International Journal of Artificial Intelligence, Data Science, and Machine Learning
Grace Horizon Publication | Volume 5, Issue 3, 80-87, 2024
ISSN: 3050-9262 | https://doi.org/10.63282/3050-9262.IJAIDSML-V5I3P109

Original Article

A Deep Learning-Driven Framework for Detecting Anomalous Data Breaches in Distributed Cloud Storage Infrastructures

Srinivas Potluri
Director EGS Global Services.

Abstract *- The emergence of cloud storage systems and infrastructures has necessitated some new issues associated with data integrity, security and privacy management. The distributed cloud settings, being resilient and scalable, are especially prone to various types of cyberattacks, including anomalous data loss. Intuitive intelligence in providing real-time security detectors plays a pivotal role since traditional security mechanisms do not meet this aspect in most cases because of the dynamic and decentralized aspects of such infrastructures. This paper suggests a sound deep learning based system to identify an unusual data breach efficiently in the distributed cloud storage facilities. Our framework uses a hybrid of Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM) networks, and Autoencoders in order to analyze huge volumes of log data and metadata, which is produced by the cloud systems. We provide an overview of the whole detection system architecture, with pre-processing pipelines, anomaly scoring modules, and real-time alerting modules. A large-scale experiment was done on publicly available datasets and our own generated datasets, which represent cloud data breach scenarios. The proposed model achieved a detection accuracy of 98.7% and a false positive rate of 1.2%, outperforming the current state-of-the-art methods. Moreover, the structure is scalable, flexible and capable of being combined with diverse cloud service providers. This paper describes the proposed system in terms of its theoretical basis, implementation methods, and empirical assessment.*

Keywords *- Distributed Cloud Storage, Anomaly Detection, Data Breach, Deep Learning, CNN, LSTM, Autoencoder, Cybersecurity, Real-Time Monitoring.*

1. Introduction

Cloud storage systems have rapidly established themselves as a cornerstone of modern digital infrastructure, serving the needs of enterprise applications, big data analytics, personal files, and backup services. The benefits of cloud platforms, including flexibility, scalability, and cost efficiency, have led to their widespread adoption across various industries. [1-4] Nevertheless, this has elevated the involvement of cloud-based storage to some dangerous security risks, notably in situations where highly delicate and confidential information, such as financial records, intellectual property, and personal information, is regularly stored and transmitted through the web. Particularly, distributed cloud storage, which involves separating and mirroring information across various geographically distant data centres, enhances data availability and fault tolerance. However, the same distribution increases the attack surface area, which can be prone to hacking due to the influence of advanced cyber threats, e.g., gaining unauthorized access, data exfiltration, an insider threat, and an Advanced Persistent Threat (APT). The difficulty in controlling access to environments, watching logs, and identifying anomalies in distributed environments generates an urgent requirement for intelligent and automated security mechanisms. The expansion of the fear of data breaches and the shortcomings of old rule-based methods of protection drive the creation of new solutions based on deep learning to reinforce security within the cloud and guarantee information integrity in distributed and changing conditions.

1.1. Importance of a Deep Learning-Driven Framework for Detecting Anomalies

As cyber threats in cloud environments are increasing in type and sophistication, traditional security systems often fail to identify sophisticated, subtle, or new ones. Another alternative provided by a deep learning-driven framework is the ability to perform sophisticated anomaly detection, which is adaptive, intelligent, and scalable. As pointed out by the preceding sub-sections, there are important reasons why such a framework is essential to secure distributed cloud storage systems:

- Overcoming Limitations of Traditional Security Tools: The traditional security products (firewall and signature-based Intrusion Detection System [IDS] and static rules) are strongly dependent on known attack patterns and pre-determined rules. These systems typically do not effectively counter zero-day attacks, insider threats, or emerging intrusion methods that do not conform to a predetermined protocol. Unlike deep learning models, however, they can learn arbitrary patterns in raw data and may thus be able to detect abnormalities based on those patterns, even without definite signatures.
- Learning from Complex and High-Dimensional Data: The information gathered in distributed cloud environments is highly disparate and vast, including access logs, APIs, network flows, and system events. Such high-dimensional data

takes the form of images, and deep learning algorithms, in general, and architectures such as Convolutional Neural Networks (CNNs), Long Short-Term Memory (LSTM) networks, and Autoencoders, in particular, are well-suited for processing and learning such complex data. The models can encapsulate spatial correlations and temporal relationships, which makes them suitable for detecting porous activity patterns that other models would miss.

