International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 05 May 2026
Page No: 3453-3460
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0447

Predicting Host Compromise Risk on Linux Hosts Using Machine Learning over CIS Compliance Failures

*Rahul Bhattarai[1], Sandeep M[2], G Jai Ganesh[3], *Maranco M[4].*
[1,2,3] UG - CSE-Cyber security, Department of Networking and Communications, SRM Institute of Science and Technology, Chennai, Tamilnadu, India.
[4]Assistant Professor, Department of Networking and Communications, SRM Institute of Science and Technology, Chennai, Tamilnadu, India.
***Email ID:** rb6738@srmist.edu.in[1], sm1995@srmist.edu.in[2], jg0824@srmist.edu.in[3], marancom@srmist.edu.in[4].*

Abstract
There are a lot of users of configuration compliance tools such as security standards like the Center for Internet Security (CIS) and security frameworks like NIST SP 800-53 to make Linux systems secure. However, these tools are mostly used to generate a report on pass or fail based on the configurations of the Linux systems. Therefore, it is the responsibility of the security administrator to make sense of the results of the configuration compliance failure. The main objective of this study is to introduce the readers to the machine learning-based system to convert the problems identified in the CIS benchmark compliance to warning signs for possible cyber attacks. The proposed framework does not rely on the CVE database or exploit traces, as in traditional vulnerability-based methods. Rather, the proposed framework only depends on the misconfigurations and hardening issues detected by the CIS Benchmark scans. The feature engineering process, being a systematic approach, processes the compliance data from various Linux virtual machines, resulting in the classification of low-level control failures into security domains. A deterministic rule-based approach is applied to map the misconfigurations to potential adversarial objectives, resulting in a supervised multi-class classification problem. A random forest approach is applied to classify the types of attacks that are likely to occur, including brute force, privilege escalation, persistence, remote, multi-vector attacks, etc. This can be achieved through experimental assessment, whereby multi-class performance metrics are used to evaluate the results obtained. The results indicate that there is enough predictive information in the configuration compliance failures to make precise predictions about the nature of attacks that can be expected. Feature importance gives us even more useful information about the CIS domains, which are of most importance in the nature of attacks predicted. The research has bridged the gap between configuration auditing and proactive defence in a significant manner

***Keywords:** Configuration compliance; Linux security; Machine learning; Predictive security analytics; Risk assessment.*

1. Introduction
Linux-based operating systems are the heart of the current business IT infrastructure. These systems are the platform for servers, cloud systems, network appliances, and critical systems. These systems are often the target of cybercriminals because they are so popular. While people often focus on the vulnerabilities of the code, many of the real-world security issues are often the result of things like systems being configured incorrectly, authentication being weak, privilege management being weak, and

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 05 May 2026
Page No: 3453-3460
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0447

