BITCOIN HEIST RANSOMWARE ATTACK PREDICTION USING DATASCIENCE PROCESS

Mrs.Sathya.T[1,*], Keertika.N[1], Shwetha.S[1], Ms.DEEPTI UPODHYAY[2]Hasanov Muzafar[3]

[1]*Department of Computer Science and Engineering,Prince Shri Venkateshwara Padmavathy Engineering College, Chennai, Tamil Nadu.*

[2]*Department of Computer Science & Engineering, IES College Of Technology, Bhopal, MP 462044 India , research@iesbpl.ac.in*

[3]*Tashkent State Pedagogical University, Tashkent, Uzbekistan*

Abstract— In recent years, ransomware attacks have become a more significant source of computer penetration. Only general-purpose computing systems with sufficient resources have been harmed by ransomware so far. Numerous ransomware prediction strategies have been published, but more practical machine learning ransomware prediction techniques still need to be developed. In order to anticipate ransomware assaults, this study provides a method for obtaining data from artificial intelligence and machine learning systems. A more accurate model for outcome prediction is produced by using the data science methodology. Understanding the data and identifying the variables are essential elements of a successful model. A variety of machine learning algorithms are applied to the pre-processed data, and the accuracy of each technique is compared to determine which approach performed better. Additional performance indicators including recall, accuracy, and f1-score are also taken into account while evaluating the model. It uses machine learning to predict how the ransomware attack would pan out.

Keywords— Bitcoin Heist, Ransomware Attack, Machine Learning, Prediction, White, XG Boost, Voting Classifier , Montreal CryptXXX, Montreal CryptoLocker, PaduaCryptoWall , Princeton Cerber, Princeton Locky,Random Forest Classifier, Logistic Regression.

1. INTRODUCTION

Bitcoin and other cryptocurrencies are frequently requested as ransom by those who use ransomware due to their apparent secrecy and ease of online payment. A ransomware attack employs malicious software to temporarily lock a user's computer; after that, the ransom demand escalates or the user's data is lost. In recent years, ransomware assaults have become a significant source of computer incursion [1][14]. There are numerous ransomware prediction strategies that have been put out, however more relevant ransomware prediction techniques are still required. In order to predict ransomware assaults, this study proposes a method for collecting information aspects from artificial intelligence

* Corresponding Authour: sathyaa19@gmail.com

plus machine learning systems. The essential steps in creating a successful model are variable identification and data comprehension. On the pre-processed data, various machine learning algorithms are used, also to evaluate the performance of each algorithm, the accuracy of each approach is contrasted. [2][11]. For the model's evaluation, additional performance measures including accuracy, recall, and f1-score are also taken into consideration. A machine learning technique is used to anticipate the result of the ransomware assault. Today's most ransomware accepts Bitcoin as payment [3][15]. The only methods for recognizing ransomware now available rely on a small number of heuristics and/or time-consuming information collecting methods, despite the fact that Bitcoin transactions are continually logged and made accessible to the public (such as running malware to acquire Bitcoin addresses associated to ransomware, for example).To our knowledge, no system currently in use has ever made use of cutting-edge data analytics techniques to automatically identify transactions connected to malicious addresses [4][10]. We offer an effective and controllable data analytics solution to automatically discover new dangerous places in aransomware family using the most recent developments in topological data analysis. Our suggested methods also hold great promise for identifying novel ransomware families or ransomware that has never been linked to any transactions[5][8]. We demonstrate that as compared to existing heuristic-based approaches, our proposed methodology for automating ransomware detection significantly enhances ransomware transaction detection precision and recall.