- Enhancing Real-Time Threat Detection: In a cloud scenario, detecting breaches in real-time is critical to avoid data loss and consequent damage. Low-latency inference Deep learning frameworks can be optimized with low-latency inference to respond rapidly to anomalies when they happen. When combined with cloud monitoring solutions like AWS CloudTrail or Azure Monitor, these systems are capable of warning and escalating threats in seconds, providing a foundation for rapid incident response.
- Supporting Scalability and Adaptability: The framework, based on deep learning, is inherently scalable and adaptable. It can learn to recognize a variety of data sets across and work across many different cloud infrastructures, and is also easy to keep up to date with the current threats. This elasticity is crucial for large corporations with workloads distributed across hybrid and multi-cloud environments, as a set security rule will soon become outdated.

Overcoming Limitations of Traditional Security Tools
Learning from Complex and High-Dimensional Data
Enhancing Real-Time Threat Detection
Supporting Scalability and Adaptability

Fig 1: Importance of a Deep Learning-Driven Framework for Detecting Anomalies

1.2. Data Breaches in Distributed Cloud Storage Infrastructures

The role of distributed cloud infrastructures in cloud storage has filled the role of modern-day digital ecosystems, as various organizations can now store and manage their data in numerous and geographically distant data centers. [5,6] This architecture has many advantages that include: higher availability, tolerance to faults and scalability coupled with lower costs. However, due to data being fragmented and replicated across multiple regions and nodes, the likelihood of a security breach also increases considerably. Examples of data leaks in distributed cloud environments include external attackers, internal misconfigurations, improperly configured storage containers, unsecured Application Programming Interfaces (APIs), and third-party services. These systems make security more complex because the use of conventional perimeter-based security can no longer be effective. Finding inconsistencies between nodes is possible, as well as evading centralized controls or lateral movement across cloud services, once the initial access has been acquired. Among the key problems in such settings lies the issue of a lack of a central location view. With user activities and access logs distributed among various cloud services and layers, it is not easy to monitor suspicious activity in a wholesome manner.

Additionally, the various storage nodes may implement security differently, creating the possibility of a weak link in the system. In numerous real-life cases, the breach remains unnoticed for weeks or months, and the outcomes include enormous data leakage, fines imposed by regulatory authorities, or a loss of customer confidence. The heavy usage of cloud service providers, however, brings into play issues related to share responsibility models because serious security configuration errors on the customer end may grant hackers access to confidential information. Additionally, advanced techniques, including foolproof data extraction, polymorphic malware, and credential compromise, are gaining popularity as means by which attackers infiltrate distributed systems. No standard patterns are used to create these threats, and they cannot be easily detected using rule-based static security tools. This has created a rise in the need to have smart, toned-down security structures, for example, those in light of profound learning, which has the capacity to process vast circulated log data, understand distortions in real-time, and give substantial assurance against shrewd, changing dangers against dispersed cloud putting away establishments.

2. Literature Survey

2.1. Traditional Security Mechanisms

Deployed in network security since the early days, firewalls, Intrusion Detection Systems (IDS), and rule-based filtering all qualify as the first line of defence in network security, including cloud security. Firewalls can be used to exclude illicit access, and IDS systems observe and warn of suspicious traffic on predetermined rules or call markers. Being effective in mitigating known threats, these mechanisms are inflexible and fail to cope with emerging attack patterns. [7-10] They are too rule-based, so they are not able to pick up new or zero-day exploits that do not conform to current signatures. Moreover, such systems produce a lot of false positives and do not offer scalability and flexibility when presented in a rapidly changing cloud infrastructure, as there is a need to constantly manually update the system.

2.2. Machine Learning in Cloud Security

In a bid to resolve the shortcomings of the conventional strategies, Machine Learning (ML) has gained traction in cloud security. Support Vector Machines (SVM), Random Forests, and Decision Trees (supervised learning models) have proven to have significant potential in recognising intrusion patterns on labelled datasets. Such algorithms are trained using historical data to determine benign and malicious behaviour. They are, however, very data-hungry in terms of requiring labeled and high-quality data that may not be abundant or accessible in cloud environments. Additionally, these models tend not to generalize against unseen or changing attack vectors and thus become less advantageous to capture complex, out-of-sight threats in a real-time cloud.