systems being exposed to the public. These are often the result of not properly hardening the system, but through flaws in the application code. Organisations use standardised hardening guides, like the Center for Internet Security (CIS) Benchmarks, and security control guides, like NIST Special Publication 800-53 [7], [8], to minimize the risks that come with configuration. These tools, which use these standards, like CIS-CAT and OpenSCAP, generate comprehensive compliance reports that indicate whether a security control has passed or failed. These reports, although useful for auditing and baselining, remain static and only indicate what is or is not there. They do not measure risk at a host level or speculate what kind of attacks might be possible with a given configuration [1]. As a result, security administrators must read these long reports manually without assistance from predictive tools. Previous studies conducted on cybersecurity analytics have shown the efficiency of the application of machine learning methodologies for the prediction of the existence of vulnerabilities, the probability of exploitation, and patterns of intrusions based on the information available from the vulnerability database, system log information, and source code metrics. In a study conducted on vulnerability prediction for the Linux kernel, vulnerability severity estimation, and overall risk modeling based on the application of machine learning methodologies, the efficiency of predictive models has been shown for the application in the security domain. However, the predictive model has been limited mostly to vulnerabilities and has been based on the information available from CVE/CVSS and attack patterns. There has been no information available on the application of compliance information for predictive modeling. Recently, research has been done on the application of artificial intelligence for compliance monitoring and automated auditing [5]. It has shown that it might be possible to automate the interpretation of compliance on a large scale, although this is currently mainly based on rules and reactive. Explainability has become a key factor for the application of machine learning for cybersecurity scenarios [3]. It is a fact that machine learning models that can be explained easily are necessary to ensure that the results of the prediction provide useful information rather than vague risk scores. Even though all these advancements have been achieved, little research has been done to investigate the potential usage of CIS benchmark non-compliance as structured indicators to predict the risk on the host. This research aims to answer the following question: Is it possible for machine learning algorithms to make predictions about the nature of possible cyber attacks on Linux systems, considering only the non-compliance issues detected by CIS benchmark scans? To address this research question [2], a machine learning-based framework has been conceptualized to convert CIS benchmark scan results into machine learning security features, and the concept of risk assessment for the Linux host has been conceptualized as a supervised multi-class classification problem. The configuration issues are mapped to potential adversarial attack scenarios, including brute force, privilege escalation, persistence, remote attacks, and complex scenarios involving multiple vectors of attack. The random forest classifier has been proposed to model the non-linear relationship between the misconfiguration domain and potential adversarial attack scenarios. In order to respond to this inquiry, the current research introduces a framework based on machine learning, which converts the results of CIS benchmark compliance into security features, as well as the idea of the host risk assessment as a supervised multi-class classification issue [4]. The problems associated with the configuration are aligned with the potential adversarial goals, including brute force, privilege escalation, persistence, remote exploitation, as well as complex scenarios involving multiple vectors of attack. The random forest classifier is proposed in order to deal with the non-linear relationships that exist in the misconfiguration domain and the potential adversarial attack scenarios. By bridging the gap that has existed between compliance auditing and predictive threat modelling, the current research demonstrates the manner in which the data associated with the configuration can be utilised and leveraged in order to make proactive and intelligent security

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 05 May 2026
Page No: 3453-3460
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0447

decisions within the context of a Linux-based system [6].

2. Method

The proposed framework uses a modular pipeline to transform data on Linux configuration compliance into predictive security intelligence. The five stages of the proposed architecture are as follows:

- Injection of Configuration
- Doing a compliance audit
- Extraction and Aggregation of Features
- Assigning Attack Labels
- Classification with Supervised Machine Learning

This proposed model is intended to ensure the reproduction of all the experimental samples and features as well as their isolation and reproduction. All the machine learning algorithms make use of the same feature engineering pipeline as well as depending on the outputs of the CIS benchmark compliance. The entire workflow begins with the raw data obtained from the compliance audit and is transformed into a structured form of numbers to carry out multi-class classification in fig 1 [9].