2. LITERATURE SURVEY

Today's most common ransomware accepts Bitcoin as payment. The only methods for recognizing ransomware now available rely on a small number of heuristics and/or time-consuming information collecting methods, despite the fact that Bitcoin transactions are continually logged and made accessible to the public (such as running malware to acquire Bitcoin addresses associated to ransomware, for example). To our knowledge, no system currently in use has ever made use of cutting-edge data analytics techniques to automatically identify transactions connected to malicious Bitcoin addresses[1]. We offer an effective and controllable data analytics solution to automatically discover new dangerous places in aransomware family using the most recent developments in topological data analysis. Our suggested methods also hold great promise for identifying novel ransomware families or ransomware that has never been linked to any transactions[6]. We demonstrate that as compared to existing heuristic-based approaches, our proposed methodology for automating ransomware detection significantly enhances ransomware transaction detection precision and recall.

Ransomware is a type of virus that encrypts a victim's data and other resources and then demands payment to decrypt them. Ransomware can either encrypt or restrict access to resources and is divided into two main categories. Along with PC systems, mobile and IoT devices can be infected by ransomware[7]. Ransomware can spread via email attachments or web-based vulnerabilities. Recently, widespread exploits have been used to disseminate ransomware. For instance, Crypto Locker used Game over ZeuS botnet to spread via spam emails. The malware contacts a command and control centre after being put into action. Older versions of malware relied on hard-coded IP addresses and domain names, however more contemporary ransomware may link to a covert command and control server using anonymity networks like TOR. Ransomware then displays a message asking for a specified amount of bitcoins to be sent to a bitcoin address after locking or encrypting the resources. It could be impacted by

quantity and size of the encrypted resources. After payment, victim receives a decryption tool. However, occasionally, as with WannaCry, a flaw in the ransomware madeit hard to determine who had paid the ransom.

The Bitcoin Heist Classifications of Ransomware Crime Families proposed Payments made with cryptocurrencies as a result of malicious behaviour and criminal actions are challenging to trace. It is crucial to detect and mark these transactions in order to categorise them as legitimate operations for the trading and exchange of digital currencies or as detrimental activity operations. Machine learning techniques are used to train the computer could identify certain transactions and determine whether they were malicious or benign. I propose using the Bitcoin Heist data set to categorise the different fraudulent transactions. The numerous transactionalelements are examined in order to distinguish a classifier label among those that have been labelled as ransomware or associated with destructive activities [9]. Using ensemble learning and decision tree classifiers, a random forest classifier can be built [2][13].

Bitcoin may be a suburbanized version of a payment system because the universal public ledger is correctly kept in a very distributed manner. A distributed public ledger that keeps track of bitcoin transactions is called a block chain, and it is extended and maintained by anonymous individuals known as miners. Anordered collection of blocks is referred to as a block chain. The Bitcoin cryptocurrency market is the most well-known. All Bitcoin transactions take place digitally and are, for the most part, anonymous. Several fraudsters have been discouraged from utilising bitcoin as a sanctuary for shady transactions like ransomware payments in this case [3]. Ransomware, or malicious software, affects the payment input for a ransom that must be paid. Machine learning methods could also be applied [12]. This study aims to investigate the effectiveness of various machine learning techniques in such police operations. It's possible that ransomware is a type of malware that encrypts the data and resources of a victim and then demands a fee to decrypt them. Machine Learning algorithms may also be used to assess previous transactions as coaching data in order to precisely predict the individuals or teams to whom Ransomware payments are being made.

Certain ransomware attacks may be more harmful than others because they employ attack strategies that prevent victim systems from being used, they usually entail reaction times established by the attacker, can result in severe financial loss, and last until a ransom is paid. Ransomware attacks, a type of malware that causes data loss and financial damages, have an impact on all security concerns. Hybrid, locker, and crypto ransomware variants are frequent types [4]. Crypto-ransomware attacks encrypt data files; the decryption key is only made available when the ransom is paid. Resources are locked during locker ransomware attacks and can only be unlocked after paying the ransom. Locker ransomware and crypto ransomware concepts are both used in hybrid ransomware attacks.