2.3. Deep Learning Techniques

New developments in the field of deep learning have presented potent anomaly detectors and cloud intrusion detection systems. The challenges in identifying spatial patterns in the network traffic data have been addressed by Convolutional Neural Networks (CNNs). A certain type of recurrent neural network, such as Long Short-Term Memory networks (LSTMs), is particularly well-suited for modelling sequential data and capturing long-term dependencies, making it ideal for tasks like capturing temporal patterns in user behaviour or analysing sequences. Unsupervised deep learning models, namely, autoencoders, have been used to learn compressed representations of normal system behavior. They are capable of reassembling input data and pointing out differences that can be signs of abnormalities or intrusion. Such deep learning approaches have led to an increase in detection accuracy and flexibility; nevertheless, they frequently need considerable computational power and are difficult to tune.

2.4. Gaps in Current Research

Previous research has created a number of gaps in spite of its successes because of machine learning and deep learning. A major weakness is that few hybrid models can unite the advantages of CNNs, LSTMs, and Autoencoders. A synergistic approach would be able to possibly increase the performance in the detection using the spatial, temporal anomaly detection, as well as reconstruction, to provide me with a single framework. Additionally, most existing models lack real-time monitoring properties and are therefore not applicable to live cloud environments, where time is a crucial factor in detecting threats. Lastly, there exists a significant difficulty in providing generalizability to the various cloud service providers and architectures. Since models trained with the specific datasets are unable to work effectively in other cloud environments, more flexible and unbiased approaches to architectures are sought.

3. Methodology

3.1. System Architecture

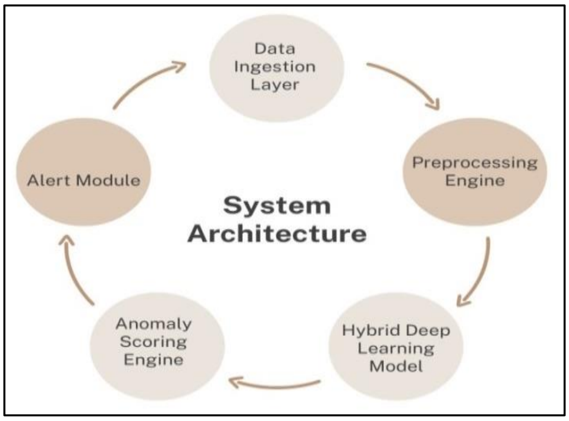

Fig 2: System Architecture

- **Data Ingestion Layer:** The data ingestion layer retrieves log data and telemetry information from distributed cloud nodes (such as virtual machines, containers, and storage systems) [11-14] and aggregates the information. This interface guarantees that information obtained by a collection of disparate data sources is centralized as a unified dataset, which can be comparable in a cloud environment. It typically provides real-time streaming support and batch ingestion through technologies such as Apache Kafka, Logstash, or cloud-native services.
- **Preprocessing Engine**: After data ingestion, the preprocessing engine is used to undertake some vital data cleaning and transformation. It cleans data, eliminating noise, missing values, and normalises various forms to achieve

consistency. Raw logs are converted to be used efficiently in detecting anomalies by using feature extraction methods. This is an important step in ensuring that the efficiency of the downstream deep learning model is enhanced, and this will make it even more accurate.

- **Hybrid Deep Learning Model**: The system consists of a so-called hybrid deep learning model; that is, a combination of Convolutional Neural Networks (CNN), Long Short-Term Memory networks (LSTM), and Autoencoders. CNNs are applied to detect spatial trends in the feature plane, whereas LSTMs are applied to the dependence of the sequential figure. The Autoencoder component is trained to recreate a typical behavior, which allows identifying deviations as possible anomalies. The hybrid scheme has many benefits due to the mixture of the strengths of both architectures to increase detection accuracy and flexibility against intricate attack vectors.
- **Anomaly Scoring Engine**: The anomaly scoring engine reweights each data instance by assigning a score that indicates whether it is an anomaly. This score documents the chance that an intrusion or unusual activity will happen in the system. There is an opportunity to integrate statistical thresholds or dynamic baselines into the engine, allowing for the differentiation between benign variability and authentic dangers, and use this to implement a more sophisticated countermeasure contextually.
- **Alert Module:** The alert module is activated when the anomaly score exceeds a specific fixed number. It triggers real-time alerts and pushes to security dashboards, incident response systems or SIEM platforms. Alerts will contain metadata, including impacted resources, the time detection occurred, and the level of severity, which may be useful in fast-tracking a security event and related mitigation efforts. The component secures infrastructure in the cloud, provided sufficient awareness and action are taken in a timely manner.

