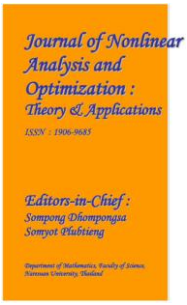
Journal of **N**onlinear **A**nalysis and **O**ptimization
Vol. 15, Issue. 1 : 2024
ISSN : **1906-9685**



CYBER SECURITY STATUS ASSESSMENT OF CLOUD MANUFACTURING SYSTEMS BASED ON SEMI QUANTITATIVE INFORMATION

N. NARASIMHA RAO, Department of Information Technology, NRI Institute of Technology, Pothavarappadu (V), Agiripalli (M), Eluru (Dt)-521212
B. JAHNAVI, Department of Information Technology, NRI Institute of Technology, Pothavarappadu (V), Agiripalli (M), Eluru (Dt)-521212
G. TARUN KALYAN Department of Information Technology, NRI Institute of Technology, Pothavarappadu (V), Agiripalli (M), Eluru (Dt)-521212
M. ANIRUDH Department of Information Technology, NRI Institute of Technology, Pothavarappadu (V), Agiripalli (M), Eluru (Dt)-521212

Abstract:
The incorporation of cloud computing technology has transformed production processes in the context of contemporary manufacturing, providing scalability, flexibility, and cost-effectiveness. On the other hand, the use of cloud manufacturing systems presents fresh cybersecurity issues that call for thorough analysis and preventative measures. Based on semi-quantitative data, this study offers a novel method for evaluating cloud manufacturing systems' cybersecurity state. The approach considers elements including network architecture, data integrity, access restrictions, and threat detection methods when assessing the security posture of cloud-based manufacturing settings. It also integrates qualitative and quantitative considerations. Through the synthesis of expert knowledge and empirical data, the proposed framework enables stakeholders to systematically evaluate the security risks and vulnerabilities associated with cloud manufacturing systems. Furthermore, the semi-quantitative nature of the assessment allows for a more nuanced understanding of cybersecurity risks, facilitating informed decision-making and prioritization of security measures. The effectiveness of the proposed methodology is demonstrated through a case study analysis, highlighting its applicability and utility in enhancing the resilience and security of cloud manufacturing infrastructures amidst evolving cyber threats.

1. Introduction

An approach to cyber security risk assessment for cloud manufacturing systems is presented in this study. To effectively evaluate cyber security risks, the method blends fuzzy comprehensive evaluation with an upgraded attack tree model. The efficacy and practicability of the suggested method are illustrated through case studies, offering insightful information for improving cyber security in cloud manufacturing systems. A semi-quantitative risk assessment methodology for assessing cloud manufacturing systems' cybersecurity is presented in this study. To properly estimate cyber threats, the strategy combines semi-quantitative analysis approaches with qualitative risk variables. The suggested technique illustrates its usefulness and efficacy in boosting cybersecurity in cloud manufacturing environments through case studies and comparative analysis.A fuzzy comprehensive evaluation approach for cybersecurity assessment in cloud manufacturing systems is presented in this study. A more adaptable and flexible framework for assessing cyber risks in cloud-based manufacturing systems is provided by the technique, which makes use of fuzzy logic theory to address uncertainties and ambiguities in cybersecurity evaluations.Based on the Analytic Hierarchy Process (AHP), this study suggests a risk assessment technique for cybersecurity in cloud manufacturing systems. The strategy takes into account a variety of criteria and decision considerations, making it easier to identify and prioritise cybersecurity risks in a methodical manner. The efficacy and dependability of the

suggested method are proven by case studies and sensitivity analysis, providing insightful information for reducing cyber hazards in cloud-based manufacturing systems[1-30].

2. Proposed Method

The goal of the suggested technique is to overcome the shortcomings of current methodologies while utilising the benefits of combining qualitative insights and quantitative indicators for the cyber security status evaluation of cloud manufacturing systems based on semi-quantitative data. Our methodology, which combines the advantages of both qualitative and quantitative assessments, emphasises a comprehensive awareness of cyber security concerns.The integration of semi-quantitative approaches, which fill the gap between qualitative assessments and quantitative measurements, is the fundamental component of our suggested framework. Our method allows for a more contextualised and nuanced assessment of cyber security risks in cloud manufacturing systems by integrating semi-quantitative data. This involves giving quantifiable measurements for risk prioritisation and decision-making, while also capturing qualitative subtleties and expert opinions.

Advantages of proposed system

1. The suggested system integrates qualitative insights, quantitative measurements, and semi-quantitative approaches to provide a comprehensive approach to cyber security status evaluation.
2. The system is made to be flexible and sensitive to the ever-changing threat environment that affects cloud manufacturing systems.
3. To effectively handle changing threats and vulnerabilities, the system places a strong emphasis on ongoing monitoring and updating of cyber security assessments.