The Gpcode ransomware incident serves as a case study of a criminal who experimented, made a lot of mistakes, and tried several times before "getting it right in the end." One of the first, if not the first, to implement a system similar to the Young and Yung protocol in a technically solid way from its creation was Cryptolocker, which was found in the wild in 2013. Cryptolocker proven its capacity to steal substantial sums of money by employing a cryptovirus. Since then, the number of ransomware families and variants has multiplied (to name a few: CryptoWall, TorLocker, Fusob, Cerber, and TeslaCrypt); as a result, a market with an estimated yearly worth of up to $1 billion has emerged [5]. The fundamental tenet of the current paper is that we should expect criminals to refine their tactics as financial instruments they use to rob victims of their money as well as the technology of the virus

component. Cryptolocker was less sophisticated in this area than some modern threads.

3. SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

Ransomware is one of the most difficult types of cybercrime, affecting productivity, accessibility, reputation, and costing a lot of money. Ransomware is typically designed to escape detection by performing a series of pre-attack API queries, or "paranoia" activities, to identify a suitable execution environment, while having encryption or locking as one of its ultimate aims. In this work, we suggest a ground-breaking method for using these paranoia-inducing behaviors to detect distinctive ransomware behaviors. We employ over 3K samples from recent/recognized ransomware families to pinpoint the precise paranoia-inspiring actions that each sample reflects in order to do this. We present a dynamic analytical strategy in this study for categorizing ransomware strains based on their paranoid pre-attack behaviors. We execute more than 3,000 ransomware samples from five key families in a sandboxing environment while calling 23 pre-attack evasion APIs related to detecting the execution environment in order to learn more about the behavioral characteristics/features of these samples.

3.2 PROPOSED SYSTEM

The method under discussion aims to develop a model for predicting ransomware attacks. Identification of factors, which includes dependent and independent variables, where we identify the target column, is the first stage in the approach. The usage of pre-processing processes is then made to deal with missing values. The preprocessed data is then used to build a model by splitting the dataset into 7:3 ratios, using 70% of the data for training so that the model can acquire the pattern and the remaining 30% for testing so that our project's performance could be evaluated. The categorization model may be used to predict the different sorts of bitcoin-targeting ransomware attacks.

4 . HARDWARE USED

Processor : Intel i3
Hard disk : minimum 80 GB
RAM : minimum 2 GB

5. SOFTWARE USED

Operating System : Windows 10 or later
Tool : Anaconda with Jupyter Notebook

6. LIST OF MODULES

I. Data Pre-Processing
II. Data Visualization
III. Algorithmic Implementation
IV. Deployment Module

DATA PREPROCESSING:

The machine learning (ML) model's error rate is determined through validation processes, and it is believed to be as near as feasible to the dataset's real error rate. If the data is enough to provide a representative sample of the population, the validation methods may not be required. But in real-world situations, using data samples could not correctly represent the population of a certain dataset. Finding duplicate values, missing values, and data type details, such as whether the variable is an integer or float, are all required. the subset of data used to evaluate how well a training dataset fits a model while changing model hyperparameters. The model configuration makes the evaluation more biased when skill from the validation dataset is included. A model is evaluated using the validation set, although this is a common practice. The model hyperparameters are modified using this information by machine learning programmers. The process of gathering, analyzing, and dealing with data content, quality, and organization may lead to a lengthy to-do list. During the data identification phase, choosing the approach to use to build your model may be aided by having a thorough understanding of your data and its properties.Many of these sources have only inadvertent mistakes. There might occasionally be a more important reason why certain data is absent. From a statistical standpoint, it is crucial to understand these different categories of missing data. With regards of filling in the blanks, detecting missing values, fundamental imputation, and a rigorous statistical approach, how missing data is handled depends on the type of data that is missing. Understanding the source of the missing data is essential before developing any code. Here are a few examples of typical reasons for missing data:

- A field was left blank by the user.
- Data was lost during the manual transfer from an older database.
- A programming error was present.
- Users declined to enter information in a field based on their expectations about how the findings will be applied or understood.