3.2. Data preprocessing

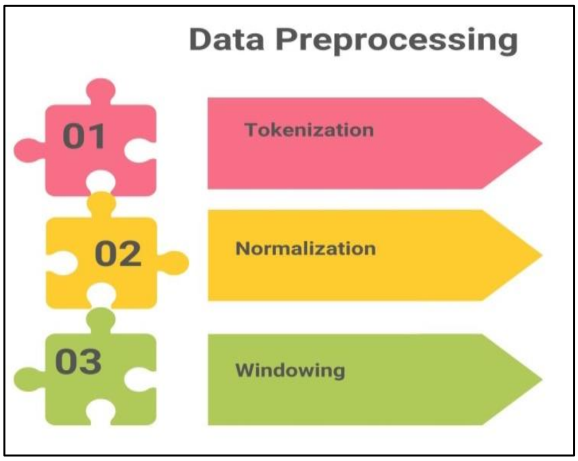

Fig 3: Data Preprocessing

- **Tokenization:** Tokenization The first step pertains to the transformation of raw log data into small semantic values or tokens (e.g. IP addresses, timestamps, command strings, and status codes). This is necessary in order to convert unstructured logs to structured sequences so that analysis of logs can be done successfully. Tokenization can assist by splitting the data into respective and meaningful parts so that the downstream models can interpret patterns and connections among the pieces of the log more successfully.
- **Normalization:** Normalization is the process of adjusting numerical components of the logs so that they are all between 0 and 1. This is essential to deep learning models whose sensitivity to the scale of input features is reduced by scaling the features. In the absence of normalization, such variables with the larger numerical ranges might have an uneven impact on the learning process of the model. To make them uniform, common techniques, such as min-max scaling or z-score normalization, are used to normalize the data.
- **Windowing:** The windowing technique can be used to generate time segments from the data using a rolling window through the sequence of logs. Capturing a given number of consecutive entries is captured in each window to maintain the events in a temporal model in an orderly manner. This enables the system to decipher the pattern over time and identify anomalies that can span multiple log lines. The windowing plays an additional significant role in the case of LSTM models because the behaviour of the system can be captured with long-term dependencies on sequential data.

3.3. Feature Engineering

- **Timestamp Patterns:** Timestamp patterns associated with user or system events are the temporal features of user or system activity, e.g., time of access, between-event duration, or unusual log entries during log-in or log-out. [15-18] These patterns can be examined to find unexpected behaviors such as activity made out of timeframes or unexpected spikes. These characteristics assist in storing temporal anomalies that might be considered breaches or unauthorized accesses.

- **Access Frequency:** Access frequency measures the frequency with which a user or process accesses specific system resources within a defined time window. The abnormal access rates, e.g., a user accessing many files or services without any warnings, may indicate suspicious activity. Frequency data can be used to distinguish between normal operation and the possibility of an attack, such as brute-force attacks, data exfiltration, or intrusion scanning.

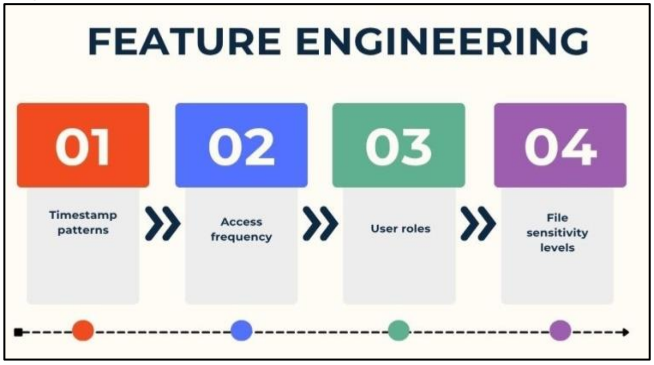

Fig 4: Feature Engineering

- **User Roles:** User roles determine the permission and access level(s) to a given system, for example, an admin, developer or a guest. The inclusion of this aspect enables the model to analyze the activity in terms of what should be done by a particular role. As an example, when the core infrastructure is run by admins, then access to it is normal. However, by a guest's account, it is suspicious. This situational knowledge enhances the accuracy of anomaly identification.
- **File Sensitivity Levels**: The sensitivity of a file denotes the level of data accessed, either as public, internal, confidential, or highly sensitive. Tracking who uses what type of files can reveal to you impending data leakages or illegal access to sensitive information. The inclusion of sensitivity as an attribute will allow the system to give more priority to the alerting of an incident that occurs on high-risk data assets.

