypervisor efficiency measurements, the virtualization technology at the heart of cloud computing performs intruder detecting protection. Using the hypervisor-collected efficiency metrics for virtual machines, which include attempts to read from and write to block devices, sent and collected packets, and CPU consumption, it can be demonstrated and checked if the malicious activity can be categorized with no extensive information on the operating system that runs within the virtual machines. There is no need for the proposed hypervisor-based cloud intrusion detection system to install extra software installed in virtual machines and offers several benefits in comparison to using host-based and network-based intruder detecting methods that may supplement these traditional intruder detecting methods.

In [11], Dildar et al (2017) recommended using Virtual Machines and Hypervisor Intrusion Detection System (VMHIDS), as a method for identifying and preventing hypervisor intrusions in a virtualization cloud environment. The VMHIDS has used a variety of features obtained from the other approaches through the execution of the tasks often which is capable of averting the occurrence of malicious events. The hypervisor attack is prevented by the VMHIDS. The hypervisor assault is lessened because of the VMHIDS.

In [12], Fenget al (2012) highlighted the topic of resource distribution in cloud computing, which is thought to be an optimizer for a large firm given the potential scale of their clientele and resources. With this issue in mind, a Particle Swarm Optimization (PSO) method is created. The algorithm's goal is to determine the appropriate task scheduler for resources based on several factors, including total task execution time, resource reservations, and the quality of service for each job. A Pareto-domination strategy is used in the algorithm aiding the search for multi-objective optimal solutions. It is proven from the results of experiments that the proposed algorithm is both effective and resourceful.

In [13], Tenget al (2020) analyzed Advanced Encryption Standard (AES) algorithm. Next, a modified advanced encryption standard aimed at data security in cloud computing is presented by using random disturbance information for enhancing data security. In addition, there is also improvement seen in column mix operation and key choreography in AES. At last, experiments are carried out on Hadoop. It is shown through a strategic security analysis and performance comparisons that the proposed solutions simultaneously guarantee the privacy of attributes and enhance the decryption efficiency in the case of outsourced data storage in mobile cloud computing.

In [14], Anumukonda (2021) gave an in-depth analysis of several security vulnerabilities, and different solutions used to solve them, along with a comparison of their strengths and weaknesses. Additionally, numerous assaults against hypervisors are thoroughly discussed, along with the defensive and attacking strategies utilized in each case. The two particular methods that will be investigated are important for identifying various hypervisor threats. One strategy focuses on the effective use of Genetic Algorithms (GA) to stop malicious assaults that are being launched against cloud hypervisors. The second strategy is to create a positive defensive strategy to detect and stop threats, increasing the level of cloud security via the efficient usage of a select number of AI-based apps.

2.1. Problem Statement

IDS are very prone to single-point faults or flaws, which intruders might find and take advantage of. The number of undiscovered assaults may be decreased by using several intruder-detecting devices, although this solution comes with additional expense. The effectiveness of IDS depends on its capacity to strike a compromise between the number of responders and the number of false positives or detecting losses. The primary challenge that this technical work-study is hypervisor attack detection in cloud computing. Many different research efforts and techniques have been presented, however, security is not ensured considerably. The available techniques have a setback with attacks and security factors.

3. PROPOSED METHODOLOGY

In this work, we have to create a scenario in which the hypervisor itself is attacked by hackers. Since the hypervisor is attacked, the VMs under the hypervisor is also attacked by the attackers. Hence, to get over the above-stated problem, in this research work, proposed framework is proposed which will overcome attacks that happened by attackers. The suggested system's general block diagram is shown in Figure 1.

3.1. System Model

The system model contains the data center model, migration request model, and energy model over the cloud computing environment. In this work, consider the number of resources, number of tasks, number of VMs, number of packets, and number of cloud users over cloud computing. A cloud is developed to finish a suite of programs/applications such that specific tasks can be completed. Running those programs necessitates using a few cloud resources [15]. The basic aspects comprising a resource allocation and VM migration problem in the cloud environment consist of the task of cloud users, cloud resources, data packets, and the resource allocation technique. Cloud resource allocation helps in the assignment of task-based resources considering that user satisfaction is ensured such that the task execution time is minimal. The cloud resources are shared by the users by entering tasks to the cloud, and the cloud resource allocation process re-assigns these tasks to the suitable resources abiding a mechanism, such that the effective allocation algorithm can leverage the processing capability of the cloud system to increase the throughput of the entire cloud system.

3.1.1. Data Center Model

The data center network is denoted in the form of a weighted graph $g(V,E)$. The vertices in Vindicate are either a server or a network middlebox. E is the set of all the edges in the data center network. $A[j,k]$ is the bandwidth capacity of two directly connected nodes $j\ and\ k\ in\ E$ and is zero otherwise. C_i is the capacity of node I in V.