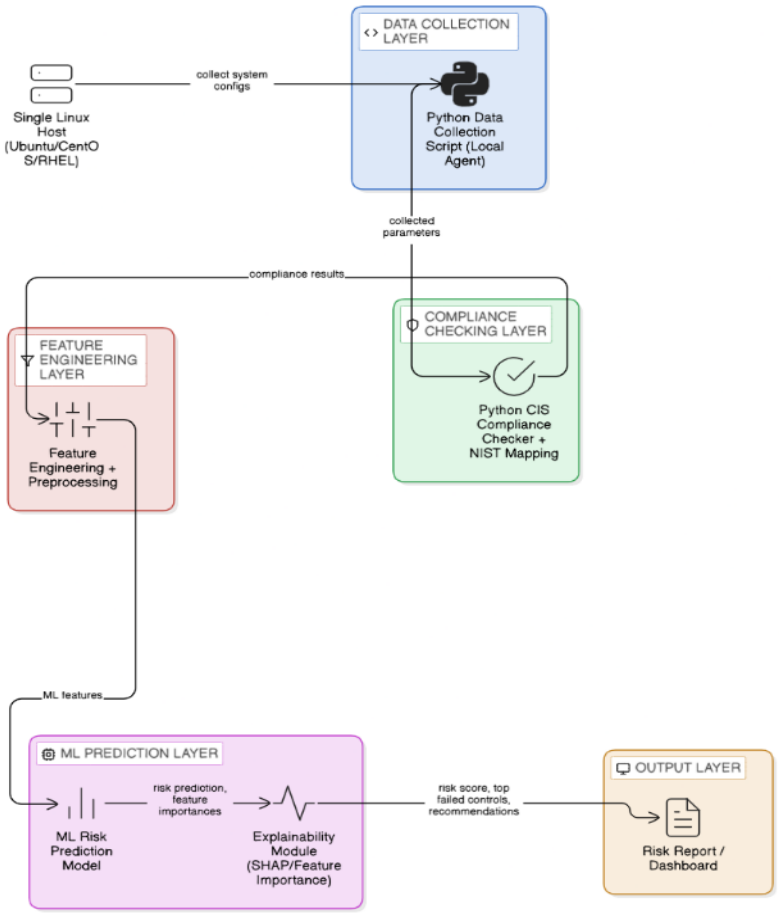

Figure 1 Flowchart of System Architecture

2.1.Compliance Data Collection and Parsing

The experiments were carried out in a VMware-based Virtualised Linux environment. Snapshot Isolation was utilized to ensure that all configuration experiments were carried out in a pristine baseline state. This ensured that the configuration results of the experiment did not affect the results of the next audit. **The steps involved in the experiment were as follows**:

- Hardened baseline snapshot restore

IRJAEH
International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 05 May 2026
Page No: 3453-3460
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0447

- Usage of the selected misconfiguration modules
- Compliance audit, as per CIS guidelines
- Exporting the results in the JSON format
- Storing the output in a persistent directory on the host

2.2. Automated Parsing of CIS Reports

Audit tools used by CIS generated JSON reports with the following attributes:

- Identifier for Control
- Control Description
- Status: Pass or Fail
- Control Severity Level (Level 1 / Level 2)
- Security Domain Category

Python parser was implemented for the purpose of automatic extraction of the required attributes from the JSON reports, which would be used for the purpose of transforming the raw audit reports into a structured tabular form.

2.3. Feature and building dataset building

Prior to model training, the data was made easier to understand and the number of features was reduced through the process of feature engineering. [10]Each configuration for the Linux hosts was presented by utilizing 29 features based on the status of the CIS control, Domain-level indicators that, when combined, provide categories for security-related information with 3,500 total host configurations and 29 engineered features.

There were security domains, which included:

- Controls for authenticating
- Setting up SSH
- Managing privileges
- Making the network stronger
- Keeping records and checking them
- Security for boot
- Services that aren't safe

2.4. Attack Labelling and Problem Definition

For building a supervised data set without depending on any external attack samples, a set of rules was clearly defined [12]. **Each of the configurations was mapped to any of the seven attack categories as follows:**

- Brute Force
- Credential Attack
- Privilege Escalation
- Staying with it
- Avoiding Defence
- Exploitation from a distance
- Multi_Vector_Attack

Each of the configurations was given labels based on the following conditions:

- A lot of failures in important areas
- Authentication and network exposure weaknesses occurring at the same time
- Errors in the settings of the privilege or logging configurations
- Failures in multiple domains occurring at the same time

This was done to ensure that the results were reproducible, clear, and had no bias. There were 500 samples for each of the 7 perfectly balanced classes.

2.5. Machine Learning Model Selection

We selected a random forest classifier as our main learning model based on the following reasons:

- Robustness to correlated features
- Can handle non-linear relationships
- Reduces the chance of overfitting
- Has an inherent feature importance estimation

The classifier contains 100 decision trees, and all of them have been trained using the method of bootstrap aggregation.The majority vote method is used for making predictions.Using the method of mean decrease in impurity, we were able to achieve the feature importance scores and determine which security domains are the most important for the different types of attacks.

2.6. Attack Prediction and Explainability

After the Random Forest model has been trained, it has the capability of determining the kind of attack that is likely to occur on Linux systems that have not been encountered before, based on the best practices provided by CIS. The way in which the model is able to explain the prediction is through the feature importance mechanism, in which it is clear what CIS domains are being affected the most in terms of the kind of attack.This is what ensures that the prediction is not being carried out in a way in which it seems to be a black box, since it is clear what kind of configuration issues are being encountered in order to

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 05 May 2026
Page No: 3453-3460
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0447

address them accordingly.