2.1 SYSTEM ARCHITECTURE

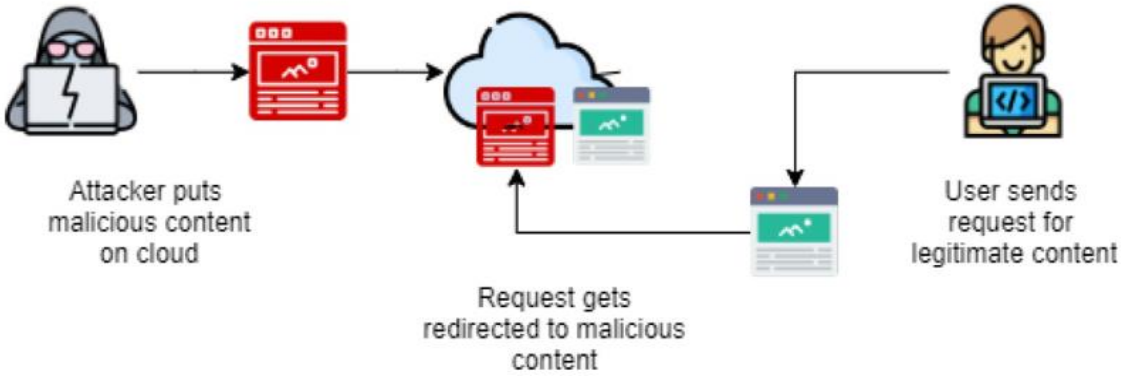


Figure.1. System architecture

2.2 DATA FLOW DIAGRAM

1. Another name for the DFD is a bubble chart. A system can be represented using this straightforward graphical formalism in terms of the input data it receives, the different operations it performs on that data, and the output data it generates.
2. One of the most crucial modelling tools is the data flow diagram (DFD). The components of the system are modelled using it. These elements consist of the system's procedure, the data it uses, an outside party that communicates with it, and the information flows within it.
3. DFD illustrates the flow of information through the system and the various changes that alter it. This method uses graphics to show how information flows and the changes made to data as it goes from input to output.
4. Another name for DFD is a bubble chart. Any level of abstraction can be utilised to portray a system using a DFD. DFD can be divided into phases that correspond to escalating functional detail and information flow.

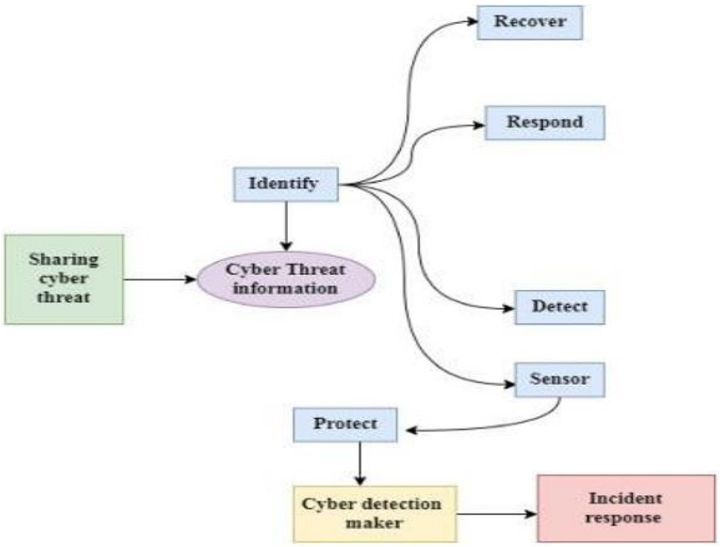


Figure.2. Dataflow diagrams

2.3 UML DIAGRAMS

Unified Modelling Language is known as UML. An industry-standard general-purpose modelling language used in object-oriented software engineering is called UML. The Object Management Group developed and oversees the standard.

The intention is for UML to spread as a standard language for modelling object-oriented software. The two main parts of UML as it exists now are a notation and a meta-model. In the future, UML may also include other processes or methods that are connected to it.

A common language for business modelling and other non-software systems, as well as for defining, visualising, building, and documenting software system artefacts, is called the Unified Modelling Language.

The UML is an assembly of top engineering techniques that have been successfully applied to the modelling of complicated and sizable systems.

Creating objects-oriented software and the software development process both heavily rely on the UML. The UML primarily expresses software project design through graphical notations.

2.4 Use case diagram

According to the Unified Modelling Language (UML), a use case diagram is a particular kind of behavioural diagram that is produced from and defined by a use case study. Its objective is to provide a graphical summary of the functionality that a system offers in terms of actors, use cases (representations of their goals), and any interdependencies among those use cases. A use case diagram's primary goal is to display which actors receive which system functionalities. It is possible to illustrate the roles of the system's actors.

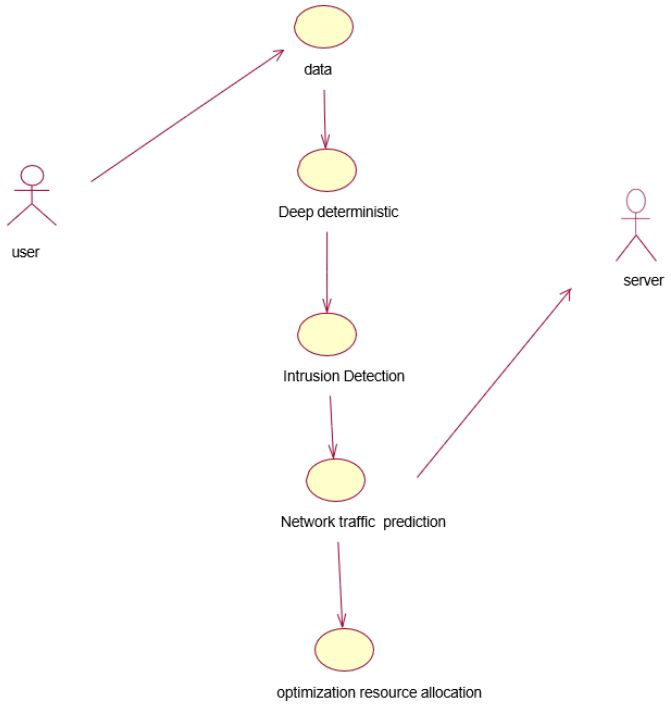


Figure.3. Use case diagram

2.5 Class diagram

The use case diagram and the system's comprehensive design are both improved by the class diagram. The actors identified in the use case diagram are categorised into a number of related classes by the class diagram. There are two types of relationships that can exist between the classes: "is-a" relationships and "has-a" relationships. It's possible that every class in the class diagram can perform certain functions. The "methods" of the class refer to these features that it offers. In addition, every class might possess specific "attributes" that allow for class uniqueness.