RESEARCH ARTICLE

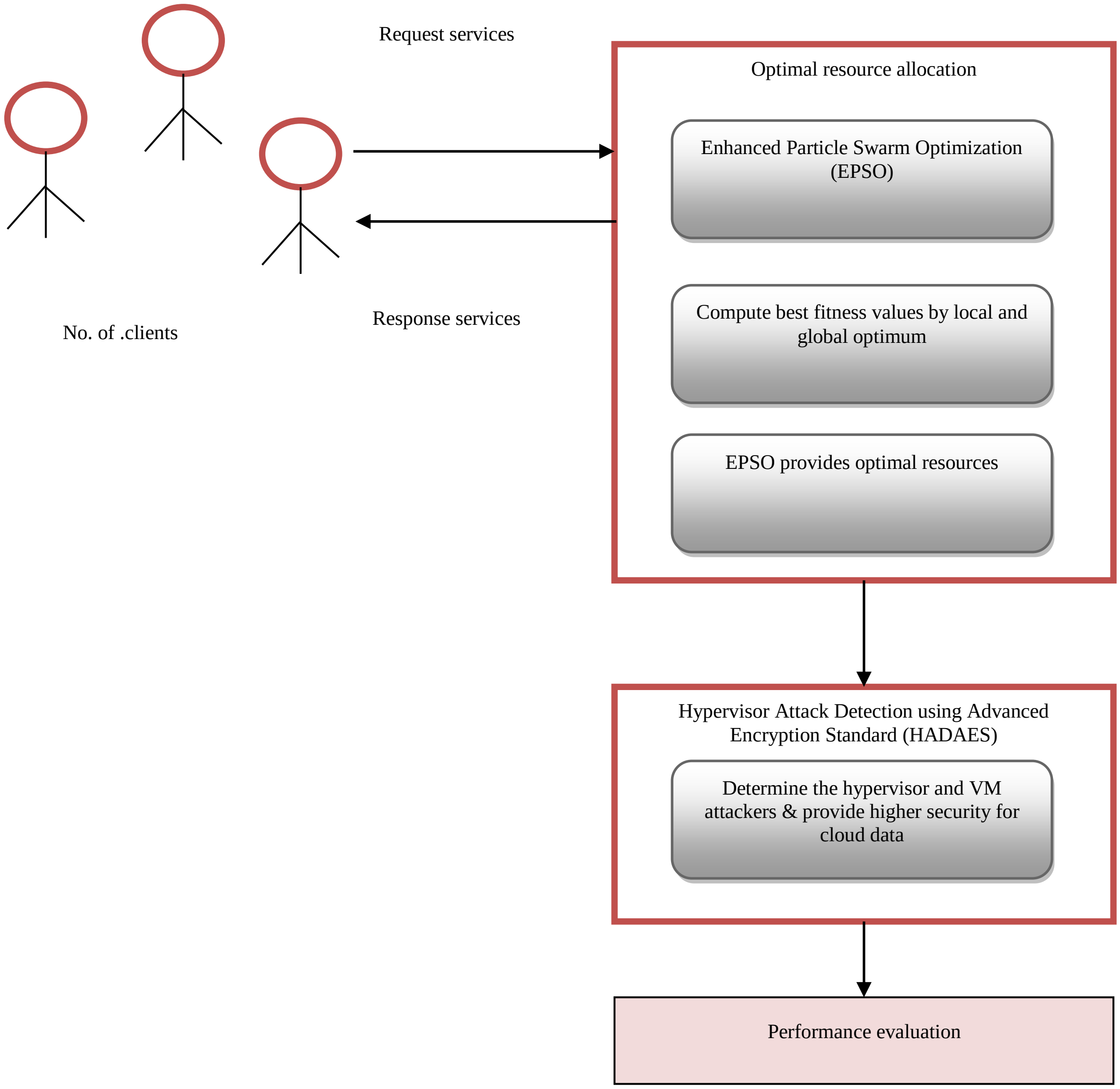

Request services
Response services
No. of .clients
Optimal resource allocation
Enhanced Particle Swarm Optimization (EPSO)
Compute best fitness values by local and global optimum
EPSO provides optimal resources
Hypervisor Attack Detection using Advanced Encryption Standard (HADAES)
Determine the hypervisor and VM attackers & provide higher security for cloud data
Performance evaluation

Figure 1 Overall Architecture of the Proposed System

3.1.2. Migration Request Model

A VM migration request R is represented as a quadruple $R(s, d, b, m, \mu)$, where s is the source host, d is the destination host, b bandwidth requirement, $and\ m$ is the amount of VM data that is to be transferred to the destination, and μ is a threshold on eviction time for the request.

3.1.3. Energy Model

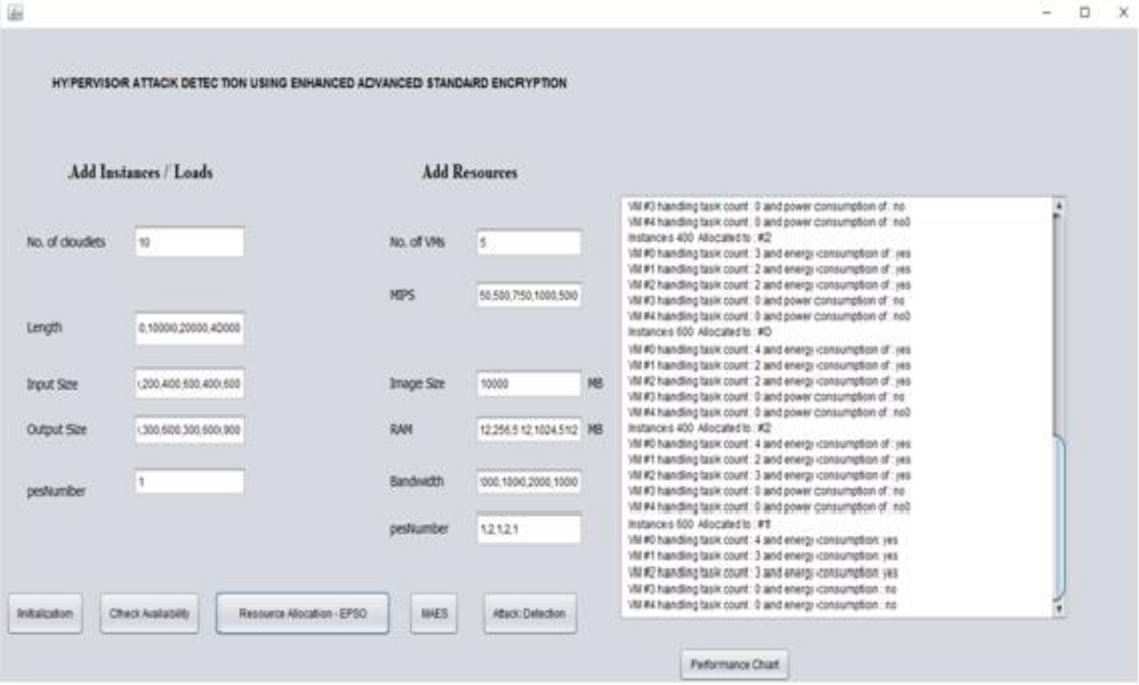
Middleboxes consist of network elements that do specialized tasks like load balancing, security, and performance enhancement in the data center networks. Either servers or middleboxes are considered intermediate nodes for Scatter-Gather VM migration. The power model considered for the server or middlebox is a utilization-based energy model. The power consumed by a server or middlebox at utilization as equation (1),

$$P_u = (P_{max} - P_{idel})u + P_{idel} \qquad (1)$$

Where P_{idel} is the power consumed by the node in an idle state, i.e when there is no load on it and P_{max} is the power consumed by the node at the maximum (100%) load. The energy consumed by the intermediate node for a given duration of time as equation (2),

$$E_i = \int_{t_0}^{t_1} P\big(u(t)\big)dt \qquad (2)$$

Figure 2 shows the total number of VM used in the simulation, and resources availability of VM.



HYPERVISOR ATTACK DETECTION USING ENHANCED ADVANCED STANDARD ENCRYPTION

Add Instances / Loads

No. of cloudlets: 10
Length: 0,10000,20000,40000
Input Size: ,200,400,600,400,600
Output Size: ,300,600,300,600,900
pesNumber: 1

Add Resources

No. of VMs: 5
MIPS: 50,500,750,1000,500
Image Size: 10000 MB
RAM: 12,256.5 12,1024,512 MB
Bandwidth: 1000,1000,2000,1000
pesNumber: 1,2,1,2,1

```
VM #3 handing task count : 0 and power consumption of : no
VM #4 handing task count : 0 and power consumption of : no0
Instances 400 Allocated to : #2
VM #0 handing task count : 3 and energy consumption of : yes
VM #1 handing task count : 2 and energy consumption of : yes
VM #2 handing task count : 2 and energy consumption of : yes
VM #3 handing task count : 0 and power consumption of : no
VM #4 handing task count : 0 and power consumption of : no0
Instances 600 Allocated to : #0
VM #0 handing task count : 4 and energy consumption of : yes
VM #1 handing task count : 2 and energy consumption of : yes
VM #2 handing task count : 2 and energy consumption of : yes
VM #3 handing task count : 0 and power consumption of : no
VM #4 handing task count : 0 and power consumption of : no0
Instances 400 Allocated to : #2
VM #0 handing task count : 4 and energy consumption of : yes
VM #1 handing task count : 2 and energy consumption of : yes
VM #2 handing task count : 3 and energy consumption of : yes
VM #3 handing task count : 0 and power consumption of : no
VM #4 handing task count : 0 and power consumption of : no0
Instances 500 Allocated to : #1
VM #0 handing task count : 4 and energy consumption: yes
VM #1 handing task count : 3 and energy consumption: yes
VM #2 handing task count : 3 and energy consumption: yes
VM #3 handing task count : 0 and energy consumption : no
VM #4 handing task count : 0 and energy consumption : no
```

Figure 2 VM Load and Resource Initialization

3.2. Optimal Resource Allocation Using Enhanced Particle Swarm Optimization (EPSO) Algorithm

In this work, resource allocation is done by using Enhanced Particle Swarm Optimization (EPSO) algorithm optimally. Resource Allocation (RA) addresses the allocation of the existing resources between cloud users and applications effectively and effectively. RA for IaaS in cloud computing yields many advantages: it is cost-economic since installation and update of the hardware or software is not necessary for the users to have access to the applications, and the system is flexible enough to permit access to applications and data on any system located worldwide, and no constraints exist on the medium or usage sites.

