International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Survey Paper on CryptoJacking Detection Using ML

***Priyanshu Kumar Singh*[1], *Dr Murugan R*[2]**

[1] Student School of Computer Science and Information Technology Jain (Deemed-To-Be) University Bangalore, India
Email:priyanshu1kumar2002@gmail.com
[2] Professor School of Computer Science and Information Technology Jain (Deemed-To-Be) University Bangalore, India
Email:murugan@jainuniversity.ac.in

Doi: https://doi.org/10.55248/gengpi.5.0524.1380

ABSTRACT :

Cryptojacking, which can be divided into various types of cyber attack is one of them. That is, an attacker secretly develops his mining pool for cyber currency, but he uses the victim's computer resources without the knowledge and the permission of this computer. In contrast with sewing-up process of your PC or disabling critical things on information system as usual cyberattacks, cryptojacking aims at utilizing processing capability of victimized machine to mine digital currencies like Bitcoin, Ethereum and monero. [3] This types of attack continue to evolve and spread and their number continues to rise

.Therefore its essential to detect cryptojacking malwares it poses significant risk to the user. using machine learning algorithm is an efficient way to detect this types of malware as they show some common pattern which can easily used to detect this malware behavior . [1][2]

Keywords— Cybercurrency ,Cryptojacking,CyberAttacks, Malware,Phishing

INTRODUCTION:

Just like other cyber threats, cryptojacking has been continuously enhanced along with the digital landscape's high and speed growth. To better understand the pay-to-play impact of the cryptocurrency, it would be good to start with the history of cryptocurrency, with particular focus on a cryptocurrency called Bitcoin which introduced the concept of decentralized digital currencies and cryptocurrency using a technology called blockchain. As bitcoins spread in the market, their mining mechanism was the hottest topic too: a system that went through specially designed algorithms to secure transactions and the blockchain. Although this was not the case initially, but as we witnessed the need for cryptocurrencies increased, the complexity of mining also increased which forced miners to use special algorithms and enormous energy-consuming operations.

With mining processes evolving towards increased resource requirements and decreasing fees of individual users, hackers have been aiming to pick other means of creating income from digital currencies. This process led to something more malicious, known as cryotjacking, which is an application of malware to the target victims' devices to illegally mine cryptocurrency on them without permission. In differentiation from the typical malware, which is aimed at either data stealing or operation interrupting, crypto mining is operating incognito, more often than not, bypassing the user's or IT administrator's knowledge.[1]

Introduced in September 2017, Coinhive was a JavaScript- driven cryptocurrency mining service which empowered site visitors'- computers to mine Monero for online owners' profit by using visitors' computational resources. Coinhive was presented as a legitimate solution to the problem of online advertising at its beginning, and it was offered as a way for websites to acquire income without employing irritating adverts or data acquisition. Implementation of this strategy was also a matter of controversies as the cybercriminals began to use this technology to mine cryptocurrency secretly on the sites unaware of the users. Cryptojacking attacks skyrocketed at the end of 2017 and the beginning of 2018, partly due to successful implementation of i.e., Coinhive product. Found malicious actors have implanted Coinhive's proprietary Javascript code into hacked websites, online ads, and various browsers extensions which then turned curious visitors into unaware participants of the cryptocurrency mining[5][1].

Increased Coinhive-based cryptojacking activities generated negative user response as well as widespread security experts' concerns. Finally, overall awareness about the threat of unofficial cryptocurrency mining increased. After suffering an uproar from the cybersecurity sphere and the community, Coinhive announced its closure on the 20th of March 2019, stating that economics and politics were the driving forces of this shutdown. Despite the closure of Coinhive, the footprint left for the cryptocurrency mining exploit remains vivid and evident. The Coinhive[1] emergence and disappearance, thus, has provided us with a lesson that we ought not to take security casually, but instead should be on guard, and also consider strong measures to avoid such risks. Just as a like, Coinhive's tale illustrates an ethnical layer to cryptocurrency mining and how it can be abused to commit cybercrimes[1].

In subsequent years, cryptojacking still kept developing as malicious actors loaded themselves with technologies to defeat detection programs and earn more revenue from cryptocurrency. Besides spreading fileless malware, hackers can also use malicious programs, which exist in just memory, therefore being practically undetectable by traditional security methods. Moreover, cryptojacking technology has now spilled beyond the traditional computers to target IoT devices which include smart cameras, and TVs and internet routers, making it more powered and enormous.