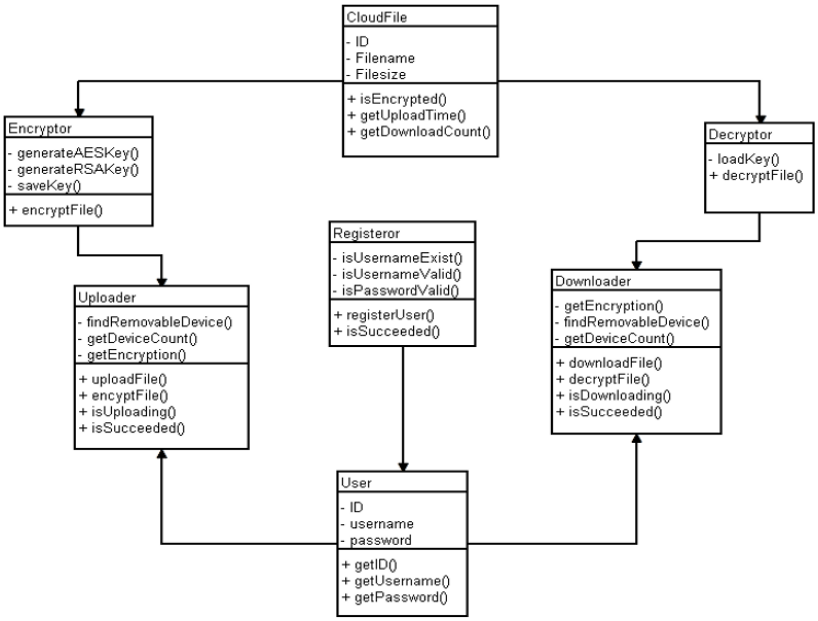


Figure.4. Class diagram

2.6 Activity diagram

The activity diagram shows how the system's processes are organised. An activity diagram has the same elements as a state diagram: activities, actions, guard conditions, initial and final states, and transitions.

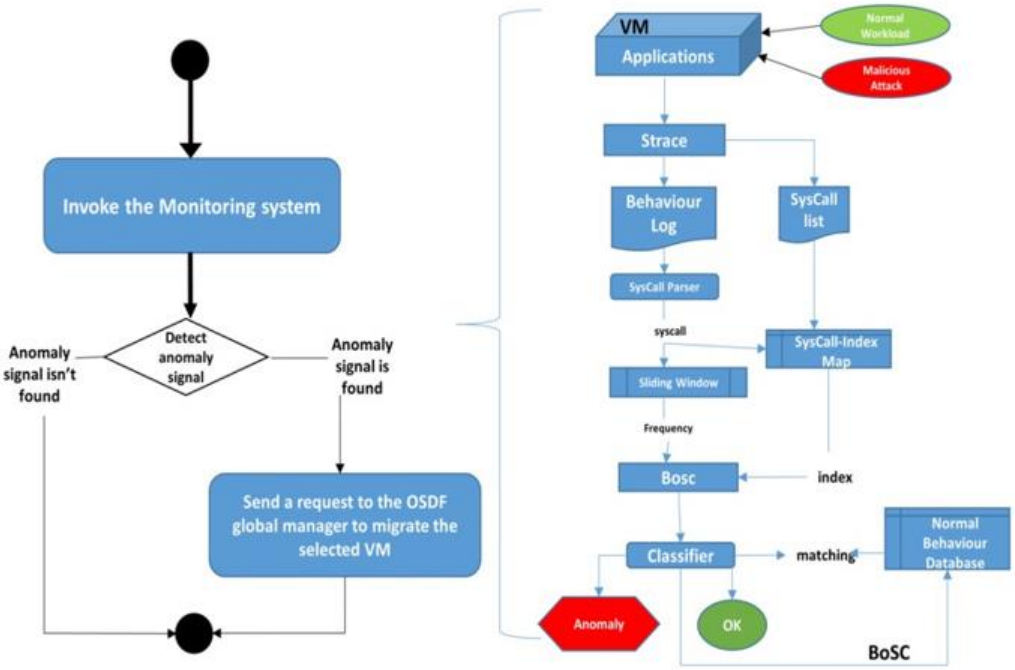


Figure.5. Activity diagram

2.7 Sequence diagram:

A sequence diagram represents the interaction between different objects in the system. The important aspect of a sequence diagram is that it is time-ordered. This means that the exact sequence of the interactions between the objects is represented step by step. Different objects in the sequence diagram interact with each other by passing "messages".

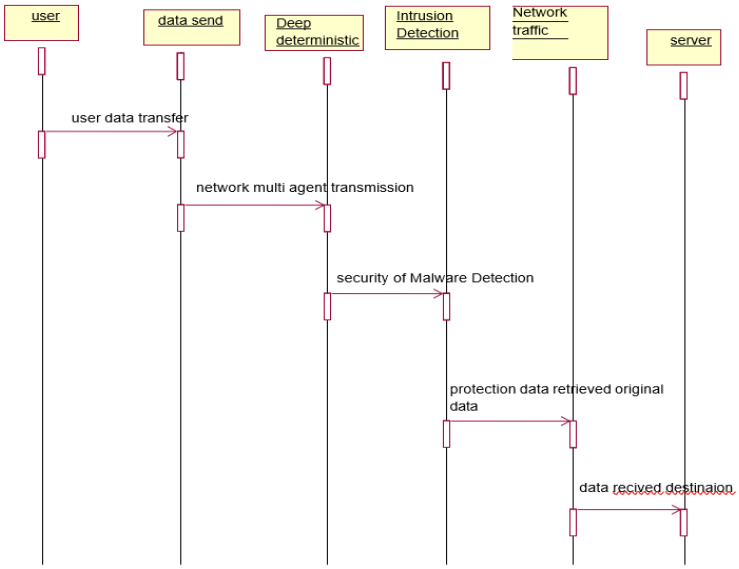


Figure.6. Sequence diagram

2.8 Collaboration diagram:

A cooperation diagram combines the ways in which various things interact with one another. To make it easier to follow the order of the encounters, they are listed as numbered interactions. All potential interactions between each object and other objects are identified with the aid of the cooperation diagram.