PSO approach takes its inspiration from a simple paradigm in the animal world, where a flock of birds flies out in search of food. During their search for the food location, an individual flies based on the direction not just from their individual experience but also guided by other birds, particularly the ones that are more nearby foods. In PSO, a flock of birds is referred to as a particle swarm because they resembled particles. A particle is encoded to make describing scheduling easier. After each particle has been evaluated for a certain duration, the key concept of PSO is to choose the optimal scheduling from that particle. It showed the expressions created for each particle using the equations (3-4),

$$v_i^{(t+1)} = wv_i^{(t)} + c_1r\left(pb_i^{(t)} - x_i^{(t)}\right) + c_2R\left(gb_i^{(t)} - x_i^{(t)}\right) \quad (3)$$

$$x_i^{(t+1)}=x_i^{(t)}+v_i^{(t+1)} \quad (4)$$

$v_i^{(t)}$& $x_i^{(t)}$ it denotes the speed and position of particle i in the t^th^ iteration correspondingly. $pb_i^{(t)}$and $gb_i^{(t)}$indicatesthe respective localized greatest of component i and the overall best of the swarms. r and R signify the random values in [0,1], and $w, c_1$and $c_2$refer to weight parameters.

In the PSO algorithm, every solution is regarded as a Particle in space. The individual position and speed exist for each one and the fitness value is decided using an optimal function. Moreover, every particle is aware of the best position and current location now in the whole swarm, which uses the following information to modify its current position: (a) The current position; (b) The current speed; (c) The earlier best position; (d) The ideal position of the whole swarm.

At first, the initialization of the population is done. Next, the population is randomly divided into a group of subpopulations [16,17]. In addition, the problem space is divided into different virtual subspaces. Every subspace is given by a hypercube. Rather, every dimension amongst all the dimensions is divided into the same-sized partitions. Hence, it has subspaces. Later, the particles will be moved at a lower speed to the more useful subspaces. In addition, in every subpopulation, a specific group of movement coefficients is utilized. To add, the specific set of the movement coefficients for every subpopulation is changed adaptively during optimization. In the end, the best solution generated during optimization is regarded as the ideal solution for the problem that the proposed optimization algorithm known as the Enhanced Particle Swarm Optimization (EPSO) algorithm. The best and the mean costs of the population members and the execution time when applying the EPSO method. In this study, an innovative time-adaptive PSO is proposed based on the movement patterns named the movement pattern adaptation PSO (EPSO).

1. A large V_c and a small $|\rho_1|$ are desirable during the early phases to facilitate the particle in searching through an extensive range (highρ_1)and not towards any specific direction (small $|\rho_1|$). F= 1 would be a balance between $p$and$g$, and this would prove to be a better selection during the start.
2. With the rising number of iterations, a higher $|\rho_1|$is useful to maintain good directions that the particles get. A higher $V_c$is quite advantageous since the exploration could be still useful.
3. It would be better for the next phases of the search to highlight the best-got solutions (bigger), surrounding the best-available solutions (lesser$|\rho_1|$ and lesserV_c)

The EPSO can be achieved by changing all the coefficients $(w, c, and\ \alpha)$ at the same time. Depending on this configuration, the values of V_c, ρ_1, and F through equation (5),

$$V_c^{(t)} = \begin{cases} V_{max} & t < t_1 \\ \frac{(t-t_1)(V_{min}-V_{max})}{t_2-t_1} + V_{max} t_1 < t < t_2 & \\ V_{min} & t > t_2 \end{cases} \quad (5)$$

It is ensured by this function that the value of$V_c$is the highest at the initial and last at the final phases of the search when it is linearly decremented from iteration$t_1\ to\ t_2$by equation (6),

$$F^{(t)} = \begin{cases} F_{min} & t < t_1 \\ 1 & t_1 < t < t_2 \\ F_{max} & t > t_1 \end{cases} \quad (6)$$

This equation makes sure that the personal best is focused the most, and later a balanced search is carried out surrounding both personal and global best and at last focuses the search surrounding the global best. The values of coefficients are changed during the time of optimization. As discussed in Algorithm 1.

```
For each resource
Initialize the particles (resources) with velocity
Initialize the position
Do
For each resource
Calculate the fitness values
If fitness value is more than the best fitness saved
Fix  current value to be new pbest
Fix  current value to be new pbest
End
End
Select the resource having the best fitness value of all resources as the best
For each resource
Compute the resource with velocity
Compute the resource position
Update resources with velocity and position using equations (3-4)
Adaptively change the coefficients using equations (5-6)
No. of. Tasks and no. of cloud resources
Compute execution time and response
Select best resources
Return optimal solution
```

Algorithm 1 Enhanced Particle Swarm Optimization (EPSO)

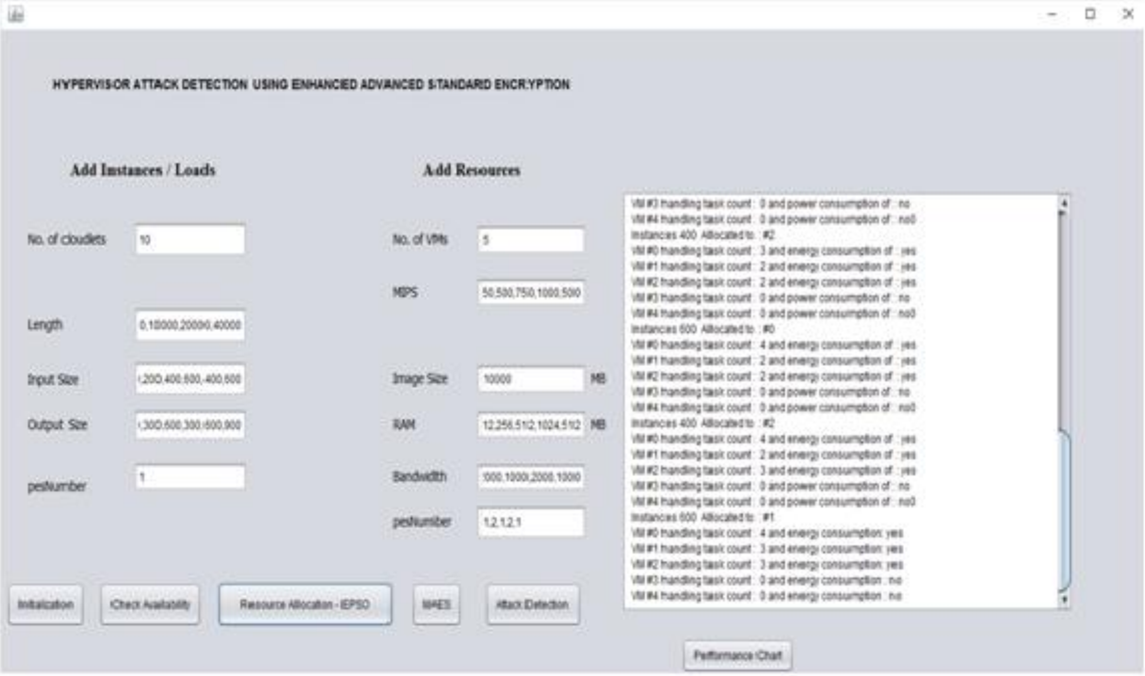
The essential parameters such as QoS are time, cost, and reliability. As such, cost and time are inversely proportional to each other; if a task can be allocated more users, thus advanced resources and tools may be used, resulting in lesser time for its completion and vice-versa. This research work aims at balancing the cost and time factor by using optimal resource allocation, henceforth, resulting in improved reliability as well as reducing failure rates. By allocation of resources as per user needs significantly raises the QoS coefficient values.

