IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

MACHINE LEARNING APPROACH TO ANOMALY DETECTION IN CLOUD INFRASTRUCTURE

[1]Dr Kusuma T, [2]Dr Jyothi S
[1]Associate Professor, [2]Associate professor
[1]Department of Computer Science and Engineering, [2]Department of Computer Science and Engineering,
[1]Kammavari Sangha Institute of Technology, Bangalore, India, [2] Global Academy of Technology, Bangalore, India

Abstract: The need for secure communication and data protection has become increasingly important in the digital age. As the use of digital technologies continues to grow, the need for secure communication and data protection also increases. Anomaly detection is a crucial aspect of data analytics that can identify suspicious behavior and detect malicious activities. This is particularly crucial in the cloud computing environment, where data is stored on multiple servers and accessed remotely by various users. The Local Pollination Grey Wolf Optimizer (LPGWO) is a global optimization algorithm that has been utilized in various applications. It is based on the concept of "cognitively guided exploration," which is a form of local search that utilizes an individual's experiences to direct the exploration. It has been successfully used to solve optimization problems in various fields, such as image processing, communication networks, and cryptography. The proposed algorithm is evaluated using various parameters, including the number of iterations, time complexity, and success rate. The performance of the proposed algorithm is then compared to existing RSA algorithms to determine its superiority. This paper presents a Local Pollination Grey Wolf Optimizer (LPGWO)-based RSA algorithm for anomaly detection in heterogeneous cloud data using machine learning techniques.

Index Terms **- LPGWO, Grey Wolf Optimizer, RSA algorithm, Cloud computing, Machine learning, Anomaly Detection.**

I. INTRODUCTION

In recent years, cloud computing has rapidly gained popularity due to its potential to offer numerous services and applications. The global cloud computing market is estimated to reach $331.2 billion by 2022 (Harvey, 2017). However, managing resources for heterogeneous cloud data poses a challenge for cloud providers. Heterogeneous cloud data refers to data that is stored in different formats, including structured, unstructured, and semi-structured data, and can be sourced from various origins. Jyothi S et. al. proposes the managing heterogeneous cloud data resources requires the ability to efficiently and effectively identify and allocate resources [14] [15] [16]. Anomaly detection is the process of identifying unusual behavior or patterns in a dataset. It is used to detect suspicious behavior and malicious activities. Anomalies can be divided into two categories: point anomalies and background anomalies. A point anomaly is an individual data point that significantly differs from the rest of the data points in the dataset. Text anomalies are patterns of data that are unusual in the context of the entire dataset.

Anomaly detection in cloud computing is a challenging problem due to data heterogeneity. Heterogeneous cloud data consists of various types of data, including text, images, and video. Additionally, scalability and reliability are important considerations when performing anomaly detection on large datasets. A single private server is utilized in a shared public cloud environment to streamline data management for any organization. When there is a need to decrypt secure sensor data, private servers are able to do so using the LPGWO-based RSA algorithm. In addition, the shared public server utilizes the Neutrosophic C-Means (NCM) clustering algorithm, which primarily focuses on data segmentation and anomaly classification, to conduct data clustering computations on encrypted data. The performance of the proposed study is evaluated in terms of accuracy, specificity, sensitivity, and runtime using four publicly available sensor datasets from Intel Laboratories.

1.1 Machine Learning Algorithms to secure resources in managing Cloud

Machine learning techniques have the ability to handle diverse cloud data resource management concerns. They can be used to efficiently and cost-effectively discover and allocate resources by analyzing existing data and choose the best resources for a given task. This can assist in guaranteeing that resources are deployed in an efficient and cost-effective way. Furthermore, machine learning methods can be utilized to ensure service quality. The techniques can be utilised to find anomalies in data and identify possible vulnerabilities.

This can assist in ensuring that the services offered are dependable and secure. Also, to ensure scalability and dependability, machine learning methods might be applied to ensure the quality of service. For example, machine learning algorithms can be used to predict demand for services and adjust resources accordingly. This can help ensure that resources are available when needed and that the services provided are of the highest quality. There are several advantages to using machine learning algorithms to manage resources in a heterogeneous cloud data environment. First, machine learning algorithms can help ensure that resources are allocated efficiently and cost-effectively. Second, machine learning algorithms can help ensure the quality of service by detecting anomalies in data and identifying potential security threats. Third, machine learning algorithms can help ensure scalability and reliability by predicting the demand for services and adjusting resources accordingly. Finally, machine learning algorithms can help reduce the time and effort required to manage resources.