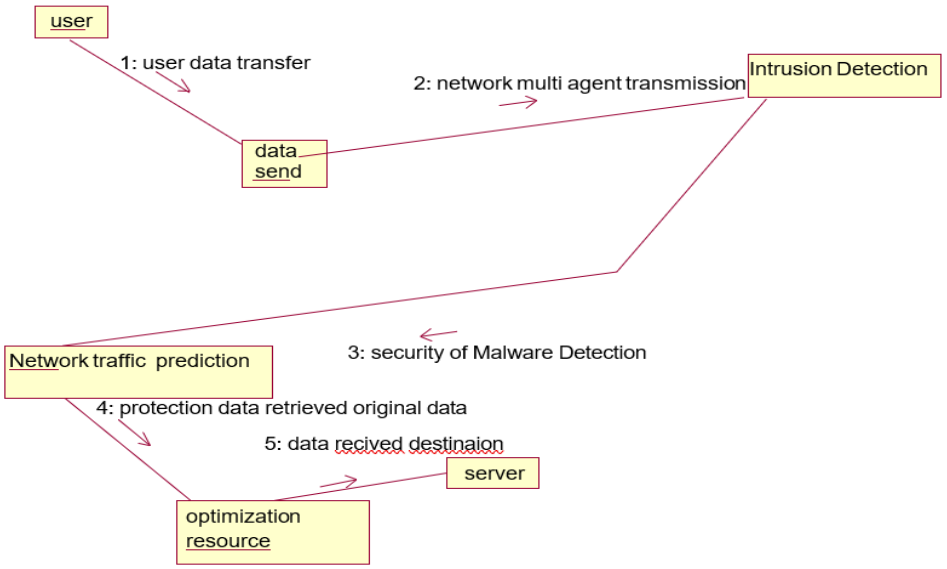


Figure.7. Collaboration diagram

2.9 Component diagram

The component diagram represents the high-level parts that make up the system. This diagram depicts, at a high level, what components form part of the system and how they are interrelated. A component diagram depicts the components culled after the system has undergone the development or construction phase.

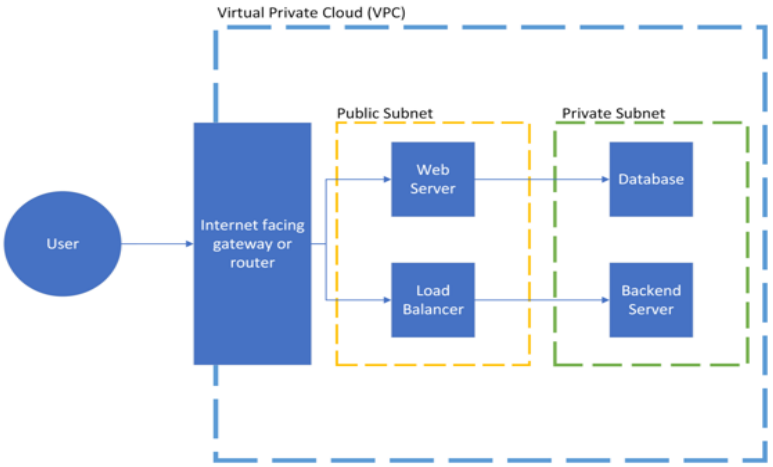


Figure.8. Component diagram

2.10 Deployment diagram

The deployment diagram captures the configuration of the runtime elements of the application. This diagram is by far most useful when a system is built and ready to be deployed.

Figure.9. Deployment diagram

2.11 SYSTEM TESTING

System testing is the process by which a quality assurance (QA) team assesses how the many components of an application interact with one another in the complete, integrated system or application. It is also known as system-level testing or system-integration testing. System testing confirms that a programme works as intended. This phase, which is a form of "black box" testing, is

concerned with an application's functioning. For instance, system testing might verify that all user input results in the desired output across the application.
System testing phases: A video guide for this particular test level. System testing looks at each and every part of an application to ensure that it functions as a cohesive whole. System testing is usually carried out by a quality assurance team following the examination of individual modules through functional or user-story testing, followed by integration testing for each component. Before a software build is put into production, where users use it, it undergoes acceptance testing to make sure it meets all requirements set forth in system testing. A team working on app development keeps track of all errors and decides what sorts and quantities are acceptable.

2.12 Software Testing Strategies:

The greatest strategy to make software engineering testing more effective is to optimise the approach. A software testing plan outlines the steps that must be taken in order to produce a high-quality final product, including what, when, and how. To accomplish this main goal, the following software testing techniques—as well as their combinations—are typically employed:
Static Examination: Static testing is an early-stage testing approach that is carried out without really operating the development product. In essence, desk-checking is necessary to find errors and problems in the code itself. This kind of pre-deployment inspection is crucial since it helps prevent issues brought on by coding errors and deficiencies in the software's structure.

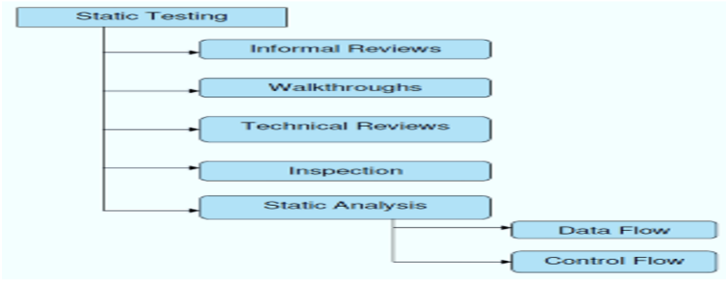


Figure.10. Static Testing

2.12 Structural Testing

Software cannot be tested efficiently unless it is run. White-box testing, another name for structural testing, is necessary to find and correct flaws and faults that surface during the pre-production phase of the software development process. Regression testing is being used for unit testing depending on the programme structure. To expedite the development process at this point, it is typically an automated procedure operating inside the test automation framework. With complete access to the software's architecture and data flows (data flows testing), developers and quality assurance engineers are able to monitor any alterations (mutation testing) in the behaviour of the system by contrasting the test results with those of earlier iterations (control flow testing).