Figure 3 shows the optimal number of resources used by VM with their energy usage achieved by proposed EPSO algorithm.

3.3. Hypervisor Attack Detection Using Advanced Encryption Standard (HADAES) Algorithm

In this work, hypervisor attack detection is performed through the Advanced Encryption Standard (AES) algorithm effectively.AES depends on a design paradigm called a substitution–permutation network [18]. AES is a variation of Rijndael with a key size of 128, 192, or 256 bits and a predefined block size of 128 bits. Rijndael per se, on the other hand, is specified using block and key sizes, which may be any multiple of 32 bits and have a bit count between 128 and 256. For 128-bit keys, there are 10 rounds; for 192-bit keys, there are 12 rounds; and for 256-bit keys, there are 14 rounds. Each cycle includes several computing steps, one of which is dependent only on the encryption key. A group of inverse rounds is used for transforming the ciphertext into the actual plaintext making use of the same encryption key.

Authentication refers to the process through which the system makes a confirmation about the identity of a user and authorization indicates the access limits over what the user is permitted on the system after his/her authentication is done. The hypervisor also referred to as the VM monitor has the responsibility of VMs management in cloud computing. It carries out the monitoring of the resource sharing happening among the virtual machines. Additionally, it removes any virtual machines that are not licensed from the cloud environment [18]. Creation is one of the main duties of the hypervisors, termination, movement of the VMs, and resource allocation.


HYPERVISOR ATTACK DETECTION USING ENHANCED ADVANCED STANDARD ENCRYPTION

Add Instances / Loads

No. of cloudlets: 10
Length: 0,10000,20000,40000
Input Size: ,200,400,600,-400,600
Output Size: ,300,600,300,600,900
pesNumber: 1

Add Resources

No. of VMs: 5
MIPS: 50,500,750,1000,500
Image Size: 10000 MB
RAM: 12,256,512,1024,512 MB
Bandwidth: 000,1000,2000,1000
pesNumber: 1,2,1,2,1

```
VM #3 handling task count : 0 and power consumption of : no
VM #4 handling task count : 0 and power consumption of : no0
Instances 400  Allocated to : #2
VM #0 handling task count : 3 and energy consumption of : yes
VM #1 handling task count : 2 and energy consumption of : yes
VM #2 handling task count : 2 and energy consumption of : yes
VM #3 handling task count : 0 and power consumption of : no
VM #4 handling task count : 0 and power consumption of : no0
Instances 600  Allocated to : #0
VM #0 handling task count : 4 and energy consumption of : yes
VM #1 handling task count : 2 and energy consumption of : yes
VM #2 handling task count : 2 and energy consumption of : yes
VM #3 handling task count : 0 and power consumption of : no
VM #4 handling task count : 0 and power consumption of : no0
Instances 400  Allocated to : #2
VM #0 handling task count : 4 and energy consumption of : yes
VM #1 handling task count : 2 and energy consumption of : yes
VM #2 handling task count : 3 and energy consumption of : yes
VM #3 handling task count : 0 and power consumption of : no
VM #4 handling task count : 0 and power consumption of : no0
Instances 600  Allocated to : #1
VM #0 handling task count : 4 and energy consumption: yes
VM #1 handling task count : 3 and energy consumption: yes
VM #2 handling task count : 3 and energy consumption: yes
VM #3 handling task count : 0 and energy consumption : no
VM #4 handling task count : 0 and energy consumption : no
```

Figure 3 Optimal Resource Allocation and Energy Usage

The Hypervisor attack classified as the external attack refers to the way the weaknesses of the hypervisor are taken advantage of letting the cybercriminal get access and get authorized with the hypervisors [18]. At first, the cybercriminals wage an attack on a specific virtual machine. The rogue virtual machine will then infiltrate the hypervisor. Later, the cybercriminals use the chances to increase the attacks on every VM, which runs under the compromised hypervisor. Finally, the compromised hypervisor will gradually infect all of the virtual machines. To detect and discard the hypervisor attacks, the double key-based AES algorithm is introduced. Hypervisor-based attacks observe the system metrics using the cloud requests that are made from the hypervisor and help in the detection of any probable misuse activities. A Hypervisor-based attack system carries out the tracking and monitoring of the communication happening between VM, hypervisor, and VM, and virtual networking. This detection approach operates within the hypervisor and it could prove to be efficient in the cloud.

Keyword Generation: In this work, AES encryption key generation is an important process for encryption purposes. The input size and the size of the key used for encryption are equivalent. Here, the suggested method uses 16 bytes of the encryption key for encryption. The 16 bytes key is arranged in the form of 4 × 4matrices by equation (7),

$$\text{Key}=\begin{matrix} k0 & k4 & k8 & k12 \\ k1 & k5 & k9 & k13 \\ k2 & k6 & k10 & k14 \\ k3 & k7 & k11 & k15 \end{matrix} \qquad (7)$$

Generally, the AES algorithm has different phases of processing for the encryption of the input data files. Additionally, it creates an iterated block cipher with a 128-block size and adjustable key length. The different transformations operate on intermediary outcomes known as states, which are primarily composed of a rectangle array of bytes. The rectangle array has the size of 4 x 4 matrices if the block size is either 128 bits or 16 bytes, as shown in the table below.

Block size of the AES algorithm

b0	b1	b2	b3	b4	b5	b6	……………	b15

The arrangement of the input files is 4 x 4 as illustrated below. The first four-byte of the input by equation (8),

$$\begin{matrix} b0 & b4 & b8 & b12 \\ b1 & b5 & b9 & b13 \\ b2 & b6 & b10 & b14 \\ b3 & b7 & b11 & b15 \end{matrix} \qquad (8)$$

The state has few operations at each round. The operations include:

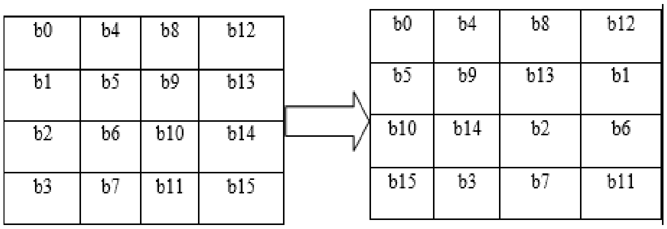
- Sub-bytes
- Shift row
- Mix column
- Add round key

Sub-byte operation: The sub-byte's procedure generally involves a nonlinear byte replacement that affects each state byte separately. The reversible substitution table (S-Box), and is built by making use of two transformations