1.2 Proposed Model

A proposed privacy model for scalable and reliable anomaly detection in heterogeneous cloud data using machine learning algorithms is based on the following principles: Data Pre-processing: The first step in data pre-processing is to clean, normalize, and discretize the data. This step is necessary to prepare the data for machine learning algorithms. Abnormality detection using machine learning algorithms: The data is preprocessed and then fed into machine learning algorithms to identify anomalies. Privacy Protection: This step is necessary to ensure that your data is protected from unauthorized access. This can be accomplished using cryptographic techniques such as encryption, hashing, and digital signatures. Scalability and Reliability: The proposed model should be able to scale to handle large datasets and be reliable enough to accurately detect anomalies.

II. FORMULATING AN LOCAL POLLINATION GREY WOLF OPTIMIZER (LPGWO) ALGORITHM

One of these algorithms for anomaly detection in cloud environments is the Local Gray Wolf Pollination Optimizer (LPGWO). This algorithm is an evolutionary optimization technique that uses crowd intelligence to find the optimal solution to a problem. This article describes the LPGWO algorithm, its components, and how to use it for anomaly detection in cloud environments.

2.1 Gray wolf optimization

Gray Wolf Optimization (GWO) is a meta-heuristic optimization algorithm inspired by the leadership hierarchy and hunting mechanisms of gray wolves in nature. It is an evolutionary algorithm based on the collective intelligence of wolves. This algorithm is designed to solve non-linear single-objective continuous optimization problems. The GWO algorithm follows the same principles of collective intelligence as a wolf. It is based on the gray wolf leadership hierarchy and hunting mechanics of the gray wolf. The collective intelligence of wolves is utilized to discover global optima for specific problems. This algorithm consists of three operators: leader, follower, and scout. Each of these operators has a distinct role and responsibility in the optimization process. Gray Wolf Optimization (GWO) is a meta-heuristic optimization algorithm that mimics the behavior behaviour of gray wolves. This program was developed by Mirjalili and Mirjalili in 2014 and is known for obtaining a global solution quickly.

GWO is a population-based global optimization method inspired by the behavior of gray wolves. This algorithm mimics the leadership hierarchy and cooperative hunting behavior of gray wolves in the wild. In the GWO algorithm, the population is represented by a group of wolves, including alpha, beta, and omega wolves, which explore the problem domain to find optimal solutions.

Alpha wolves are pack leaders and are responsible for scouting potential areas for solutions. Beta Wolf is the second-in-command and is tasked with exploring the area and refining the solution space. The omega wolf is the weakest wolf and is responsible for evaluating solutions and choosing the best one. The GWO algorithm works by moving the alpha wolf in the optimal direction.

Once a potential prey is located, wolves enter the pursuit phase. During this phase, wolves will chase their prey using their speed and agility. Wolves are capable of running for long distances and have an impressive ability to turn quickly and change direction, making it difficult for prey to escape.

The third phase is capture. During this phase, wolves will use their powerful jaws and sharp teeth to capture their prey. Wolves have the ability to quickly immobilize their prey, often within a matter of seconds.

Finally, the fourth phase is killing. During this phase, wolves will kill their prey by biting, crushing their neck, or suffocating them. This phase is often quick and efficient, allowing wolves to conserve energy and maximize their hunting success.

The following equation is used to update the distance between the grey wolf and its prey:

$$D = \left|C * X_p(t) - X(t)\right| \qquad (1)$$

where,

X, X_p : Coefficient vector

C : Current iteration as t

Equation 1 determines the coefficient vector C in the following way:

$$C = 2 * r_1 \qquad (2)$$

Let r_1 be a random vector with the interval (0, 1),(T=1,F=0). The prey location is identified by

$$X(t+1) = X_p(t) - A * D \qquad (3)$$

$$A = 2 * d(r_2 - d) \qquad (4)$$

where the range of t is (0, 1), r_2 (t) is a random vector and d (t) is a component which linearly decreases from 2 to 0 across the course of iterations.