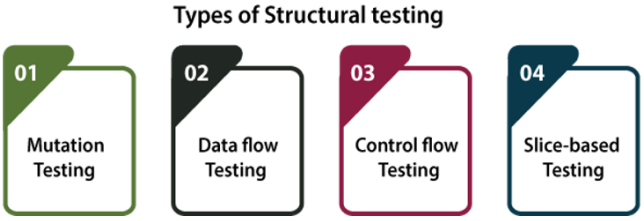


Figure.11. Structural Testing

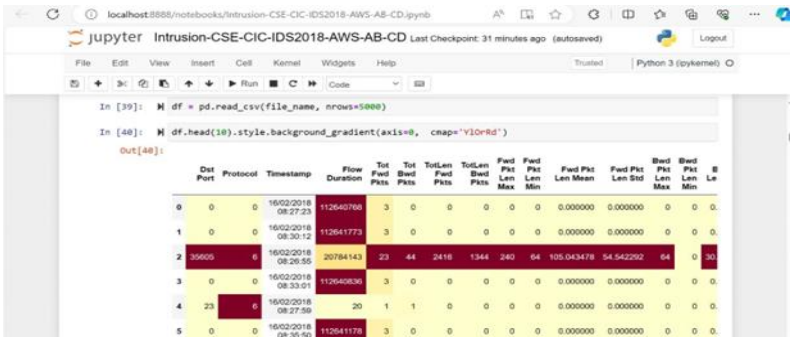
2.13 Behavioural Testing

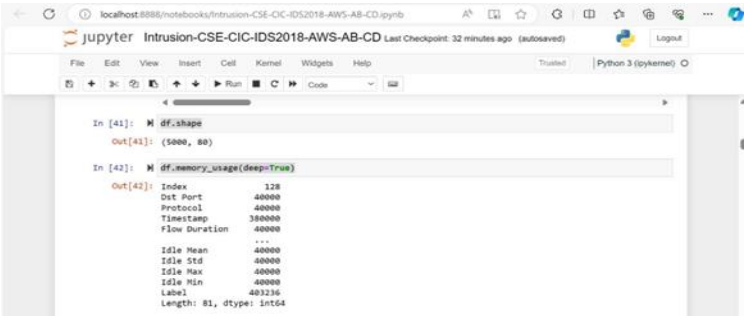
Rather than the mechanics underlying these reactions, the final testing phase concentrates on how the programme responds to different activities. Put differently, behavioural testing, commonly referred to as black-box testing, relies on conducting multiple tests, the majority of which are manual, in order to examine the product from the perspective of the user. In order to perform usability tests and respond to faults in a manner similar to that of ordinary users of the product, quality assurance engineers typically possess specialised information about a company or other purposes of the software, sometimes known as "the black box." If repetitive tasks are necessary, behavioural testing may also involve automation (regression tests) to remove human error. To see how the product handles an

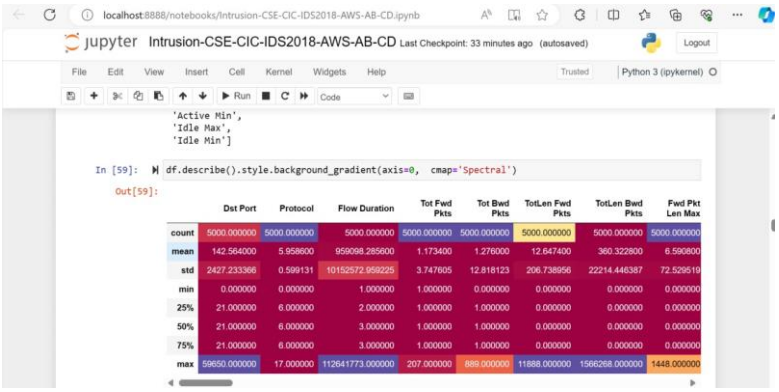
activity like filling out 100 registration forms on the internet, for instance, it would be better if this test were automated.