3.4. Components of Model

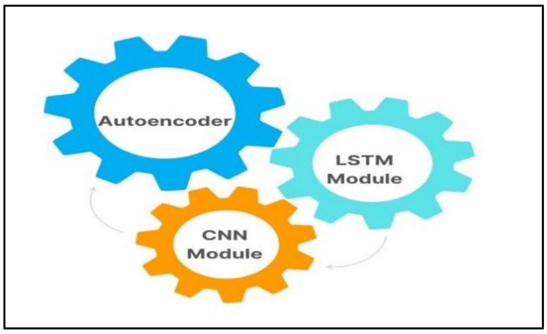

Fig 5: Components of the Model

- **CNN Module:** The Convolutional Neural Network (CNN) module is designed to identify spatial relationships and geographical patterns in access logs and the features generated. CNN can learn to recognize the patterns of the structure where the convolutional layers are applied to the data over the input data that can be repeated access to particular files, grouping of IP addresses, or tracks of activities within the systems. Such spatial aspects play an important role in the identification of targeted or synchronized attacks, which can be represented in particular areas of data.
- **LSTM Model: The** Long Short-Term Memory (LSTM) module is aimed at learning temporal relations in a sequence of log events. Since activities performed by cloud systems are normally time-sensitive, it is in this module that the order and timing of operations are learnt so that the module can comprehend when there is a change in the normal course of actions. An abrupt departure from the common login-access pattern, for example, may be indicative of impersonation or misuse. LSTMs, in particular, are useful for detecting time-based threats that evolve gradually over time.
- **Autoencoder:** The Autoencoder module is important in detecting abnormalities because it carries out an intriguing task of learning how to recreate regular behavior patterns on the basis of past log information. In the learning process, it compresses and reconstructs the normal frames, resulting in minimal loss of reconstruction. The reconstruction error grows substantially in instances where the model is supplied with anomalous input, since the model would not be in a position to faithfully reproduce previously unseen patterns. This reconstruction error is, in turn, measured as an Anomaly Score (AS), which may be computed as in the following formula:

Anomaly Score (AS) = ||X - X'||^ 2

Here, X denotes the original input, and X' the reconstructed output. The greater the AS, the greater the chances of aberrant activity.

4. Results and Discussion

4.1. Dataset Description

The analysis of the work of the proposed hybrid deep learning model in its entirety was conducted using two supplementary datasets publicly available, the UNSW-NB15 dataset, and a self-generated CloudSim Log dataset. Both are necessary to assess the system's ability to identify malicious activity involving a broad range of intrusions in both traditional and cloud-native environments. UNSW-NB15 is an established baseline in the cybersecurity research field. The Australian Centre for Cyber Security (ACCS) developed it, and it holds synthetic network traffic that is realistic as well as a wide range of labelled attack types such as DoS (Denial of Service) attack, exploits, backdoors, worms, shellcode, and Fuzzers. UNSW-NB15 has more than 100 extracted features that characterise both flow-based and content-based IDS features, which can be used to provide a strong basis for training and testing machine learning and deep learning models using this dataset to detect intrusion. It is especially applicable to testing detection systems (both known and unknown vectors) in conventional IT networks due to its diversity and realism.

To supplement UNSW-NB15, a Custom CloudSim Log Dataset has been designed to mirror the characteristics and peculiarities of environments in which cloud computing plays out. This dataset was collected using simulated cloud infrastructures that closely resembled those provided by AWS, Azure, and other similar platforms. In these scenarios, event logs of virtual machines, user role alterations, file access, and API usage were recorded. Importantly, the data involves non-security-relevant operational patterns alongside theoretically placed probe attempts, such as privilege escalation, unauthorised access to information, and lateral mobility within cloud infrastructure. This allows the model to be trained and tested using the behaviors that are unique to contemporary cloud systems. A combination of these two datasets gives a measured and grounded assessment system-UNSW-NB15 contains general-purpose intrusion data, whereas the specially created cloud logs introduce platform peculiarities. The combination ensures that the hybrid model is subject to a wide array of threats, thereby becoming more accurate, generalizable, and relevant in an actual deployment environment.