The grey wolf position of α, β and δ is updated by

$$D_\alpha = |C_1 * X_\alpha - X| \quad (5)$$
$$D_\beta = |C_1 * X_\beta - X| \quad (6)$$
$$D_\delta = |C_1 * X_\delta - X| \quad (7)$$
$$X_1 = X_\alpha - A_1 * D_\alpha \quad (8)$$
$$X_2 = X_\beta - A_2 * D_\beta \quad (9)$$
$$X_3 = X_\delta - A_2 * D_\delta \quad (10)$$
$$X(t+1) = \frac{X_1(t) + X_2(t) + X_3(t)}{3} \quad (11)$$

2.2 LPGWO Working Steps

1. Initialization. This step involves randomly generating the initial positions of the wolves.
2. Intra-species Competition. In this step, each wolf assesses its fitness relative to the other wolves in the population. The wolves then update their positions based on the relative fitness.
3. Inter-species Competition. In this step, the wolves interact with each other in order to determine which of them will become the leader (alpha wolf), and which will become the followers (beta and delta wolves). The alpha wolf is the one with the highest fitness, while the beta and delta wolves are the ones with the second and third highest fitness, respectively.
4. Pollination. This step involves the alpha wolf exchanging its position with the beta and delta wolves. This exchange is done in order to encourage exploration of the search space.
5. Update. In this step, the positions of the wolves are updated based on the new position of the alpha wolf.
6. Termination. This step indicates that the algorithm has converged and the optimization is complete.

III. PRESERVING PRIVACY THROUGH A MODEL OF CLOUD COMPUTING

This section introduces models for anomaly detection that protect privacy and process data efficiently. By adopting a novel RSA scheme based on LPGWO, Privacy Manager solves the key leakage problem in symmetric key encryption technology. Data is encrypted before being processed by private and public servers in the cloud.

Figure 1 shows the key components of the proposed system model. The three main features of a private server are: (a) remote user access management in public cloud environments; (b) offloading analytical tasks to a shared public server; c) Performing encryption or decryption methods.

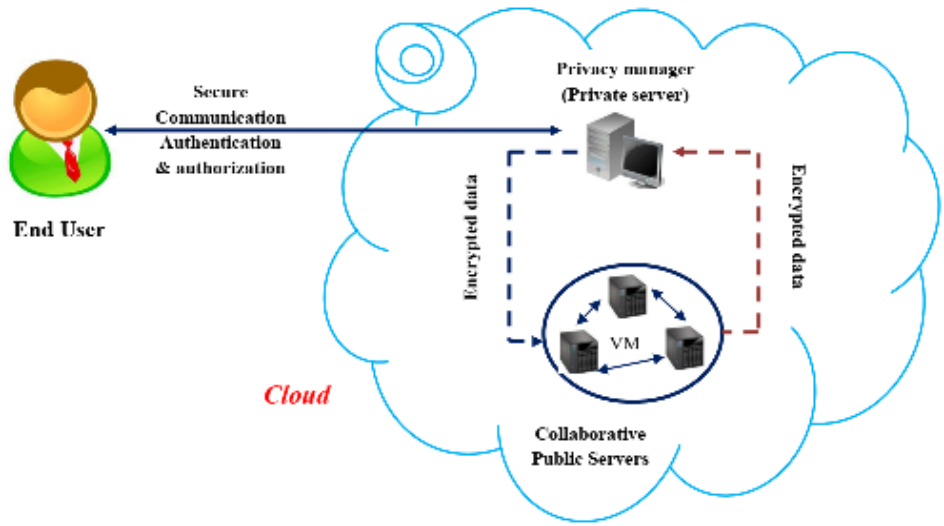

Fig.2: Proposed System Architecture

3.1 A RSA Scheme based on the LPGWO algorithm is proposed.

This paper presents an RSA scheme based on the Local Pollination Gray Wolf Optimizer for secure communication. RSA is one of the most widely used public key cryptographic algorithms and is employed in numerous applications and protocols. The Local Pollination Gray Wolf Optimizer (LPGWO) is a newly designed meta-heuristic algorithm that has been applied to various optimization problems. In this article, RSA is a public key cryptographic algorithm used for secure communication between two parties. It is used in various applications and protocols, such as digital signature, authentication, and key exchange. In RSA, communication security is based on the difficulty of factoring large integers. The security of the RSA algorithm relies on the selection of RSA parameters, including the modulus (N), public exponent (e), and private exponent (d). The security of the RSA algorithm depends on the selection of these parameters. Therefore, it is important to choose parameters that are difficult to factor and provide secure communication.