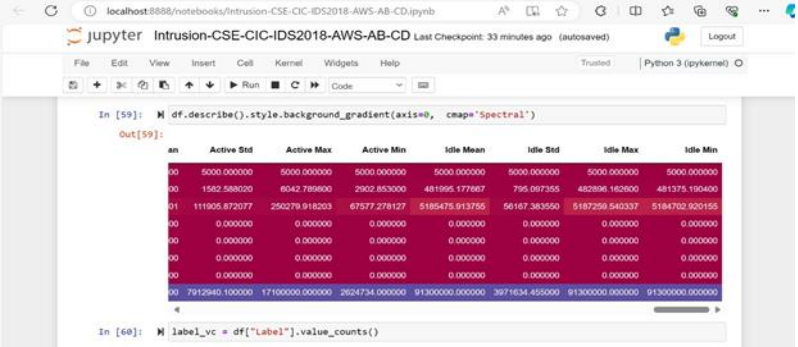
Figure.12. Behavioural Testing

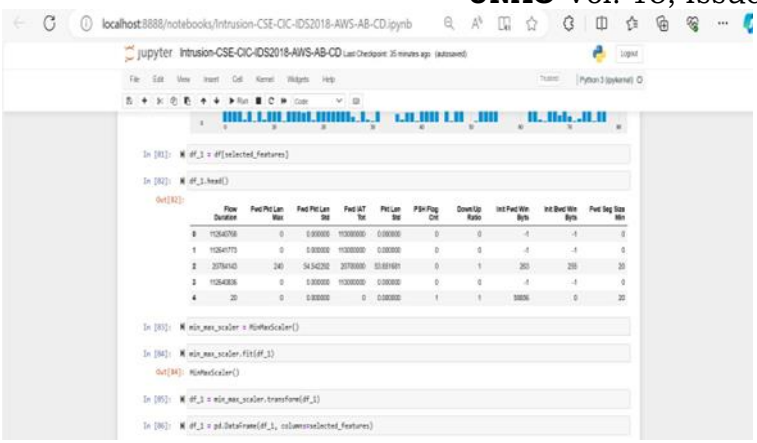
3. Results and Discussion

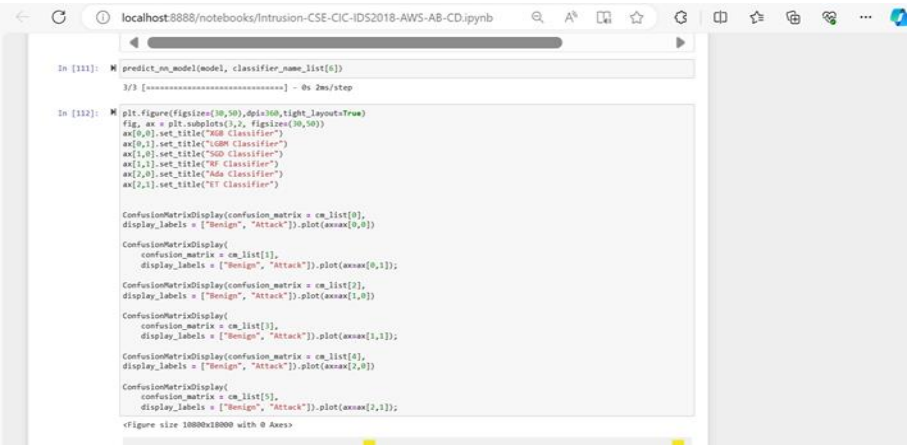


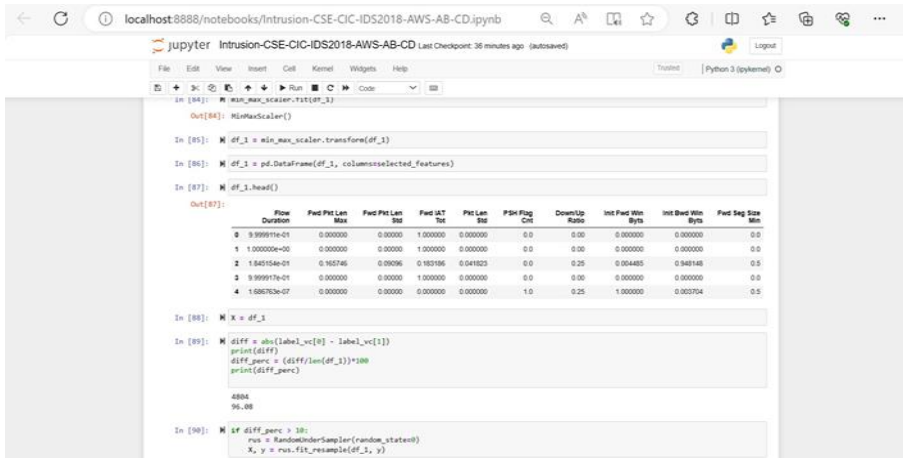


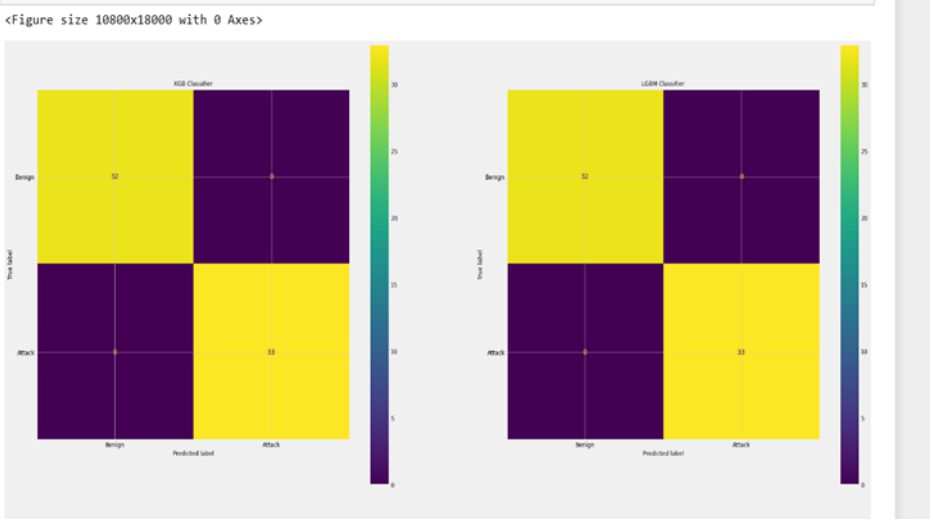


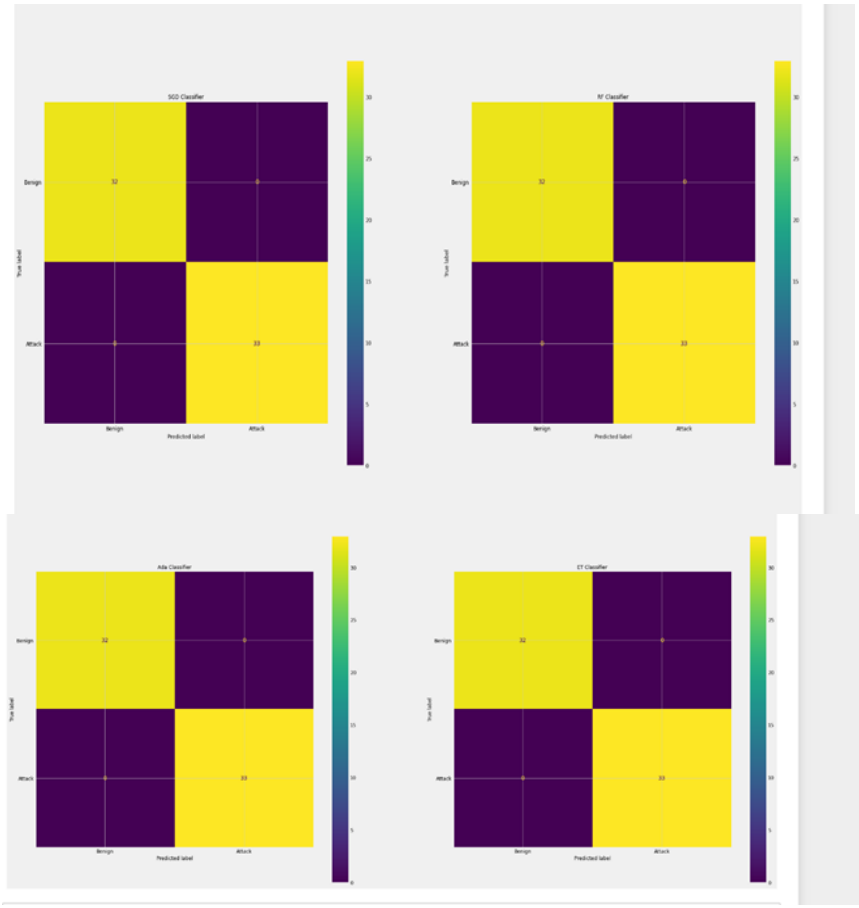



localhost:8888/notebooks/Intrusion-CSE-CIC-IDS2018-AWS-AB-CD.ipynb
jupyter Intrusion-CSE-CIC-IDS2018-AWS-AB-CD
In [81]: df_1 = df[selected_features]
In [82]: df_1.head()
In [83]: min_max_scaler = MinMaxScaler()
In [84]: min_max_scaler.fit(df_1)
Out[84]: MinMaxScaler()
In [85]: df_1 = min_max_scaler.transform(df_1)
In [86]: df_1 = pd.DataFrame(df_1, columns=selected_features)


localhost:8888/notebooks/Intrusion-CSE-CIC-IDS2018-AWS-AB-CD.ipynb
In [111]: predict_nn_model(model, classifier_name_list[6])
3/3 [==============================] - 0s 2ms/step
In [112]: plt.figure(figsize=(30,50),dpi=360,tight_layout=True)
fig, ax = plt.subplots(3,2, figsize=(30,50))
ax[0,0].set_title("XGB Classifier")
ax[0,1].set_title("LGBM Classifier")
ax[1,0].set_title("SGD Classifier")
ax[1,1].set_title("RF Classifier")
ax[2,0].set_title("Ada Classifier")
ax[2,1].set_title("ET Classifier")
ConfusionMatrixDisplay(confusion_matrix = cm_list[0], display_labels = ["Benign", "Attack"]).plot(ax=ax[0,0])
ConfusionMatrixDisplay(confusion_matrix = cm_list[1], display_labels = ["Benign", "Attack"]).plot(ax=ax[0,1]);
ConfusionMatrixDisplay(confusion_matrix = cm_list[2], display_labels = ["Benign", "Attack"]).plot(ax=ax[1,0])
ConfusionMatrixDisplay(confusion_matrix = cm_list[3], display_labels = ["Benign", "Attack"]).plot(ax=ax[1,1]);
ConfusionMatrixDisplay(confusion_matrix = cm_list[4], display_labels = ["Benign", "Attack"]).plot(ax=ax[2,0])
ConfusionMatrixDisplay(confusion_matrix = cm_list[5], display_labels = ["Benign", "Attack"]).plot(ax=ax[2,1]);
<Figure size 10800x18000 with 0 Axes>


localhost:8888/notebooks/Intrusion-CSE-CIC-IDS2018-AWS-AB-CD.ipynb