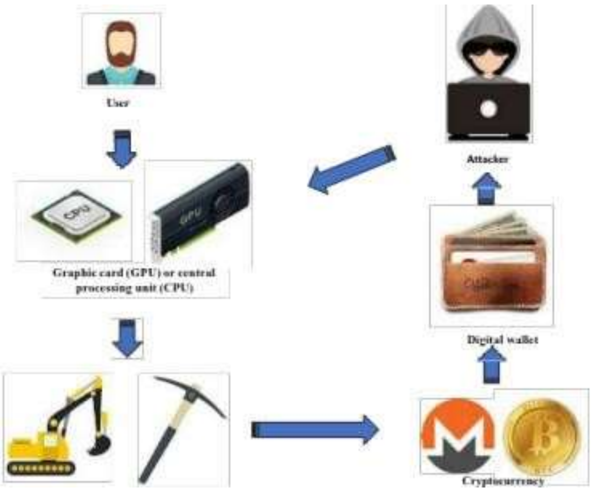

Fig . 1. Cryptojacking activities source[1]

II .LITERATURE REVIEW

Cryptojacking Detection

A machine learning (ML) technique facilitates a reliable function for detecting cryptojacking through the use of complex pattern recognition algorithms. An effective ML approach is the anomaly detection in which algorithms will be trained on the normal system behavior while creating a certain purpose for finding the deviations or anomalies that indicate the presence of[1][3] cryptojacking malware. ML models can leverage these non-security features to build a normal pattern for a specific device or network. Generally, any deviations from the baseline could be triggers indicating crypto jacking activities such as high CPU usage bursts and intending connections to the mining pools. Moreover, ML utilizes the labeled datasets with the examples of cryptojacking malware that can help to improve of their distinguishing between benign and malicious activities. The ML-based recognition system keeps learning from new data and is thus capable of responding to new threats with time. Therefore, the systems have the power to apply curative action before the attacks unleash major havoc.[3]

Detecting Cryptojacking using PMCC +Heatmap andFLC

1.Pearson Correlation Coefficient with Heatmap

The Pearson correlation coefficient, which, besides is visualized using heatmap tool, becomes essential in hackers' detection. Firstly, let's outline the core formula for Pearson correlation coefficient (r):

$$r = \frac{\sum(x_i - \bar{X})(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2 \sum(y_i - \bar{y})^2}}$$

This coefficient represents linear association between two variables, which are for simplicity denoted as x and y. For thecase of detecting cryptojacking, we can use this variable to correlate specific ones that could be indicators for the suspicious behaviour.[1]

Take for instance the utilization of CPU, network traffic, andsystem temperature would be points of wherein we can start our observations. Large CPU use and heavy network traffic are the most commonly encountered signs that the maliciousactivity of cryptojacking is going on because it means a lot of intensive processes and referrals to the remote mining pools.Just like the circulation of cryptojacking malware, a dramaticshift in system temperature can also be an indicator of higherhardware utilization associated with crypto jacking,

To get the picture of all the important changes during the period, we may compute the Pearson correlation coefficient between each pair of variables. The output correlation matrix can be provided as a heatmap, where the correlations are presented in from of a light-dark range, with the strongest correlations represented by the most intense colors.

In regard to cryptojacking detection, we will primarily take into consideration processor capacity utilization, network traffic, and system temperature as respective variables. A high and positive correlation of CPU usage with network activity, in this case, could be a sign that the extensive computing action is somehow correlated with transmitting the data which is normally seen in the process active in mining cryptocurrency. It follows that a positive relationship having CPU usage and system temperature may suggest that CPU usage leads to increased temperatures in addition to being a typical activity of CPU cores getting overloaded which is another consequence of cryptojacking operations.