3.2 Encryption and decryption process

The LPGWO based RSA algorithm is used to generate the optimal key, which in turn is employed for encrypting and decrypting the original plain text into ciphertext and from ciphertext back into plain text, respectively.

Encryption:

$$C = P^{e\_optimal} \, mod \, n \qquad (12)$$

Using the computed private key (d,n), the receiver deciphers the ciphertext

Decryption

$$P = C^{d} \, mod \, n \qquad (13)$$

III. ANOMALY DETECTION

Anomaly detection is the process of identifying patterns in data that deviate from normal behavior. The goal is to identify and alert on activities that do not follow expected behavior. This could include suspicious user activity, unusual network activity, or unexpected system behavior. Anomaly detection can also be used to identify and investigate potential security threats such as malware or malicious agents trying to access your resources. The purpose of anomaly detection is to identify patterns in data that appear uniform and deviate from the norm. Anomaly detection is an important way to identify fraud, network intrusions, and other rare events that can have significant impact but are difficult to detect. Machine learning is increasingly used to automatically detect anomalies.

This approach uses C-means clustering algorithm for anomaly detection. Clustering is a machine learning technique that, given a set of data points, involves grouping those points into separate clusters. Data points in the same cluster should have similar features and/or characteristics, while data points in different clusters should have very different features and/or characteristics. This unsupervised learning method is used to group data points with similar characteristics.

4.1 NCM algorithm

Neutrosophic C-means (NCM) is an extension of the fuzzy C-means (FCM) algorithm, which is designed to handle the presence of noise and outliers in data. NCM is a powerful clustering technique that can effectively handle data with imprecise and uncertain values. NCM works by assigning a membership grade to each point in the dataset, which is based on the probability of belonging to a certain cluster. This allows NCM to provide more accurate clustering results even in the presence of noise and outliers. The virtual machine tier processes sensor data in parallel by performing NCM clustering.

Mid-level virtual machines access the clustering output from the previous level and perform data fusion to detect anomalies.

This makes it easier to detect high-level anomalies. Correlation-based analysis of all leaf nodes is performed by anomalous nodes in the VM hierarchy. Here are the steps to implement NCM clustering:

We form a new membership and objective function to perform NCM clustering that considers both definite and indeterminate cluster degrees, as shown in Equation 14

$$J_{NCM}(\mathrm{T}, \mathrm{I}, F, c) = \sum_{a=1}^{M}\sum_{b=1}^{c}(\beta_1 T_{ab})^n \|u_a - c_b\|^2 + \sum_{a=1}^{M}(\beta_2 I_a)^n \|u_a - \bar{c}_{a\,max}\|^2 + \delta^2 \sum_{a=1}^{M}\beta_3 F_a \qquad (14)$$

where

β_i=Weight factor

u_a (a =1,2,…,M) = data set

c_b =cluster center

c (2≤ c ≤N) = required number of classes

δ = to check the number of objects considered as outliers

n = constant, which is any real number greater than 1

T_{ab}, I_a, F_a = membership values belonging to certain clusters, boundary regions and noise data set. , which satisfy equation 14

V RESULTS AND DISCUSSION

In this section, analyzes are performed to check the performance, efficiency and accuracy of the proposed model in detecting anomalies. Experiments are performed in the Amazon cloud environment using a system configuration of two Intel Xeon E4 processors with 4 cores, 12 GB of memory and a speed of 2.6 GHz. The model is implemented using MATLAB software. The performance of the model is evaluated using different sensor data sets from Intel Labs. These datasets contain actual sensor node values including temperature, humidity, light and voltage values obtained from 54 sensors located at Intel Berkeley Lab. Datasets 1 and 2 contain no anomalies, while datasets 3 and 4 contain 50 anomalies. As the number of clusters increases, the complexity of the anomaly detection process also increases.