2.7. What Security Frameworks Do

CIS Benchmarks is the main source of information that is used to verify the compliance. For easier understanding, the controls are grouped into security domains based on structured control frameworks such as those presented by the National Institute of Standards and Technology (NIST) [8] [13]. The security domains include access control, auditing & logging, config management, network safety, and privilege management. The feature engineering process is more semantically structured if the low-level controls presented by the CIS controls are grouped into meaningful security domains shown in table 1.

Table 1 Classification Performance per Attack Category

Attack Category	Precision	Recall
Brute_Force	0.88	0.87
Defense_Evasion	0.86	0.83
Credential_Attack	0.80	0.75
Remote Exploitation	0.77	0.82
Persistence	0.74	0.81
Multi_Vector_Attack	0.83	0.68
Privilege Escalation	0.66	0.73
Macro Average	0.79	0.78
Attack Category	Precision	Recall
Brute_Force	0.88	0.87
Defense_Evasion	0.86	0.83
Credential_Attack	0.80	0.75

3. Results And Discussion

3.1. Results

As shown by the proposed Random Forest model, balanced and consistent multi-class classification performance was achieved with a macro precision of 0.79, a macro recall of 0.78, and a macro F1-score of 0.78 for the seven different types of attacks.Furthermore, the detection of Brute_Force resulted in the highest performance with an F1-score of 0.87, high precision of 0.88, and high recall of 0.87, which shows that authentication-related misconfigurations were easily distinguishable from each other. On the other hand, Defense_Evasion was also robust in classifying the different types of attacks with an F1-score of 0.84, which shows strong discrimination of failures in logging and audit controls.In addition, the detection of Credential_Attack resulted in an F1-score of 0.77, and the detection of Remote_Exploitation resulted in an F1-score of 0.79, which shows consistent predictive power.However, the detection of Remote_Exploitation resulted in a higher recall of 0.82, which shows that this attack was more capable of identifying weaknesses in network and service-related attacks shown in fig 2.

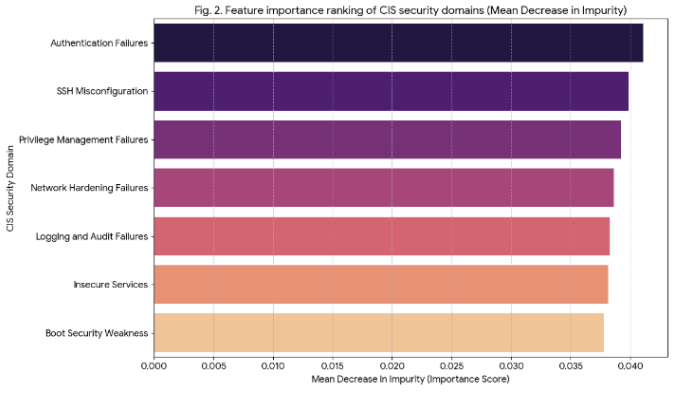

Figure 2 Feature importance ranking of CIS security domains based on mean decrease in impurity

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 05 May 2026
Page No: 3453-3460
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0447

Persistence resulted in balanced performance with an F1-score of 0.77, which shows distinct patterns of privilege and configuration controls. The Multi_Vector_Attack classification task had a good precision score of 0.83, while the recall score was relatively lower at 0.68. This indicates that the model is good at recognizing compound attack scenarios while there is some overlapping domain interaction that makes it less sensitive to detection. Privilege_Escalation was the most difficult classification task for the model with an F1 score of 0.69, possibly because there was some overlapping feature pattern with scenarios that assist individuals in staying in the game without being caught in fig 3.