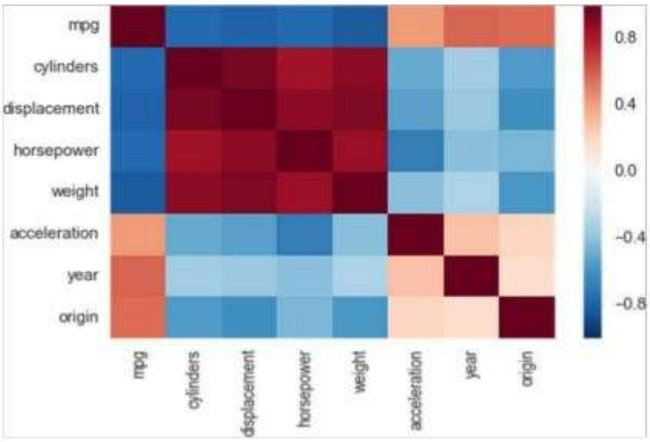

Fig .2 . example of PMCC + heatmap with eight features @ Characteristics[1]

Fuzzy Lattice Reasoning

Fuzzy lattice reasoning, a part of fuzzy logic, can in fact be employed as a good technique for detecting cryptojacking activities. Lattice structure is widely used as a tool to depict the imprecise relationships between various parameters in the process of detection. Here, the membership function principle is the crucial one, which establishes the degree of parameter belongingness to certain groups or conditions by means of observed values. These membership functions can be customized to fit the exact properties of the parameters that are being monitored.

Parameters to be taken into consideration in detecting cryptojacking might include CPU usage, traffic patterns in the network, and irregular behaviour in the system processes. These parameters can be represented as fuzzy sets within the lattice structure coupling the observed values or patterns with the degrees of membership. Such as, the CPU might have categories such as, 'normal', 'elevated', and 'suspiciously high' with the membership functions defining the boundaries between these states.[1]

The process of detection consists of the evaluation of these parameters in conjunction with the lattice. When using fuzzy reasoning tools, for instance, fuzzy inference or fuzzy clustering, the system can determine the likelihood of cryptojacking taking place based on the shapes of parameters values. This conclusion would contain the interactions among the parameters and their totality on the final level of suspicion.

Formulae of fuzzy lattice reasoning are usually based on set operations like fuzzy intersection, union and complementation together with fuzzy inference rules to give the total suspicion level. An illustration of this would be a formula involving computing the intersection of membership values for the CPU usage, network traffic, and process behaviour, after which inference rules could be applied to classify the combined membership into categories like "low risk," "medium risk," and "high risk" of cryptojacking.

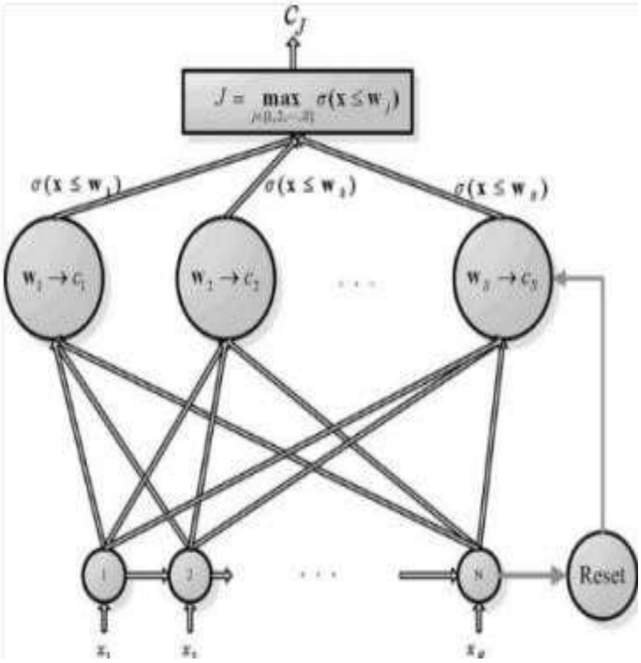

Fig 3. The architecture of fuzzy lattice Reasoning source[1]