Round-based Shift operation: In this normal shift row operation, the following operations are performed,

- 1st row to 0 positions to the left
- 2nd row to 1 position to the left
- 3rd row to 2 positions to the left
- 4th row to 3 positions to the left

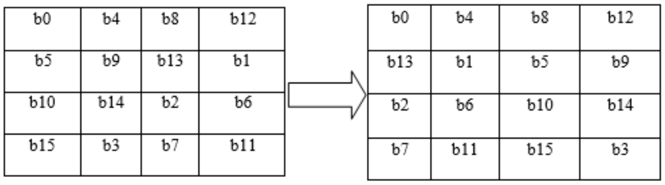
Generally, for 128-bit, the AES algorithm carries out ten-round operations. Here, the novel system is improved by changing the shift operation based on the round operation. At every round, the sub-bytes operation is carried out at first. Then, the round-based shift operation is conducted. At every round, the proposed technique checks if the round number is even or odd. If the round number is 1 (even), the shift rows transformation works on the rows of the state array. If the round number is 2 (even), the shift rows transformation moves to two places in the matrix. The shift operation, as illustrated in Figure 4 and Figure 5.



b0	b4	b8	b12
b1	b5	b9	b13
b2	b6	b10	b14
b3	b7	b11	b15

b0	b4	b8	b12
b5	b9	b13	b1
b10	b14	b2	b6
b15	b3	b7	b11

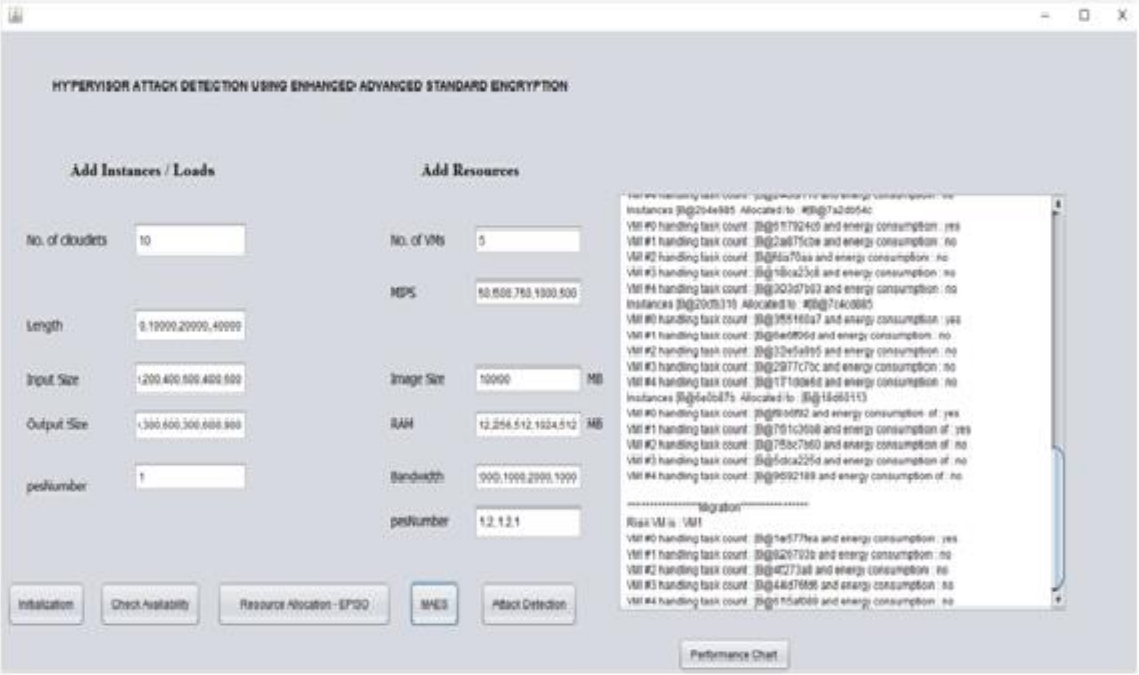
Figure 4 Shift Operation for Round Number 1 (Even)



b0	b4	b8	b12
b5	b9	b13	b1
b10	b14	b2	b6
b15	b3	b7	b11

b0	b4	b8	b12
b13	b1	b5	b9
b2	b6	b10	b14
b7	b11	b15	b3

Figure 5 Shift Operation for Round Number 2 (Even)



HYPERVISOR ATTACK DETECTION USING ENHANCED ADVANCED STANDARD ENCRYPTION

Add Instances / Loads

No. of cloudlets: 10
Length: 0,10000,20000,40000
Input Size: 200,400,500,400,500
Output Size: 300,600,300,600,900
pesNumber: 1

Add Resources

No. of VMs: 5
MIPS: 50,500,750,1000,500
Image Size: 10000 MB
RAM: 12,256,512,1024,512 MB
Bandwidth: 000,1000,2000,1000
pesNumber: 1,2,1,2,1

```
Instances [I@204e985 Allocated to : #[I@7a2db54c
VM #0 handing task count : [I@51f7924c0 and energy consumption : yes
VM #1 handing task count : [I@2a875c0e and energy consumption : no
VM #2 handing task count : [I@f0a70aa and energy consumption : no
VM #3 handing task count : [I@18ca23c8 and energy consumption : no
VM #4 handing task count : [I@303d7b93 and energy consumption : no
Instances [I@20cf5316 Allocated to : #[I@7c4c0885
VM #0 handing task count : [I@3f55160a7 and energy consumption : yes
VM #1 handing task count : [I@6e0ff06d and energy consumption : no
VM #2 handing task count : [I@32e5a9b5 and energy consumption : no
VM #3 handing task count : [I@2977c70c and energy consumption : no
VM #4 handing task count : [I@171d0e6d and energy consumption : no
Instances [I@6e0b87b Allocated to : [I@18d60113
VM #0 handing task count : [I@f0b0f92 and energy consumption of : yes
VM #1 handing task count : [I@761c3698 and energy consumption of : yes
VM #2 handing task count : [I@758c7b60 and energy consumption of : no
VM #3 handing task count : [I@5dca225d and energy consumption of : no
VM #4 handing task count : [I@9692189 and energy consumption of : no

******************Migration********************
Risk VM is : VM1
VM #0 handing task count : [I@1e577fea and energy consumption : yes
VM #1 handing task count : [I@826703b and energy consumption : no
VM #2 handing task count : [I@4f273a8 and energy consumption : no
VM #3 handing task count : [I@44d7688 and energy consumption : no
VM #4 handing task count : [I@6f5af089 and energy consumption : no
```

Figure 6 HADAES Encryption of VMs

Mix-column operation: The mix-column operation utilizes the modern mathematical computations

Add round key: The state is subjected to a bitwise XOR in add round key to apply the organization key to it. Utilizing the key scheduling, the round key may be derived using the cipher key.

The HADAES technique provides security to both the hypervisor and VMs from both internal and external attacks launched on the cloud environment. The constant monitoring performed by HADAES from hypervisors or VMs facilitates the analysis of real-time occurrences for automatic detection and defense against suspicious events. HADAES observes and keeps track of every file and process, which involves communication within the hypervisor in cloud computing. In addition, since HADAES is positioned on both VMs and hypervisors, new attacks or malicious attacks launched on hypervisors can be identified conveniently for rapid countermeasures.