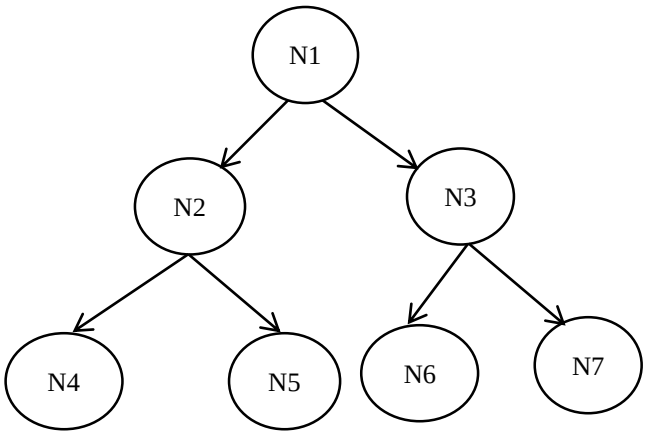

Fig.2: Detecting Anomalies in Collaborative Virtual Machines Hierarchy

Figure 2: illustrates the VM hierarchy of three distinct tiers with data processing nodes where datasets 1 to 4 are supplied to nodes N4 to N6 respectively. Tables 1 to 4 demonstrate the accuracy rate of detecting anomalies for plaintext (P_t)and ciphertext (C_t) data based on Cluster Size (CS).

Table 1 :Collaborative Virtual Machines hierarchy in detecting anomalies(T=1,F=0)

CS	$S_{EN}(P_t)$	$S_{EN}(C_t)$	$S_{PEC}(P_t)$	$S_{PEC}(C_t)$	$FP(P_t)$	$FP(C_t)$	$FN(P_t)$	$FN(C_t)$
1	N-AN	N-AN	T	T	F	F	F	F
2	N-AN	N-AN	T	T	F	F	F	F
3	N-AN	N-AN	T	T	F	F	F	F
4	N-AN	N-AN	T	T	F	F	F	F
5	N-AN	N-AN	T	T	F	F	F	F
6	N-AN	N-AN	T	T	F	F	F	F

The classification accuracy and performance of the model were evaluated using datasets 3 and 4 with anomalies. For node 6, the average values of sensitivity and specificity obtained for the plain text version were 0.90 and 0.996, respectively, and the average values of sensitivity and specificity obtained for the encrypted text version were 0.96 and 0.99, respectively. For node 7, the average sensitivity and specificity values obtained for the plain text version were 1 and 0.99, respectively, and the average sensitivity and specificity values obtained for the encrypted text version were 1 and 0.9, respectively.

Table 2: The classification accuracy obtained at nodes N4 and N5 with datasets 1 and 2

CS	$S_{EN}(P_t)$	$S_{EN}(C_t)$	$S_{PEC}(P_t)$	$S_{PEC}(C_t)$	$FP(P_t)$	$FP(C_t)$	$FN(P_t)$	$FN(C_t)$
1	0.868	N-AN	T	T	F	F	F	F
2	0.884	N-AN	T	T	F	F	F	F
3	0.919	N-AN	T	T	F	F	F	F
4	0.928	N-AN	T	T	F	F	F	F
5	0.936	N-AN	T	T	F	F	F	F
6	0.895	N-AN	T	T	F	F	F	F

Table 3: The final classification accuracy obtained at node N1

CS	$S_{EN}(P_t)$	$S_{EN}(C_t)$	$S_{PEC}(P_t)$	$S_{PEC}(C_t)$	$FP(P_t)$	$FP(C_t)$	$FN(P_t)$	$FN(C_t)$
1	0.927	0.937	0.989	0.997	188	171	13	13
2	0.957	0.951	0.995	0.999	13	12	9	7
3	0.957	0.957	0.993	0.998	24	19	7	8
4	0.961	0.957	0.996	0.992	31	33	8	7
5	0.942	0.961	0.997	0.993	32	30	8	8
6	0.942	0.932	0.998	0.998	4	8	12	10

The encrypted version achieves higher accuracy thanks to sharing the optimal public key generated by LPGWO-based RSA algorithm about data encryption in plain text format. An optimal public key prevents hackers from accessing information during data processing on public servers. Therefore, the version of the ciphertext obtained by using the optimal encryption the public key of the proposed optimal cryptographic scheme achieves more satisfactory performance compared to the original plain text version private server.