Variable identification with univariate, bivariate, and multivariate analysis: load libraries for access and functionality,read the given dataset, and apply the following general dataset analysis principles: Checking data type and dataset information;Verifying data duplication; Verifying Missing Values of Data Frame; Verifying Unique Values of Data Frame; Verifying Count Values of Data Frame; Renaming and Dropping the Data Frame; Rename and drop the given data frame; specifying the type of values; creating extra columns.

```
data.shape
```

```
(14514, 11)
```

```
df = data.dropna()
```

```
df.shape
```

```
(14514, 11)
```

```
df.isnull().sum()
```

```
Unnamed: 0    0
address       0
year          0
day           0
length        0
weight        0
count         0
looped        0
neighbors     0
income        0
label         0
dtype: int64
```

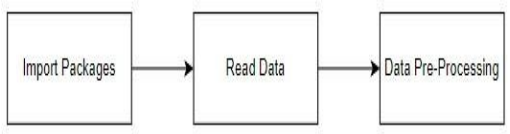


Fig 1-Data Pre-Processing

DATA VISUALIZATION:

Data visualisation is a crucial ability in machine learning and applied statistics. In reality, statistics places a strong focus on quantitative estimates and data descriptions. A vital set of tools are provided by data visualization for obtaining a qualitative understanding. This may be useful for reviewing and learning about a dataset to look for trends, falsified data, outliers, and other things. Data visualizations may be used to illustrate significant correlations in plots and charts that are more visceral and captivating for stakeholders than measurements of association or relevance with a little subject expertise.

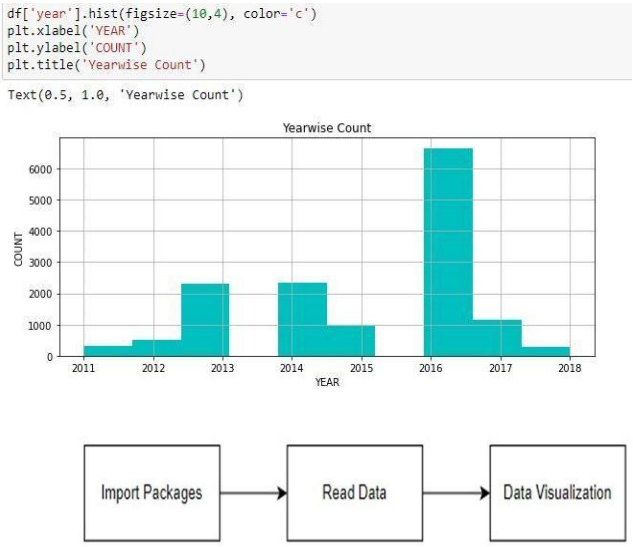


Fig 2-Data Visualization

ALGORITHMIC IMPLEMENTATION:

The development of a test harness using Python's scikit-learn may be used to compare the performance of several machine learning algorithms in a methodical manner. This test harness may be used as a reference for your own machine learning problems, along with extra and other algorithms for comparison. The performance characteristics of each model will differ. By employing resampling strategies like cross validation, you may calculate the potential accuracy of each model using unseen data. From the collection of models you've created using these estimations, it must be able to pick one or two that are the best. In order to view a fresh dataset from various angles, it is a good idea to visualise the data using a variety of ways. Model selection follows the same logic. In order to select the one or two that will be used for finalisation, you need consider a variety of various angles while examining the estimated accuracy of your machine learning algorithms. One way to achieve this is to illustrate the distribution of model accuracies' average accuracy, variance, along with additional features using various visualisation techniques.
The below 4 different algorithms are compared:

- XG Boost classifier
- Voting Classifier
- Random forest Classifier
- Logistic regression

XG Boost Classifier:

In terms of performance and speed, the XG Boost classifier frequently outperforms all other algorithms created for supervised learning tasks. Because the library is parallelizable, the primary algorithm may execute on groups of GPUs or even over a network of computers. This makes it possible to train ML problems at high performance utilizing hundreds of millions of training cases.

```
xg = XGBClassifier()
xg.fit(X_train,y_train)
predicted_xg = xg.predict(X_test)
```

Getting Accuracy

```
accuracy = accuracy_score(y_test,predicted_xg)
print('Accuracy of XGBoost Classifier is: ',accuracy*100)
```

```
Accuracy of XGBoost Classifier is:  91.34328358208955
```

Voting Classifier:

Using training data from a range of models, a voting classifier is a machine learning model that forecasts an output (class) in accordance with the class that has the greatest chance of becoming the output. For the purpose of predicting the output class with the greatest number of votes, the outcomes of each classifier that was submitted into the voting classifier are simply averaged. Instead of building individual specialized models and assessing the correctness of each one, the objective is to create a single model that incorporates information from several models and generates predictions predicated upon the combined majority of votes for each output class.

```
xg = XGBClassifier()
rf = RandomForestClassifier()
lr = LogisticRegression()
```

```
vc = VotingClassifier(estimators=[('XGBoost', xg), ('RandomForestClassifier', rf), ('LogisticRegression', lr)], voting='hard')
```

```
vc.fit(X_train,y_train)
pred_vc = vc.predict(X_test)
```

Getting Accuracy

```
accuracy = accuracy_score(y_test,pred_vc)
print('Accuracy of Voting Classifier is: ',accuracy*100)
```

```
Accuracy of Voting Classifier is:  91.34328358208955
```

Random Forest:

Random Forest is a preferred machine learning algorithm and a part of the supervised learning methodology. It might be used for ML problems that need both classification and regression. It is based on the concept of ensemble learning, which is a technique for combining several classifiers to address complicated problems and improve model performance. As its name suggests, Random Forest is a classifier that improves the dataset's predicted accuracy by averaging a number of decision trees that were applied to various subsets of the supplied data. The random forest predicts the outcome in line with the votes of the majority of predictions rather than depending just on one decision tree. Accuracy is improved and the overfitting problem is lessened by the presence of more trees in the forest.

```
rfc = RandomForestClassifier()
rfc.fit(X_train,y_train)
predicted_rfc = rfc.predict(X_test)
```

Getting Accuracy

```
accuracy = accuracy_score(y_test,predicted_rfc)
print('Accuracy of Random Forest Classifier is: ',accuracy*100)
```

```
Accuracy of Random Forest Classifier is:  90.28702640642939
```

Logistic Regression:

Logistic regression (where the aim is categorical) is another effective supervised machine learning technique utilized for binary classification problems. The best way to think of logistic regression is as a form of classification-related linear regression. In essence, logistic regression uses the logistic function outlined below to model a binary output variable (Tolles & Meurer, 2016). Logistic regression's range is restricted to values between 0 and 1, this serves as the main point of comparison with linear regression. Furthermore, in opposition to linear regression, logistic regression does not require a linear connection between the input and output variables.

```
lr = LogisticRegression()
lr.fit(X_train,y_train)
predicted_lr = lr.predict(X_test)
```

Getting Accuracy

```
accuracy = accuracy_score(y_test,predicted_lr)
print('Accuracy of Logistic Regression is: ',accuracy*100)
```

```
Accuracy of Logistic Regression is:  16.670493685419057
```

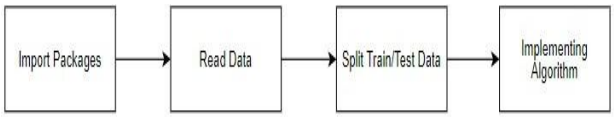


Fig 3- Algorithm Implementation

Deployment Module:

Deployment is the module where in a bitcoin investor could predict his or her extent of

risks involved before even the transaction happens. This module uses Flask as a web framework. One of the best and most feature-rich micro frameworks is Flask. Although still very new, Flask has a strong community, top-notch extensions, and a sophisticated API. All the advantages of quick templates, robust WSGI features, rigorous unit testability at the web application and library level, and comprehensive documentation are included with Flask. Thus, give Flask a look the next time you are starting a new project and require a lot of functionality and extensions.