Figure 6 shows the HADAES of resources used by VM.

4. EXPERIMENTAL RESULTS

In this CloudSim version, 4.0 were utilized for the simulation of a cloud data center, it comprises 800 heterogeneity physical hosts, of which 50% are HP ProLiant ML110 G4 and 50% are HP ProLiant ML110 G5, and which contains a mix of both types. Tables 1 provides the specifications for the physical hosts and virtual machines (VM) employed [18]. Security, energy, and QoS effects of the solution are evaluated. Table 2 depicts the performance comparison using the current and proposed techniques. The parameters used for each parameter are as follows:

Table 1 Physical Hosts Specification

VM Type	Type 1	Type 2	Type 3	Type 4
Total MIPS	2600	1860	1000	500
Total Processor units	2	2	2	2
Total RAM	8 GB	8 GB	8 GB	8 GB
Network bandwidth	100 Mbit/ seconds	100 Mbit/ seconds	100 Mbit/ seconds	100 Mbit/ seconds
Total storage size	2.5 GB	2.5 GB	2.5 GB	2.5 GB

Table 2 Performance Comparison of Security Methods

Methods	Time Complexity (sec)	Cost Complexity ($ per GB)	Throughput (Mbps)	Energy consumption (J)	Reliability (%)
PSO	52	0.10	0.80	32	80.50
TEHO-DBN	40	0.092	0.87	23	85.50
VMHIDS	25	0.046	0.94	16	93.90
EPSO+HADAES	10	0.010	0.98	8	96.80

Security measure: The temporal average for a VMI's R_i throughout its existence will serve as a representation of the VMI's average risk score, and the mean of all average risk scores will serve as the total threat rating. The risk score of the cloud will be denoted by the mean value of all of the virtual machines (VMs) that are running in the data center of the cloud. It is described by equation (9),

$$R^{i}_{avg} = \frac{1}{t-t_0} * \int_{t_0}^{t} R^{i}(t)dt \qquad (9)$$

where$t_0$indicates the start time VMI in the system, indicates the end time of VMI, $R_i$specifies the instance security risk of VMI and R_i avg gives the time average of security risk for VM_I

Energy measure: The rate of power requirements for a host at each given time instant may be approximated by utilizing a power model that is based on the amount of CPU utilization for each hot type. This model is provided in [19], and it makes use of the data that is provided in [20]. It is possible to express, using an equation, the amount of energy that a host uses throughout the period tp (10).

Ep= (fromPower +(toPower-fromPower)/2) *tp (10)

Where power and power indicate the start and end times' respective power consumption rates. The overall amount of energy used E denotes the cumulative total energy used by all hosts throughout their existence. In this research work, the performance comparison between the proposed SDN-based

CAC algorithm and the available PSO and DVMC approaches is carried out in terms of reliability, time complexity, cost complexity, throughput, and energy consumption.

Time complexity: The system yields improved performance less time is consumed for the execution of the proposed algorithm.

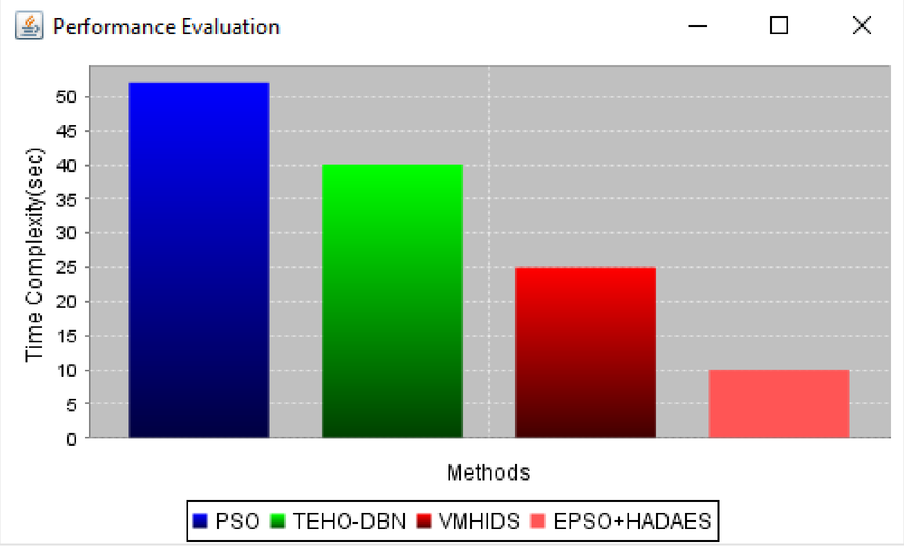


Figure 7 Time Complexity vs. Attack Detection Methods

Figure 7 shows the comparative analysis between the available and recommended methods according to their temporal complexity. The duration complication number is measured across the y-axis, while the methods are shown across the x-axis. The existing techniques including PSO, TEHO-DBN, and VMHIDS algorithms exhibit higher time complexity of 52 seconds, 40 seconds, and 25 seconds, while the proposed algorithm depicts lower time complexity of 10 seconds. In this proposed research work, optimal resources are selected by using the proposed algorithm via the best fitness function. The proposed work increases the response time speed hence it maximizes the overall VM migration performance. It can be concluded from the results that the proposed algorithm helps improve the efficiency of resource allocation over cloud computing.

Cost complexity: The cloud is superior to the proposed scheme demonstrating reduced cost complexity.

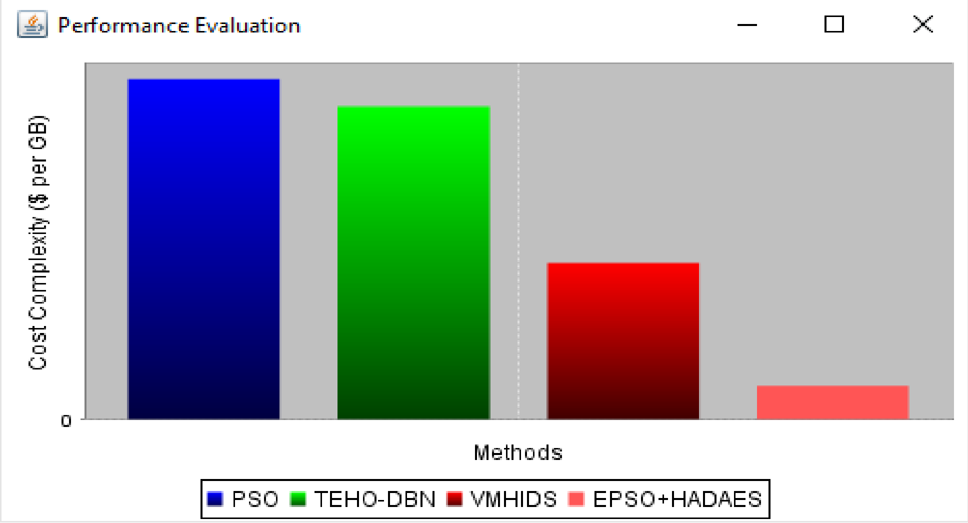


Figure 8 Cost Complexity vs. Attack Detection Methods

Figure 8 shows the comparison evaluation between the available and recommended methods according to expense difficulty. The cost complexity number is represented across the y-axis and the methods are displayed across the x-axis.