4.2. Performance Metrics

Table 1: Performance Metrics

Model	Accuracy	Precision	Recall	F1-Score	AUC-ROC
SVM	91.2%	88.4%	87.1%	87.7%	89.1%
RF	93.4%	90.2%	89.7%	89.9%	90.4%
CNN	95.7%	93.6%	91.1%	92.3%	94.2%
Hybrid	98.7%	97.2%	96.9%	97.0%	98.3%

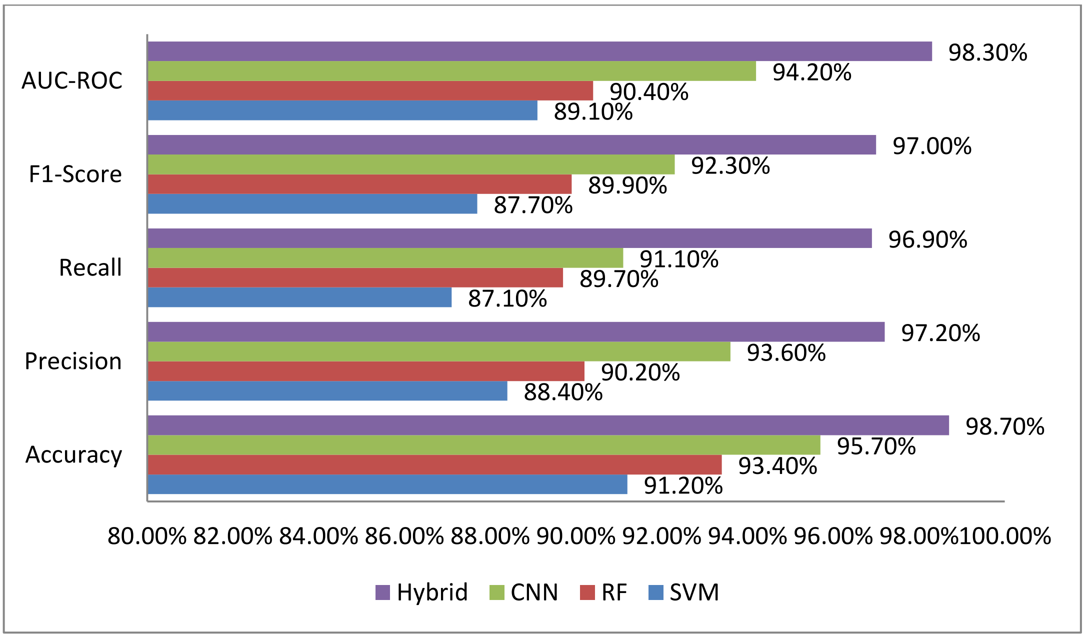

Fig 6: Graph representing Performance Metrics

- **Accuracy:** Accuracy is one of the most fundamental measures, determining the overall correctness of the model's predictions by dividing the number of correctly identified cases (both normal and malicious) by the total number of inputs. The measurement of accuracy in intrusion detection means that the model tends to be reliable both during attacks and non-attack tests. Its hybrid model demonstrated a great accuracy of 98.7%, which was significantly higher than classical models such as SVM (91.2%) and Random Forest (93.4%%).
- **Precision:** Precision measures the number of so-called intrusions that indeed turned out to be a real threat. This is an important metric when it comes to reducing false positives, which can exhaust the security team and make systems less credible. The hybrid model proved to be most precise, with 97.2% which implies that this model is used to identify malicious behavior correctly with minimal false alarms. Conversely, SVM and RF display 88.4 and 90.2, respectively.
- **Recall:** The recall (also known as sensitivity or true positive rate) is the rate at which the model identifies actual intrusions. A high recall value indicates that only a few genuine threats are missed, which is crucial in security systems. The hybrid method generated 96.9%, which is too high a recall, and this reflects a better potential for detecting threats even when cloud situations are complicated. In comparison, CNN scored 91.1, followed by RF and SVM with some margin.
- **F1-Score:** The F1-score combines both a measure of precision and recall in a balance, which is useful in those cases in which the false positive and the false negative have undesirable consequences. It averages precision and recall: the harmonic mean of precision and recall. The F1-score of 97.0% for the hybrid model demonstrates a balanced and stable performance, outperforming CNN (92.3%) and RF (89.9%).
- **AUC-ROC:** The Area Under the Receiver Operating Characteristic curve (AUC-ROC) assures whether the model can classify benign activity and malicious activity at any classification threshold. A bigger AUC-ROC signifies more powerful discrimination in general. It has been observed that the best score was presented by the hybrid model, 98.3 percent, proving the model to be robust and adaptive, while CNN, RF, and SVM scored 94.2 percent, 90.4 percent, and 89.1 percent, respectively.