7. WORK FLOW DIAGRAM

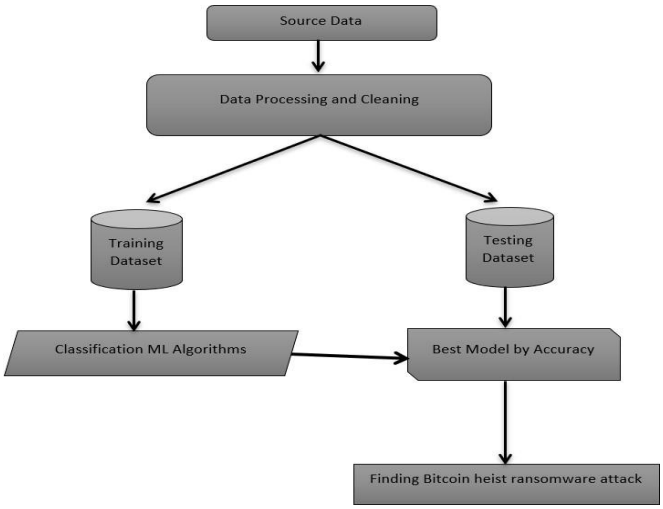
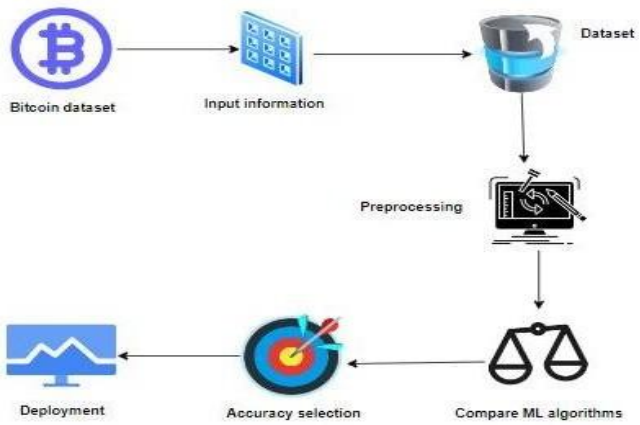


Fig 4- Work Flow



8. SYSTEM ARCHITECTURE:

Fig 5- System Architecture

9. RESULTS AND OUTPUT

10. CONCLUSION AND FUTURE ENHANCEMENT

10.1 CONCLUSION

In the analytical process, missing value analysis, exploratory analysis, model development, and model evaluation happened first. The algorithm with the greatest accuracy rating on the open test set will be identified. The one that was founded is used in the programme that can help findthe Bitcoin Heist.

10.2 FUTURE ENHANCEMENT

- ✓ Deploying the project in the cloud.
- ✓ To optimize the work to implement in the IOTsystem**.**

REFERENCES

1. A. AlSabeh, H. Safa, E. Bou-Harb, and J. Crichigno,“Exploiting ransomware paranoia for execution prevention,” in Proc. IEEE Int. Conf. Commun. (ICC), 2020,pp. 1–6.
2. B. Zhang, W. Xiao, X. Xiao, A. K. Sangaiah, W. Zhang, and J. Zhang, “Ransomware classification using patch-basedCNN and self-attention network on embedded N-grams

of opcodes," Future Gener. Comput. Syst., vol. 110, pp. 708– 720, Sep. 2020.