The contemporary techniques including PSO, TEHO-DBN, and VMHIDS algorithms exhibit higher cost complexity of 0.10($ per GB), 0.092($ per GB), and 0.046($ per GB), while the proposed algorithm demonstrates lower cost complexity of 0.010($ per GB). In this proposed research work, optimal resources are selected by using the EPSO algorithm via best fitness function values. It can be concluded from the result that the proposed algorithm improves security during hypervisor attacks in the cloud environment.

Throughput: The pace at which data packets are sent at a given transfer with success over the cloud or wireless links.

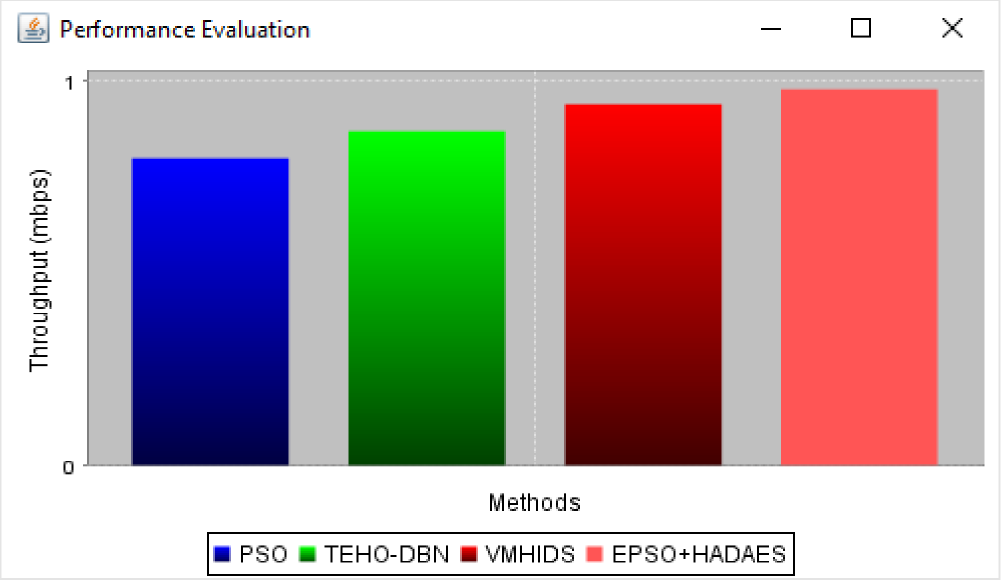


Figure 9 Throughput Comparison vs. Attack Detection Methods

Figure 9 illustrates the comparison between the PSO, TEHO-DBN, VMHIDS, and EPSO+HADAES techniques for the throughput metric. It shows that the existing PSO, TEHO-DBN, and VMHIDS algorithms provide lower throughput of 0.80 Mbps, 0.87 Mbps, and 0.94mbps, whereas the proposed scheme provides higher throughput of 0.98 Mbps. The proposed method improves the data transmission speed by allocating the optimal resources over the cloud environment. Hypervisor-based attacks observe the system metrics using the cloud requests from the hypervisor and identify any probable rogue activities. Thus it is used to increase the throughput efficiently in the cloud environment.

Energy consumption: Energy consumption is defined as the average energy required for the transmission, reception, or relaying functions of a packet to a network node during a certain period.

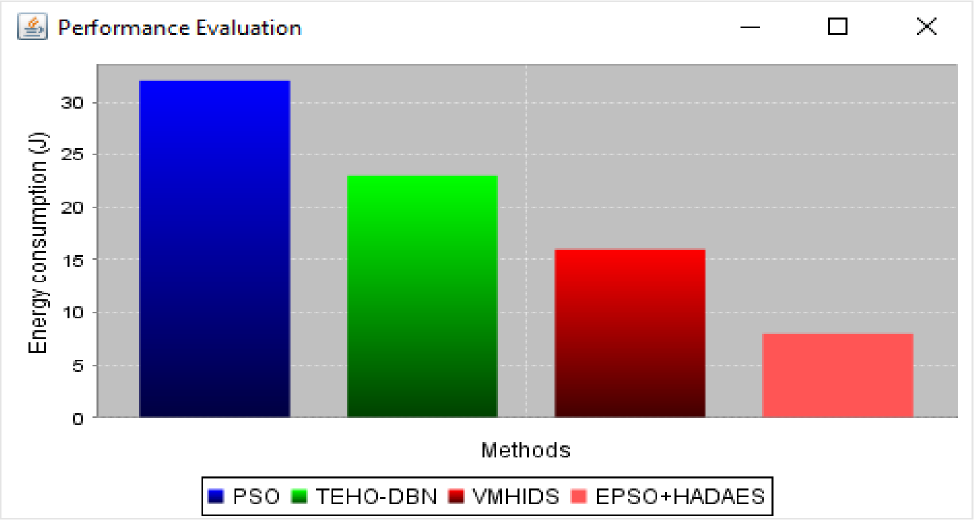


Figure 10 Energy Consumption vs. Attack Detection Methods

Figure 10, the comparison in terms of energy usage between the currently available, PSO, TEHO-DBN, and VMHIDS algorithms is shown. It is observed that the contemporary techniques consume much energy of 32 J, 23 J, and 16 While the energy usage of the proposed scheme is 6 J. The proposed method consumed lower energy for large data transmission through the best energy model construction.

Reliability: The proposed scheme offers reliable performance while the algorithm yields higher security.

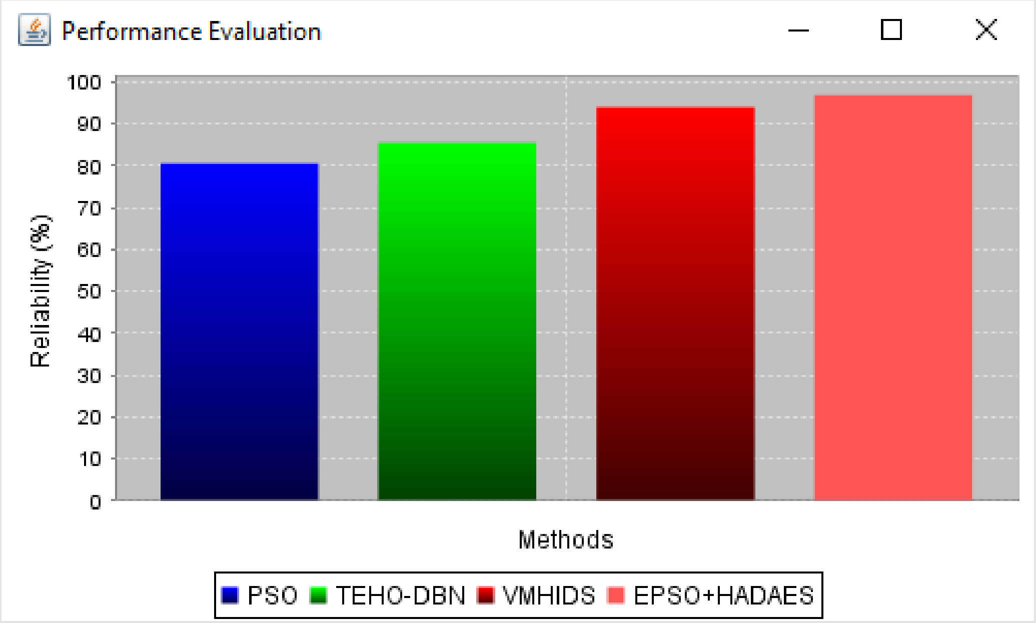


Figure 11 Reliability Comparison vs. Attack Detection Methods

Figure 11, it can be inferred that the comparison of reliability employing the contemporary, PSO, TEHO-DBN, and IOFM algorithms give lesser reliability values of 80.50%, 85.50%, and 93.90%. The proposed method provides a better security level of 96.8%, for large data transmission through the security measure. Un-authenticated users cannot able to access the data and it involves the prefer to entail limit imposed on accessing a place or any other resource when access management provides the process description. The purpose of the AES mechanism is to protect the data and detect hypervisor attacks effectively. The proposed technique provides security to both the hypervisor and VMs from both internal and external attacks launched on the cloud environment.