5.1 Performance evaluation

The performance of the model is measured in terms of implementation time, number of VMs required, encryption time and decryption time.

Table 3: Average execution time required for different dataset sizes and local level clustering

Number of clusters	# Datapoints	Avg. execution time (ms)
1	1400	243.1
	2800	500.9
	4200	1050.4
2	1400	280.9
	2800	550.1
	4200	1145.9
3	1400	300.3
	2800	618.3
	4200	1239.2
4	1400	350.4
	2800	690.0
	4200	1308.2
5	1400	411.1
	2800	700.6
	4200	1270.2
6	1400	472.40
	2800	730.9
	4200	1325.1

Data processing is usually centralized at the local level of the VM hierarchy. Table 3 shows the average execution time of the local level of the hierarchy required to complete the task of anomaly detection and data clustering as the amount of data increases. The size of clusters varies from 1 to 6. To detect anomalies, construct clusters using three sets of data points.

The proposed method distributes the responsibilities of anomaly detection and clustering to each local node in a larger set of virtual machines to reduce the execution time. By increasing the number of virtual machines at each data processing level, the execution time decreases. Fig 2 shows that as the number of virtual machines increases for the expected number of clusters, the execution time decreases. The numbers 0, 1, 2, 3 ... on the x-axis indicate the number of virtual machines used. The graph shows the execution time for different cluster sizes with different number of virtual machines**.**

Table4: The average time required to complete clustering and anomaly detection jobs.

Number of clusters	#Datapoints	Avg. execution time (ms)
1	1400	545
	2800	800
	4200	578
2	1400	545
	2800	800
	4200	1410
3	1400	578
	2800	874
	4200	1486
4	1400	634
	2800	834
	4200	1539
5	1400	682
	2800	940
	4200	1603
6	1400	728
	2800	1014
	4200	1670

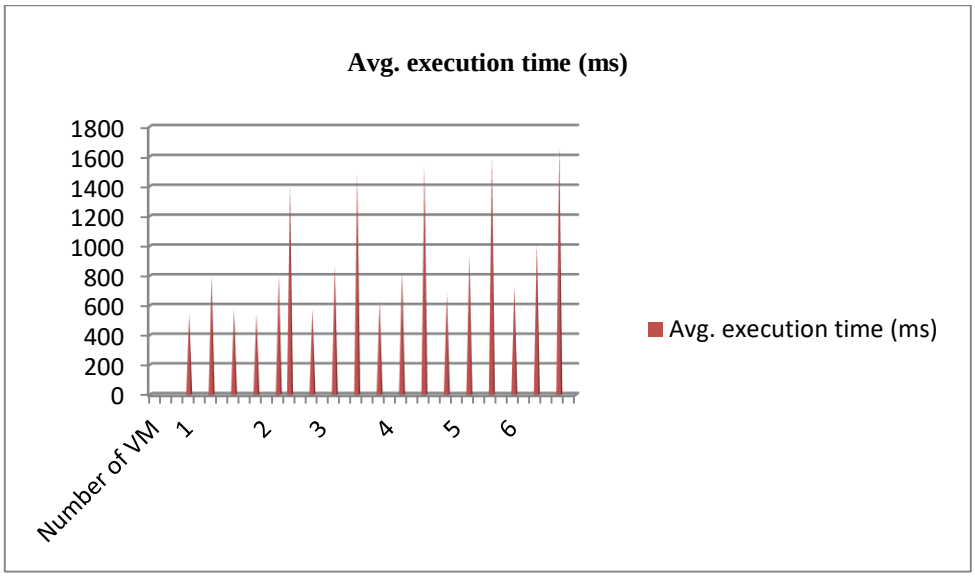

Fig 3: Number of virtual machines vs. average execution time

CONCLUSION

A newly designed cryptographic system is optimized for large-scale sensor data to provide an efficient cloud-based anomaly detection model with privacy-preserving techniques. A new RSA technology based on LPGWO is used to encrypt the data. In cloud environments, public servers process data using encrypted data. The NCM clustering algorithm is used by the joint release server for data segmentation and anomaly classification.