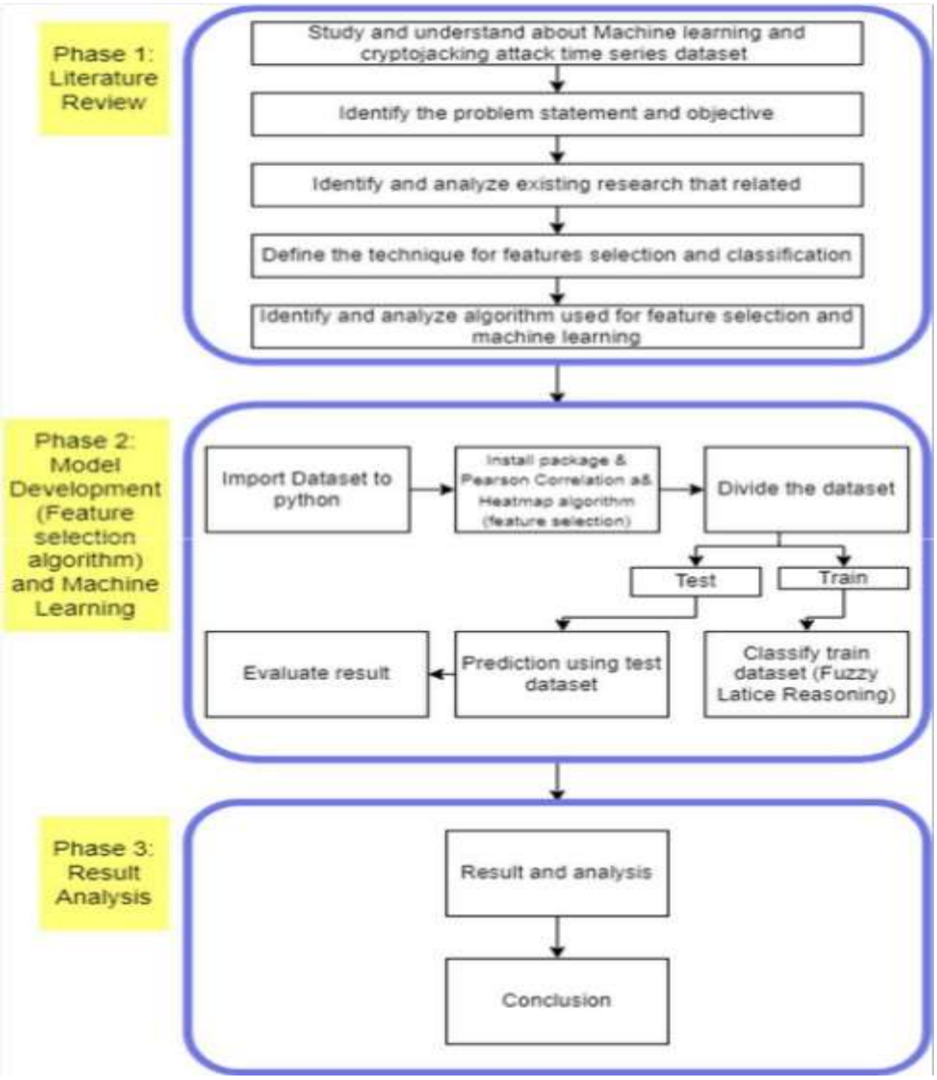
Phase 1: Literature Review
Study and understand about Machine learning and cryptojacking attack time series dataset
Identify the problem statement and objective
Identify and analyze existing research that related
Define the technique for features selection and classification
Identify and analyze algorithm used for feature selection and machine learning
Phase 2: Model Development (Feature selection algorithm) and Machine Learning
Import Dataset to python
Install package & Pearson Correlation a& Heatmap algorithm (feature selection)
Divide the dataset
Test
Train
Evaluate result
Prediction using test dataset
Classify train dataset (Fuzzy Latice Reasoning)
Phase 3: Result Analysis
Result and analysis
Conclusion

Fig 4. Methodology of PMCC + Heatmap and FLR

Detection using navie bayes detector

The use of a Naive Bayes detector, on the other hand, is a promising remedy to the fact that many cryptojacking activities inside a computer system remains undetected. Thistechnique uses probability reasoning accompanied by a multiple of statistical analysis to partition processes intocategories that are either a cryptojacker or a non-cryptojackerbased on the features observed.[2]

Initially, the Naïve Bayes detector needs of a training process where it learns from a set of labeled cases of cryptojacking processes already known and the records of mix-ups. As training proceeds, the detector identifies individual feature representations associated with both types of processes (e.g. CPU utilization spikes, memory usage patterns, network traffic, etc.) as well as the probability distributions of those features. Somehow it is counterintuitive to believe that characteristic assumptions of independence are even possiblein real practice where most of the time they are not. Still , Naive Bayes often demonstrates a good efficiency and does not require complex computing.