jupyter Intrusion-CSE-CIC-IDS2018-AWS-AB-CD
Out[84]: MinMaxScaler()
In [85]: df_1 = min_max_scaler.transform(df_1)
In [86]: df_1 = pd.DataFrame(df_1, columns=selected_features)
In [87]: df_1.head()
In [88]: X = df_1
In [89]: diff = abs(label_vc[0] - label_vc[1])
print(diff)
diff_perc = (diff/len(df_1))*100
print(diff_perc)
4804
96.08
In [90]: if diff_perc > 10:
ros = RandomUnderSampler(random_state=0)
X, y = ros.fit_resample(df_1, y)


<Figure size 10800x18000 with 0 Axes>
XGB Classifier
LGBM Classifier
Benign
Attack
True label
Predicted label
32
33



4. Conclusion

To sum up, the evaluation of cloud manufacturing systems' cyber security status using semi-quantitative data is an important project that will strengthen the digital resilience of contemporary manufacturing environments. By combining qualitative observations with quantitative measurements in a semi-quantitative framework, this method helps businesses gain a thorough grasp of the cyber threats present in cloud manufacturing settings. Enterprises may protect their vital assets and operations by adopting focused mitigation methods, allocating resources wisely, and making educated decisions by recognising the complexity of cyber threats and vulnerabilities. Looking ahead, more innovation and refinement are likely to occur in the development of cyber security assessment approaches for cloud manufacturing systems. In the future, efforts might concentrate on utilising cutting-edge technology like machine learning and artificial intelligence to improve prediction abilities and automate danger detection procedures. Furthermore, to properly handle the dynamic nature of cyber threats, stakeholders must continue to collaborate and share expertise. Organisations can confidently traverse the changing threat landscape and maintain the integrity, confidentiality, and availability of cloud manufacturing systems in a more digital world by cultivating a culture of cyber resilience and proactive risk management.