Machine learning algorithms involve automatically extracting information from data. This develops the body's ability to perform data analysis from care and carelessness, laying the foundation for identifying patterns, adequate print content, and conflicts that are beneficial to the intelligent distribution of data and

maintaining information privacy and security. The results of this study served as support for the development of new methods for detecting anomalies using machine learning. The proposed method is implemented and validated using four publicly available datasets from Intel laboratories. The validation results show that the framework is effective in protecting data privacy and detecting anomalies with high accuracy. Some of the most important anomaly detection programs in use today

REFERENCES

[1] Bharathkumar Ramachandra, Michael J Jones & Ranga Raju Vatsavai 2020, 'A Survey of Single-Scene Video Anomaly Detection', IEEE, Computer Vision and Pattern Recognition, arXiv preprint arXiv:2004. 05993, 2020 - arxiv.org.

[2] Chen Wei, Zhang Yu-fang & Xiong Zhong-yang 2010, 'Research and realization of the load balancing algorithm for heterogeneous cluster with dynamic feedback', Journal of Chongqing University, vol.33, no. 2, pp.2–14.

[3] Jinming Zhang 2019, 'Detection of Network Protection Security Vulnerability Intrusion Based on Data εining', International Journal of Network Security, vol.21, no.6, pp.979-984.

[4] Mehdi Mohammadi, Ala Al-Fuqaha, Sameh Sorour & Mohsen Guizani 2017, 'Deep learning for IoT big data and streaming analytics: A survey', arXiv preprint arXiv:1712.04301.

[5] Qun Wei, Guangli Xu & Yuling δi 2011, 'Research on cluster and load balance based on Linux virtual server', in Proc. International Computing Applications, vol.105, pp. 169–176.

[6] Shagufta εehnaz & Elisa Bertino 2020, 'Privacy-preserving Real-time Anomaly Detection Using Edge Computing', IEEE 36th International Conference on Data Engineering (ICDE), DOI 10.1109/ICDE48307 2020.00047.

[7] Seyedali Mirjalili, Shahrzad Saremi, Seyed Mohammad Mirjalili & Leandro dos S. Coelho 2016, Multi-objective grey wolf optimizer:A novel algorithm for multi-criterion optimization, Expert Syst. Appl. vol.47, pp.106–119.

[8] Yanhui Guo & Abdulkadir Sengur 2015, NCM: Neutrosophic c-means clustering algorithm, Pattern Recognit. vol.48, pp.2710–2724.

[9] Zhangjie Fu, Xinle Wu, Chaowen Guan & Xingming Sun 2016, Toward efficient multi-keyword fuzzy search over encrypted outsourced data with accuracy improvement, IEEE Trans. Inf. Forens. Secur. vol.11, pp.2706–2716.

[10] Yanhui Guo & Abdulkadir Sengur 2015, NCM: Neutrosophic c-means clustering algorithm, Pattern Recognit. vol.48, pp.2710–2724.

[11] Tao Wang, Jiwei Xu, Wenbo Zhang, Zeyu Gu & Hua Zhong 2017, 'Self-adaptive cloud monitoring with online anomaly detection', Future Generation Computer Systems, vol. 80,pp.89–101.

[12] Mingjian Cui, Jianhui Wang & εeng Yue 2019 ,"εachine δearning Based Anomaly Detection for δoad Forecasting Under Cyberattacks", IEEE Transactions on Smart Grid, vol.10, no. 5, pp. 5724 – 5734.

[13] Abdulatif Alabdulatif, Ibrahim Khalil, Heshan Kumarage, Albert Y. Zomaya & Xun Yi 2017, 'Privacy-preserving anomaly detection in the cloud for quality assured decision-making in smart cities', J. Parallel Distrib. Comput., vol.127, pp.209–223.

[14] Jyothi S , Dr. Shylaja B S, 2020, Efficient Approach for Resource Provisioning to Manage Workload in Cloud Environment, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) Volume 09, Issue 06 (June 2020),

[15] Sowmya, M. R., and S. Jyothi. "A Framework for Dynamic Relocation of Cloud Services." International Journal of New Innovations in Engineering and Technology 3.2.