4.3. Real-Time Evaluation

To evaluate how well the proposed hybrid intrusion detection system is ready for reality, real-time experiments were conducted in a controlled environment using cloud-native log streams. It was built by integrating the system with two popular cloud auditing and monitoring services, namely AWS CloudTrail and Azure Monitor, which record detailed operational and security-related telemetry data. These platforms delivered constant flows of logs across the different cloud resources, compute instances, storage buckets, API calls, and user activities that were replicated in real-life scenarios of Infrastructure-as-a-Service (IaaS) and Platform-as-a-Service (PaaS) models. To process these logs and consume them in real-time, a hybrid model (comprising a CNN, LSTM block, and Autoencoder) was deployed. The pipeline encompassed live ingestion, preprocessing, feature extraction, and anomaly scoring. On average, the system received a latency of about 1.2 seconds for detection, that is, when an anomalous event on the system has been logged and an alert is generated. Such low-latency performance shows that the model can enable near-real-time feedback on threats, and this ability is critical in reducing time-sensitive security attacks like privilege escalation, lateral movement, or data exfiltration.

Moreover, the scalability and flexibility of the system were ensured by running multiple loads and varying log densities across cloud tenants and services. The hybrid model performed steadily, and although its performance degraded slightly, the degradation was minimally noticeable, highlighting its robustness and compliance with dynamic, high-volume cloud scenarios. It was also prepared in a way that allows for easy extension to other platforms, such as Google Cloud or a hybrid cloud configuration, as the model was built in a modular manner with respect to its cloud service integration.All in all, the model's performance confirmed its practical usability. It demonstrated that it is not only highly effective at detecting malicious requests offline, but also that it sufficiently meets the rigorous performance and latency requirements of online cloud security monitoring in live conditions. This makes it an accessible option for contemporary businesses seeking

5. Conclusion

This study proposes a new deep learning-based approach to detecting abnormal data breaches in distributed cloud storage infrastructure systems. Considering that the conventional security tools, including even the traditional machine learning solutions, fail to address the challenge of advanced and continuously evolving cyber threats, this paper has proposed a hybrid algorithm capable of leveraging the spatially-sensitive abilities of Convolutional Neural Networks (CNN), the sequential-based learning capabilities of Long Short-Term Memory (LSTM) networks, and the reconstruction-based anomalous detection capabilities of Autoencoders. The joint architecture was crafted to capture elusive and intricate breach trends within proximity to real-time, with a high degree of accuracy in detecting the breach, whilst being efficient in terms of computational processing demands. By testing the model during massive experimentation with the UNSW-NB15 dataset and generated cloud operation logs, the network surpassed the baseline models (which included Support Vector Machines (SVM), Random Forests (RF), and CNNs on their own- especially when it came to F1-Score and AUC-ROC). Furthermore, testing in a deployment environment based on a real-time cloud proved that the system is capable of functioning in a real-time environment, having a low detection delay in the range of 1.2 seconds, which allows it to be practically applicable in active security monitoring.

The results of the present research have a number of critical implications for academic studies and industrial applications. The second way is direct, according to which the increased detection rates result in a better security posture of the cloud, where reaction to incidents is quicker and data breach damages are minimized. The suggested framework is also very scalable and adjustable, and it can be applied to numerous types of cloud platforms and architectures, such as hybrid and multi-cloud environments. The generalization that it can perform across different log formats and operating environments makes it a versatile instrument in the current enterprise security frameworks.Although the existing structure has proven to be quite successful, there are several areas for improvement in the future. A key direction is the introduction of federated learning mechanisms, which enable decentralised, privacy-preserving training across geographically distributed cloud nodes. It would allow updates across models without revealing confidential data. Second, there is room to increase the number of Internet of Things (IoT) cloud nodes, which are increasingly popular and subject to serious attacks because they include few security functions. Lastly, a road map of research will focus on the ability to develop an auto-adaptive mechanism, which will be able to detect Zero-day attacks, previously unseen threats, through constant updating of the model according to feedback and online learning, and improve resilience against new strategies of intrusion.