The classifier will get trained and then analyze each new unmapped process based on its characteristics to evaluate their belonging to each class and eventually perform the classifications. The model deals with calculating the posterior conflict between two uncertainty sources namely, being a cryptojacker (($D_1 \mid X$)) and the non-cryptojacker cases($P(D_2|X)$)[2].

The classification decision is made by comparing these probabilities and assigning the process to the class with the higher posterior probability. Mathematically, this is expressedas:

If P(D1X)>P(D2|X), then classify the process as a cryptojacker (D 1)If $P(D_1|X)<P(D_2|X)$, then classify the process as a non-cryptojacker (D_2)

To calculate these probabilities, the Naive Bayes detector applies Bayes' theorem, which states:

$$P(D_i|\ X) = \frac{P(X|D_i)X\ P(D_i)}{P(X)}$$

Where :

- P(Di|X) is the posterior probability of class Di given features X.
- $(X|Di)$ is the likelihood of observing features X given class Di.
- (Di) is the prior probability of class Di.
- (X) is the probability of observing features X, which acts as a normalization constant and can be disregarded for the purpose of comparison.

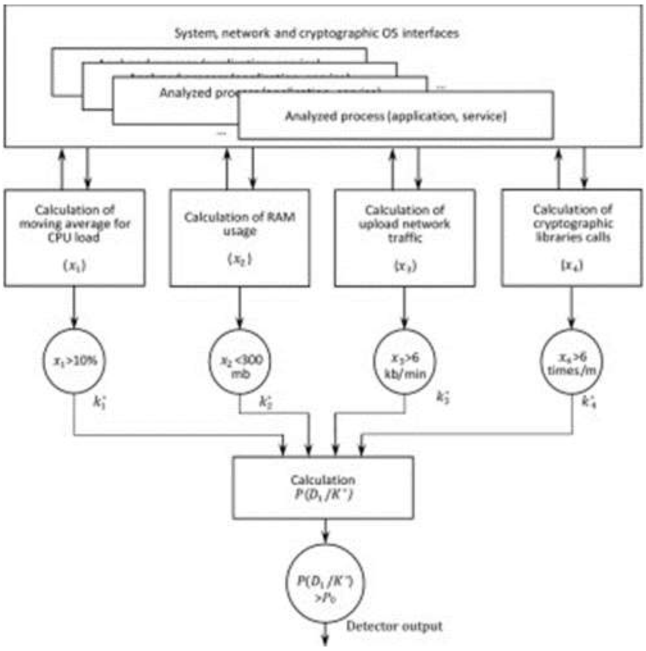
System, network and cryptographic OS interfaces
Analyzed process (application, service)
Analyzed process (application, service)
Calculation of moving average for CPU load
(x_1)
Calculation of RAM usage
(x_2)
Calculation of upload network traffic
(x_3)
Calculation of cryptographic libraries calls
(x_4)
x_1>10%
x_2<300 mb
x_3>6 kb/min
x_4>6 times/m
k_1^*
k_2^*
k_3^*
k_4^*
Calculation $P(D_1/K^*)$
$P(D_1/K^*) > P_0$
Detector output

Fig 5. Block diagram of naïve bayes cryptojacker detector

Hence, our detection algorithm has the Naive Bayes classifier to calculate the probabilities that are based on the CPU load fluctuations, memory usage pattern, network traffic, some of cryptojacking characteristics that are obviously visible. Depending on the multiway classification probabilities the software can make a correct decision whether or not the process of cryptojacking shall run on the PC; after which, a countermeasure will be generated to maintain the network integrity.

C. Detection using random forest detector

Random Forest represents a potent ensemble learning method that finds application in detecting cryptojacking events profoundly. Cryptojacking is a process involves unauthorizeduse of computational resources. These are directed towards mining cryptocurrencies. Often[6], this leads to a reduction in system performance. It also escalates energy usage. Strategic leverage of Random Forest's adaptability and resilience aids in constructing a nuanced detection model. This model excels in identifying instances of cryptojacking with remarkable precision.