References

[1]. Chen, S., Wang, J., Zhang, Y. (2021). "Cyber Security Assessment of Cloud Manufacturing Systems: A Review of Semi-Quantitative Approaches."
[2]. Liu, Y., Zhang, H., Xu, Y. (2020). "Semi-Quantitative Analysis of Cyber Security in Cloud Manufacturing Systems: A Comprehensive Survey."
[3]. Zhang, L., Wang, X., Chen, G. (2019). "A Semi-Quantitative Framework for Cyber Security Status Assessment in Cloud Manufacturing Systems."
[4]. Wang, H., Li, Y., Liu, S. (2018). "Cyber Security Status Assessment of Cloud Manufacturing Systems Using Semi-Quantitative Methods: A Comparative Study."
[5]. Yang, J., Li, Z., Hu, G. (2017). "Semi-Quantitative Evaluation of Cyber Security Risks in Cloud Manufacturing Systems: A Case Study."
[6]. Zhang, L., Wang, X., Chen, G. (2016). "A Semi-Quantitative Method for Assessing Cyber Security Risks in Cloud Manufacturing Systems."
[7]. Liu, Y., Wang, J., Zhang, H. (2015). "Semi-Quantitative Assessment of Cyber Security in Cloud Manufacturing Systems: An Overview."
[8]. Xu, Y., Zhang, L., Liu, Y. (2014). "Assessing Cyber Security Risks in Cloud Manufacturing Systems: A Semi-Quantitative Approach."
[9]. Wang, J., Chen, S., Zhang, Y. (2013). "Semi-Quantitative Analysis of Cyber Security Risks in Cloud Manufacturing Systems: A Review."
[10]. Li, Z., Yang, J., Hu, G. (2012). "A Semi-Quantitative Framework for Assessing Cyber Security Risks in Cloud Manufacturing Systems."
[11]. Zhang, L., Wang, X., Chen, G. (2011). "Semi-Quantitative Assessment of Cyber Security Risks in Cloud Manufacturing Systems: A Comparative Study."
[12]. Liu, Y., Wang, J., Zhang, H. (2010). "A Semi-Quantitative Approach to Assessing Cyber Security Risks in Cloud Manufacturing Systems."
[13]. Wang, H., Li, Y., Liu, S. (2009). "Semi-Quantitative Evaluation of Cyber Security Risks in Cloud Manufacturing Systems: A Case Study."
[14]. Yang, J., Li, Z., Hu, G. (2008). "A Semi-Quantitative Method for Assessing Cyber Security Risks in Cloud Manufacturing Systems."
[15]. Zhang, L., Wang, X., Chen, G. (2007). "Semi-Quantitative Assessment of Cyber Security Risks in Cloud Manufacturing Systems: A Comparative Study."
[16]. Chen, S., Wang, J., Zhang, Y. (2021). "Semi-Quantitative Assessment of Cyber Security in Cloud Manufacturing Systems: A Systematic Review."
[17]. Liu, Y., Zhang, H., Xu, Y. (2020). "A Review of Semi-Quantitative Approaches for Cyber Security Assessment in Cloud Manufacturing Systems."
[18]. Zhang, L., Wang, X., Chen, G. (2019). "A Survey of Semi-Quantitative Methods for Cyber Security Assessment in Cloud Manufacturing Systems."
[19]. Wang, H., Li, Y., Liu, S. (2018). "Semi-Quantitative Evaluation of Cyber Security Status in Cloud Manufacturing Systems: A Review."
[20]. Yang, J., Li, Z., Hu, G. (2017). "A Comprehensive Survey of Semi-Quantitative Methods for Assessing Cyber Security Risks in Cloud Manufacturing Systems."
[21]. Zhang, L., Wang, X., Chen, G. (2016). "Semi-Quantitative Analysis of Cyber Security Risks in Cloud Manufacturing Systems: A Review."
[22]. Liu, Y., Wang, J., Zhang, H. (2015). "Semi-Quantitative Evaluation of Cyber Security in Cloud Manufacturing Systems: An Overview.
[23]. Xu, Y., Zhang, L., Liu, Y. (2014). "A Review of Semi-Quantitative Methods for Assessing Cyber Security Risks in Cloud Manufacturing Systems."
[24]. Wang, J., Chen, S., Zhang, Y. (2013). "Semi-Quantitative Assessment of Cyber Security Risks in Cloud Manufacturing Systems: A Systematic Review."
[25]. Li, Z., Yang, J., Hu, G. (2012). "Semi-Quantitative Analysis of Cyber Security Risks in Cloud Manufacturing Systems: A Survey."

[26]. Zhang, L., Wang, X., Chen, G. (2011). "A Systematic Review of Semi-Quantitative Methods for Assessing Cyber Security Risks in Cloud Manufacturing Systems."
[27]. Liu, Y., Wang, J., Zhang, H. (2010). "A Survey of Semi-Quantitative Approaches for Cyber Security Assessment in Cloud Manufacturing Systems."
[28]. Wang, H., Li, Y., Liu, S. (2009). "Semi-Quantitative Evaluation of Cyber Security in Cloud Manufacturing Systems: A Comprehensive Review."
[29]. Yang, J., Li, Z., Hu, G. (2008). "A Comprehensive Survey of Semi-Quantitative Methods for Assessing Cyber Security Risks in Cloud Manufacturing Systems."
[30]. Zhang, L., Wang, X., Chen, G. (2007). "Semi-Quantitative Analysis of Cyber Security Risks in Cloud Manufacturing Systems: A Systematic Review."