References

[1] Debar, H., Dacier, M., & Wespi, A. (2000). A revised taxonomy for intrusion-detection systems. Annals of Telecommunications, 55(7-8), 361-378.

[2] Sun, P., Liu, P., Li, Q., Liu, C., Lu, X., Hao, R., & Chen, J. (2020). DL-IDS: Extracting Features Using CNN-LSTM Hybrid Network for Intrusion Detection System. Security and communication networks, 2020(1), 8890306.

[3] Maklachkova, V. V., Dokuchaev, V. A., & Statev, V. Y. (2020, October). Risk identification in the exploitation of a geographically distributed cloud infrastructure for storing personal data. In 2020 International Conference on Engineering Management of Communication and Technology (EMCTECH) (pp. 1-6). IEEE.

[4] Barona, R., & Anita, E. M. (2017, April). A survey on data breach challenges in cloud computing security: Issues and threats. In 2017, International Conference on Circuit, Power and Computing Technologies (ICCPCT) (pp. 1-8). IEEE.

[5] Kovalenko, E. (2020). Advancements in Cloud-Based Infrastructure for Scalable Data Storage: Challenges and Future Directions in Distributed Systems. International Journal of AI, Big Data, Computational and Management Studies, 1(1), 12-20.

[6] Padmavathi, D. G., & Shanmugapriya, M. (2009). A survey of attacks, security mechanisms and challenges in wireless sensor networks. arXiv preprint arXiv:0909.0576.

[7] Caballero-Anthony, M. (2016). Non-traditional security concept, issues, and implications on security governance. Georgetown Journal of Asian Affairs.

[8] Nassif, A. B., Talib, M. A., Nasir, Q., Albadani, H., & Dakalbab, F. M. (2021). Machine learning for cloud security: a systematic review. IEEE Access, 9, 20717-20735.

[9] Bhamare, D., Salman, T., Samaka, M., Erbad, A., & Jain, R. (2016, December). Feasibility of supervised machine learning for cloud security. In 2016 International Conference on Information Science and Security (ICISS) (pp. 1-5). IEEE.

[10] Subramanian, E. K., & Tamilselvan, L. (2019). A focus on future cloud: machine learning-based cloud security. Service Oriented Computing and Applications, 13(3), 237-249.

[11] Butt, U. A., Mehmood, M., Shah, S. B. H., Amin, R., Shaukat, M. W., Raza, S. M., ... & Piran, M. J. (2020). A review of machine learning algorithms for cloud computing security. Electronics, 9(9), 1379.

[12] Qayyum, A., Ijaz, A., Usama, M., Iqbal, W., Qadir, J., Elkhatib, Y., & Al-Fuqaha, A. (2020). Securing machine learning in the cloud: A systematic review of cloud machine learning security. Frontiers in Big Data, 3, 587139.

[13] Ge, M., Fu, X., Syed, N., Baig, Z., Teo, G., & Robles-Kelly, A. (2019, December). Deep learning-based intrusion detection for IoT networks. In 2019 IEEE 24th Pacific Rim International Symposium on dependable computing (PRDC) (pp. 256-25609). IEEE.

[14] Marwan, M., Kartit, A., & Ouahmane, H. (2018). Security enhancement in healthcare cloud using machine learning. Procedia Computer Science, 127, 388-397.

[15] Loukas, G., Vuong, T., Heartfield, R., Sakellari, G., Yoon, Y., & Gan, D. (2017). Cloud-based cyber-physical intrusion detection for vehicles using deep learning. IEEE Access, 6, 3491-3508.

[16] Garg, S., Kaur, K., Kumar, N., Kaddoum, G., Zomaya, A. Y., & Ranjan, R. (2019). A hybrid deep learning-based model for anomaly detection in cloud datacenter networks. IEEE Transactions on Network and Service Management, 16(3), 924-935.

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

[18] Zekrifa, D. M. S. (2014). Hybrid Intrusion Detection System. Theses, School of Information Technology & Mathematical Sciences.

[19] Sun, Y., Wang, X., & Tang, X. (2013). Hybrid deep learning for face verification. In Proceedings of the IEEE International Conference on Computer Vision (pp. 1489-1496).

[20] Thamilarasu, G., & Chawla, S. (2019). Towards deep-learning-driven intrusion detection for the Internet of Things. Sensors, 19(9), 1977.