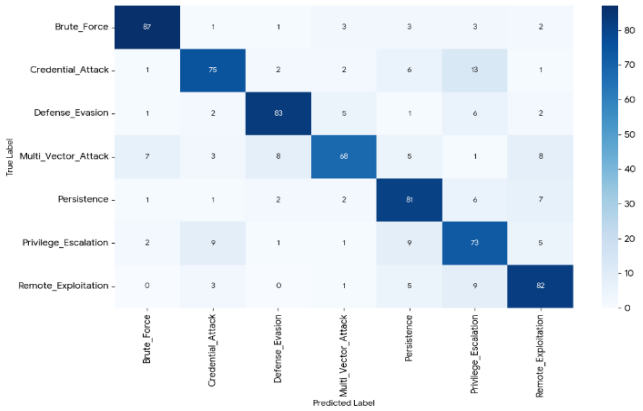

Figure 3 Confusion matrix of Random Forest multi-class attack prediction

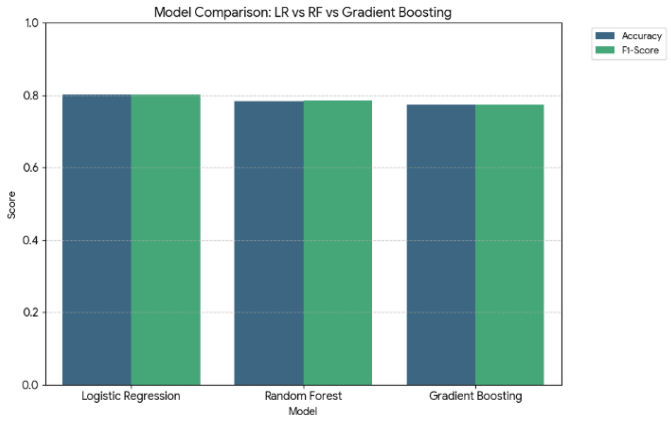

Figure 4 Model Comparison

As can be noted in the suggested model's confusion matrix in Figure 3, the model is performing well, and the classification is clear with minimal errors. Figure 4 illustrates the feature importance ranking, which shows that the authentication, SSH configuration, and privilege management domains are the most important features in classifying the attack types. Table I shows the performance metrics of the suggested model per class. The results have indicated that the CIS compliance failures have enough structural information to allow the classification of the attack types to be performed effectively [15]. This is further supported by the fact that the macro-balanced results include all the classes, ranging from class 0 to class 7, which is an indication that the predictive model can be effective in the configuration-based classification of the Linux environment to proactively assess the risk at the host level [14].

3.2. Discussion

The experimental results verify the fact that the non-compliance of CIS benchmarks can be effectively used as a predictive indicator for the classification of cyber attacks in a structural manner. The balanced macro F1 score of 0.78 has been achieved for seven different types of cyber attacks by the Random Forest Classifier. The detection scores achieved for the Brute_Force and Defense_Evasion types of cyber attacks verify the fact that weaknesses related to authentication and weaknesses related to logging are more likely to have structural patterns in the feature space, resulting in a high degree of separability to achieve precision and recall. The moderate performance achieved for the classification of Credential_Attack, Persistence, and Remote_Exploitation types of cyber attacks verify the fact that there exist certain overlapping features related to security domains. For example, poor policies related to SSH may lead to brute force attacks as well as remote exploitation attacks. The F1 score of 0.69 achieved for the Privilege_Escalation feature verifies the fact that there exist certain overlapping features related to privilege, persistence, and logging. The results suggest that certain types of attacks are structurally linked in the configuration space, which makes it difficult to draw sharp lines in the feature space. The results obtained by the lower

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 05 May 2026
Page No: 3453-3460
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0447

performance of the Multi_Vector_Attack prove the difficulty of dealing with simultaneous attacks. It has a very high accuracy rate of 0.83, which means it is capable of detecting things if they are predicted. Some of the configurations are also likely to have the appearance of being single-domain attack patterns, which makes them less sensitive. The results have shown that the predictive modelling based on the compliance of the configuration alone is capable of performing well in the categorization of the attacks, even without the use of the databases of the vulnerabilities, the exploit signatures, or the runtime telemetry data. This shows that the main idea of the use of the compliance data as a source of proactive threat intelligence is correct.

Practically speaking, this framework allows for the following:

- Remediation efforts to be placed at the top of your list based on the likelihood of the attack occurring.
- Dominant risk domains to be identified via feature importance analysis.
- A transition from static audit reporting to intelligence-driven hardening.

The results indicate that configuration-based predictive modeling should be treated as an additional layer to traditional vulnerability-based security analytics.