5. CONCLUSION AND FUTURE WORK

In this study, the EPSO+HADAES algorithm is introduced to detect and discard the hypervisor attacks and VM attacks over the cloud environment. Initially, the system model is constructed through the data center model migration model and energy model. Then the resource allocation is achieved by applying the EPSO algorithm and it is utilized for picking the resources optimally via local and global fitness values. After the allocation process is completed, we have to create the scenario the hypervisor itself is attacked by the hackers. Since the hypervisor is attacked, the VMs under the hypervisor is also attacked by the attackers. Then, the HADAES algorithm is proposed to detect and discard the VM attacks and hypervisor attacks considerably using a double key-based encryption process. Hence it increases the security higher using the AES process. Finally, the result proves that the proposed algorithm yields improved throughput, security, and lower energy consumption, time complexity, and cost complexity than the existing approaches. In future work, effective soft computing and machine learning algorithms can be developed to deal with various attacks prominently.

REFERENCES

[1] Gan, C., Feng, Q., Zhang, X., Zhang, Z., and Zhu, Q., 2020. Dynamical propagation model of malware for cloud computing security. IEEE Access, 8, pp.20325-20333.

[2] Meng, T., Wolter, K., Wu, H. and Wang, Q., 2018. A secure and cost-efficient offloading policy for mobile cloud computing against timing attacks. Pervasive and Mobile Computing, 45, pp.4-18.

[3] Desai, M.R. and Patel, H.B., 2015, Efficient virtual machine migration in cloud computing. In 2015 Fifth international conference on communication systems and network technologies, pp. 1015-1019.

[4] Hanini, M., Kafhali, S.E. and Salah, K., 2019. Dynamic VM allocation and traffic control to manage QoS and energy consumption in a cloud computing environment. International Journal of Computer Applications in Technology, 60(4), pp.307-316.

[5] Perez-Botero, D., Szefer, J. and Lee, R.B., 2013, Characterizing hypervisor vulnerabilities in cloud computing servers. In Proceedings of the 2013 international workshop on Security in cloud computing (pp. 3-10).

[6] Nezarat, A. and Shams, Y., 2017. A game theoretic-based distributed detection method for VM-to-hypervisor attacks in a cloud environment. The Journal of Supercomputing, 73(10), pp.4407-4427.

[7] Cao, T., Mao, J., Bhattacharya, T., Peng, X., Ku, W.S. and Qin, X., 2021, DDoS Detection Systems for Cloud Data Storage. In 2021 Third IEEE International Conference on Trust, Privacy and Security in Intelligent Systems and Applications (TPS-ISA), pp. 183-190.

[8] Velliangiri, S. and Pandey, H.M., 2020. Fuzzy-Taylor-elephant herd optimization inspired Deep Belief Network for DDoS attack detection and comparison with state-of-the-art algorithms. Future Generation Computer Systems, 110, pp.80-90.

[9] Varadharajan, V. and Tupakula, U., 2014. Security as a service model for cloud environment. IEEE Transactions on Network and Service Management, 11(1), pp.60-75.

[10] Nikolai, J. and Wang, Y., 2014, Hypervisor-based cloud intrusion detection system. In 2014 International Conference on Computing, Networking and Communications (ICNC), pp. 989-993.

[11] Dildar, M. S., Khan, N., Abdullah, J. B., & Khan, A. S. (2017). An effective way to defend the hypervisor attacks in cloud computing. In 2017 2ndInternational Conference on Anti-Cyber Crimes (ICACC), pp. 154-159.

[12] Feng, M., Wang, X., Zhang, Y. and Li, J., 2012, Multi-objective particle swarm optimization for resource allocation in cloud computing. In 2012 IEEE 2nd International Conference on Cloud Computing and Intelligence Systems, vol. 3, pp. 1161-1165.

[13] Teng, L., Li, H., Yin, S., & Sun, Y. (2020). A Modified Advanced Encryption Standard for Data Security. Int. J. Netw. Secure., 22(1), 112-117.

[14] Anumukonda, N.S.K., Yadav, R.K. and NS, R., 2021, A Painstaking Analysis of Attacks on Hypervisors in Cloud Environment. In 2021 6th International Conference on Machine Learning Technologies, pp. 150-157.

[15] Annadanam, C.S., Chapram, S. and Ramesh, T., 2020. Intermediate node selection for Scatter-Gather VM migration in the cloud data center. Engineering Science and Technology, an International Journal, 23(5), pp.989-997.

[16] Bansal, M. and Malik, S.K., 2020. A multi-faceted optimization scheduling framework based on the particle swarm optimization algorithm in cloud computing. Sustainable Computing: Informatics and Systems, 28, pp.1-8.

[17] Saeedi, S., Khorsand, R., Bidgoli, S.G. and Ramezanpour, M., 2020. Improved many-objective particle swarm optimization algorithm for scientific workflow scheduling in cloud computing. Computers & Industrial Engineering, 147, pp.1-23.

[18] Pendli, V., Pathuri, M., Yandrathi, S. and Razaque, A., 2016, Improvising performance of advanced encryption standard algorithm. In 2016 second international conference on mobile and secure services (MobiSecServ), pp. 1-5.

[19] Kaushik, S. and Gandhi, C., 2020. Capability-based outsourced data access control with assured file deletion and efficient revocation with trust factor in cloud computing. International Journal of Cloud Applications and Computing (IJCAC), 10(1), pp.64-84.

[20] Xiaoyu Li, Shaohua Tang, Lingling Xu, Huaqun Wang, and Jie Chen, “Two-Factor Data Access Control With Efficient Revocation for Multi-Authority Cloud Storage Systems”, IEEE Access, Volume 5, 2017.

Authors

R. Mangalagowri received a B.E. degree in Computer Science and Engineering from Madras University, India in 2004. And M.Tech degree from SRM University, India in 2011. She is currently pursuing a Ph.D. degree in Computer Science and Engineering at SRM IST, India. Her research interests include cloud security, Information security, and Web Services.

Revathi Venkataraman currently a Professor and Chairperson in the School of Computing SRMIST, India. She received her Ph.D. degree from **SRM** University (Currently SRMIST). Her research interests include Trust Computing, Cyber Security, Security enhancements and privacy considerations for IoT. She has received funding from Defence Research and Development Organization. She has also patented few of her innovative ideas in wireless networking.

How to cite this article:

R. Mangalagowri, Revathi Venkataraman, “Hypervisor Attack Detection Using Advanced Encryption Standard (HADAES) Algorithm on Cloud Data”, International Journal of Computer Networks and Applications (IJCNA), 9(5), PP: 555-567, 2022, DOI: 10.22247/ijcna/2022/215916.