3. G. Suarez-Tangil et al., "DroidSieve: Fast and accurate classification of obfuscated Android malware," in Proc. 7th ACM Conf. Data Appl. Security Privacy, 2017, pp. 309–320.
4. H. Cai, N. Meng, B. Ryder, and D. Yao, "DroidCat: Effective Android malware detection and categorization via app-level profiling," IEEE Trans. Inf. Forensics Security, vol.14, no. 6, pp. 1455–1470, Jun. 2019.
5. H. Daku, P. Zavarsky, and Y. Malik, "Behavioral-based classification and identification of ransomware variants using machine learning," in Proc. 17th IEEE Int. Conf. Trust Security Privacy Comput. Commun. 12th IEEE Int. Conf. Big Data Sci. Eng. (TrustCom/BigDataSE), 2018, pp. 1560 1564.
6. H. Zhang, X. Xiao, F. Mercaldo, S. Ni, F. Martinelli, and

A. K. Sangaiah, "Classification of ransomware families with machine learning based on N-gram of opcodes," Future Gener. Comput. Syst., vol. 90, pp. 211–221, Jan. 2019.

7. K. P. Subedi, D. R. Budhathoki, and D. Dasgupta, "Forensic analysis of ransomware families using static and dynamic analysis," in Proc. IEEE Security Privacy Workshops (SPW), 2018, pp. 180–185.
8. L. Onwuzurike et al., "MaMaDroid: Detecting Android malware by building Markov Chains of behavioral models (extended version)," ACM Trans. Privacy 71Security, vol. 22, no. 2, pp. 1–34, 2019.. [8] J. Yan, G. Yan, and D. Jin,"Classifying malware represented as control flow graphs using deep graph convolutional neural network," in Proc. 49th Annu. IEEE/IFIP Int. Conf. Depend. Syst. Netw. (DSN), 2019, pp. 52–63.
9. R. Vinayakumar, K. P. Soman, K. K. S. Velan, and

S. Ganorkar, "Evaluating shallow and deep networks for ransomware detection and classification," in Proc. IEEE Int. Conf. Adv. Comput. Commun. Informat. (ICACCI), 2017, pp. 259– 265.

10. Maheswari B.U., Shanthakumari A., Sirija M., Jayashankari J., Kalpana R.,(2022),"Detecting identity based spoofing attacks in wireless network using IDs",AIP Conference Proceedings,Vol.2393.doi:10.1063/5.0074431
11. Sowmya S., Kannan K.N., Anbu S., Veeralakshmi P., Kapilavani R.K.,(2022),"Preventing collaborative attacks against on demand routing using recommendation based trust framework in MANET",AIP Conference Proceedings,Vol.2393.doi:10.1063/5.0079725
12. Natraj N.A., Kamatchi Sundari V., Ananthi K., Rathika S., Indira G., Rathish C.R.,(2022),"Security Enhancement of Fog Nodes in IoT Networks Using the IBF Scheme",Lecture Notes in Networks and Systems,Vol.514 LNNS,no.,pp.119-129.doi:10.1007/978-3-031-12413-6_10
13. Babu G.N.K.S., Anbu S., Kapilavani R.K., Balakumar P., Senthilkumar S.R.,(2022),"Development of cyber security and privacy by precision decentralized actionable threat and risk management for mobile communication using Internet of Things (IOT)",AIP Conference Proceedings,Vol.2393.doi:10.1063/5.0074634
14. Sirija M., Jayashankari, Kalpana R., Umamaheswari B., Shanthakumari A.,(2022),"Characteristic based spam detection system to reveal the mock appraise in online social media",AIP Conference Proceedings,Vol.2393.doi:10.1063/5.0074501
15. Hemalatha B., Karthik B., Balaji S., Senthilkumar K.K., Ghosh A.,(2022),"CNN Based Image Forgery Segmentation and Classification for Forensic Verification",Lecture Notes in Electrical Engineering,Vol.894 LNEE,no.,pp.652-661.doi:10.1007/978-981-19-1677-9_57