Conclusion

This research proposes a machine learning approach for predicting potential categories of cyber attacks on Linux systems by utilizing CIS Benchmark failures as input feature structures. The proposed approach employs static configuration audits to derive predictive security intelligence, which is unique from traditional vulnerability-based approaches. The performance of the Random Forest classifier was measured by an F1-score of 0.78 based on a balanced dataset of 3,500 Linux host configurations across seven categories of attacks. This proves that the structural information within the configuration data is adequate to derive reliable models for multi-class attacks. The high performance on authentication and defence categories proves that domain-level defence-related CIS controls improve class separability. Some similarities exist between privilege escalation and compound attacks, but overall, the results prove that predictive risk models based on configurations are feasible.

The framework enables:

- Prioritization of proactive remediation
- Attribution of risks for specific domains
- Transitioning from static reporting for compliance to hardening via intelligence

This research ties together the concepts of compliance auditing and predictive threat analysis, demonstrating that configuration compliance data can be utilized as a useful and understandable way to estimate cyber risks at a host level. Future research into the subject will improve upon this current research by adding additional data streams, evaluating other architectures, and testing the methodology in a real-world environment.

Acknowledgements

Place Acknowledgments, including information on the source of any financial support received for the work being published. Place Acknowledgments, including information on the source of any financial support received for the work being published.

References

[1]. M. Jimenez, “Vulnerability Prediction Models: A Case Study on the Linux Kernel,”University of Luxembourg, 2016.

[2]. R. Watanabe, “Machine Learning Based Prediction of Vulnerability Severity,”roceedings of the 20th International Conference on Security and Cryptography (ICISSP),SCITEPRESS, 2023, pp. 1–10.

[3]. N. Capuano, G. Polese, A. Rizzo, and M. R. Guarino, “Explainable ArtificialIntelligence in Cybersecurity: A Survey,” IEEE Access, vol. 13, pp. 1–25, 2025.

[4]. A. Barlybayev, K. R. Khaydarov, and M. A. Khasanova, “Development of a FlexibleInformation Security Risk Model Using Machine Learning Methods and Ontologies,”Applied Sciences, vol. 14, no. 21, 2024.

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 05 May 2026
Page No: 3453-3460
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0447

[5]. F. Y. Loumachi, M. B. Yagoubi, and A. A. Draa, "AI in Control: RethinkingCybersecurity Compliance and Auditing", SSRN Electronic Journal, 2025.
[6]. P. Shahrivar, "Detection of Vulnerability Scanning Attacks Using Machine Learning,"Master's Thesis, Dalarna University, 2022.
[7]. Center for Internet Security, "CIS Benchmarks," CIS, Latest Version, 2024.
[8]. National Institute of Standards and Technology, "Security and Privacy Controls for Information Systems and Organisations", NIST Special Publication 800-53 Revision 5,
[9]. NIST, 2020.T. Hoang, "Creating a Security Baseline Using NIST Controls," European Journal of Electrical Engineering and Computer Science, vol. 8, no. 2, 2024..
[10]. S. M. Bridges, R. B. Vaughn, "Fuzzy Data Mining and Genetic Algorithms Applied to Intrusion Detection," Proceedings of the National Information Systems Security Conference, 2000.
[11]. K. Scarfone, P. Mell, "Guide to Intrusion Detection and Prevention Systems (IDPS)," NIST Special Publication 800-94, National Institute of Standards and Technology, 2007.
[12]. J. Zhang, M. Zulkernine, A. Haque, "Random-Forests-Based Network Intrusion Detection Systems," IEEE Transactions on Systems, Man, and Cybernetics, vol. 38, no. 5, pp. 649–659, 2008.
[13]. A. L. Buczak, E. Guven, "A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection," IEEE Communications Surveys & Tutorials, vol. 18, no. 2, pp. 1153–1176, 2016.
[14]. M. Ring, D. Schlör, D. Landes, A. Hotho, "Flow-Based Network Traffic Generation Using Generative Adversarial Networks," Computers & Security, vol. 82, pp. 156–172, 2019.
[15]. H. Hindy, D. Brosset, E. Bayne, A. Seeam, C. Tachtatzis, R. Atkinson, X. Bellekens, "A Taxonomy of Network Threats and the Effect of Current Datasets on Intrusion Detection Systems," IEEE Access, vol. 8, pp. 104650–104675, 2020.