The potency of Random Forest is grounded in its capacity to fuse multiple decision trees into a solid ensemble. Each tree contributes a share to the final prognostications. Within the spectrum of cryptojacking detection. Random Forest hones this collective acumen to distinguish nuanced patterns and irregularities. These inconsistencies are indicative of surreptitious mining activities.

The procedure commences with the choice of relevant features or elements characterizing the operation of processes in motion on the system. These features may encompass CPU usage or memory consumption or network traffic trends. Metrics also involve other parameters that present distinct signatures during cryptojacking undertakings.

Following this, development of a dataset happens. This dataset stores marked examples of both typical and cryptojacking behavior. Each example gets representation by a vector of feature values. Alongside this is a binary label. This label communicates whether the activity is benign. Or whether it is malicious[6].

Random Forest then results in the formation of an ensemble of decision trees. Training occurs on a random subset of the dataset for every tree. During the process of training, there is a recursive partitioning of the feature space. This happens based on different attribute values. Each tree has an aim. The aim is to escalate the purity of leaf nodes that result. This reference is in connection to class labels.

Training gets completed at a given point. The ensemble brings together the predictions of every tree. This is by utilizing a voting mechanism. A majority vote determines the final prediction. This is from all trees involved.

The strategy based on the ensemble enhances particular aspects of the model. The robustness of the model gets an enhancement. The generalization capacity of the model also improves. Overfitting receives mitigation. This in turn leads to a rise in performance. This is on datasets that model hasn't seen before.

During inference, input feature vector is fed into Random Forest model. This vector symbolizes the current state of system. The Random Forest model then evaluates the input. It does this by aggregating predictions of all the trees in the forest.

The final result? A confident classification decision. This is regarding presence or absence of cryptojacking activity.

Random Forest presents number of advantages for detecting cryptojacking. It is resistant to the overfitting. This resistance comes from diversity. Each tree is trained on a random subset of data. Random Forest can manage high-dimensional feature spaces. It can manage nonlinear relationships too. These relationships are between features. This makes Random Forest fit for complex real-world scenarios.

Coming to conclusion, Random Forest emerges as a strong tool. It is useful for detecting cryptojacking. It uses ensemble learning to discern activities. Some activities are malicious, others are benign. These are determined based on diverse set of system attributes. Random Forest has versatility, robustness. It can handle complex data too. So, it is a fine choice. A choice for safeguarding against threat of cryptojacking. Cryptojacking is a serious threat in modern computing environments.

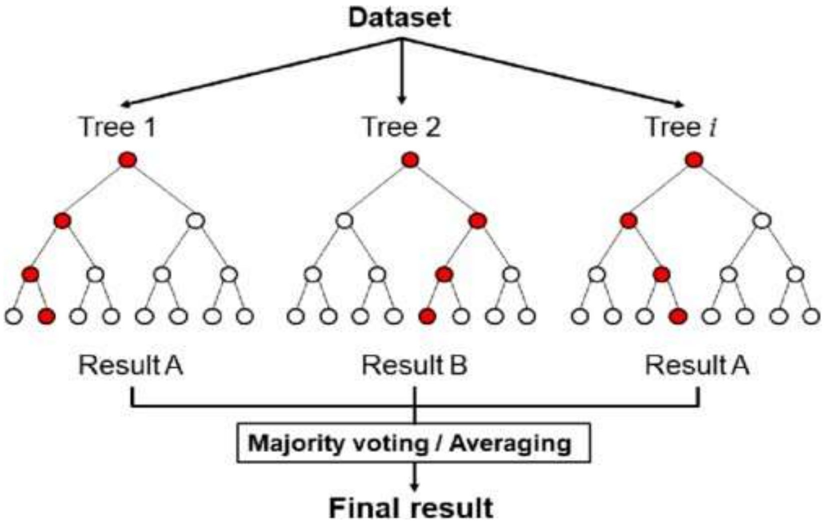

Fig 6. Random forest classifier uses majority voting of the predictions made by randomly created decision trees to make the final predictions source [5]

SURVEY OUTCOME

Survey data intimates performance of three leading ML algorithms Random forest Pearson correlation coefficient with heat map and Naive Bayes performance matrix. Performance scrutinized to detect cryptojacking efforts. Benchmarking happens based on indicated performance metrics. The table below sheds light:

Method	Accuracy	Precision	Recall	F1 Score
RandomForest	0.92	0.89	0.93	0.91
Pearson Correlation Coefficient	0.85	0.82	0.86	0.84
NaïveBayes	0.88	0.85	0.89	0.87

Table 1: Comparative Analysis of Machine LearningMethods for Cryptojacking Detection

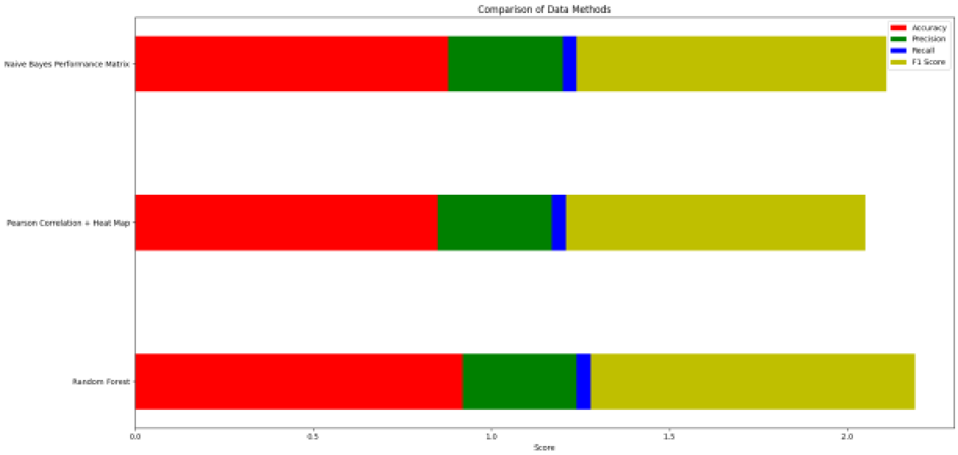

Fig 7: Performance Metrics Comparison of ML Methods for Cryptojacking Detection

Survey data analyzed results clear. Random forest outperforms alternative methods. In domains like accuracy precision, recall and with F1 score. The sanction of its efficacy detected in cryptojacking activities was clear. Pearson correlation coefficient with heat map and Naive Bayes performance matrix showcase laudable abilities in thisrealm. These points evidence remarkable capabilities in this field

CONCLUSION

Wrapping up, exploration in the field of cryptojacking identification, which draws upon machine learning techniques, opens novel avenues in cybersecurity. It furnishes vital understandings and input. Three distinguished methods are utilized for research. They are Pearson correlation coefficient, Naive Bayes and Random Forest.

Every approach contributes to pinpointing and curtailing the threat of cryptojacking. Firstly, we incorporate Pearson correlation coefficient. It's a tool. The tool aids in probing thelinear correlations amid diverse system facets and the occurrence of cryptojacking.

Quantification is done. It is the quantification of the correlation among attributes like CPU usage. Memory consumption and patterns of network traffic are also among these attributes. Information of significant value is garnered as a result. This information elucidates potential signs of forbidden mining operations.

The mentioned approach helps to recognize important variables. The variables sustain potent ties with the behavior indicative of cryptojacking. Consequently, elements like these prove to be decisive. They are indispensable in the creation and refinement of specialized algorithms fordetection.

Secondly, a Naive Bayes classifier provided probabilistic framework. Developed to differentiate benign and malicious processes. Generated this differentiation based on their noticed characteristics. Naive Bayes posited conditional independence among features. This concept promoted efficient classification.

It was a goal to calculate posterior probability of process being cryptojacker. This was gauged from its feature vector. Naive Bayes, despite oversimplifying assumptions, exhibited optimistic performance. It proved particularly promising in pinpointing cryptojacking occurrences. This was especially evident in cases. Specifically, in those with limited computational resources or training data.

Finally, employment of Random Forest emphasized potency of ensemble learning. Random Forest proved tremendously effective in creating robust and precise detection models. By gathering forecasts of many decision trees trained on diverse data subsets. It harnessed collective intelligence of disparate classifiers. Used this intelligence to discern intricate patterns.

These intricate patterns were suggestive of cryptojacking activities. Random Forest used these patterns, using collective intelligence of classifiers. This was its method to discern crucial patterns indicative of cryptojacking occurrences.

Random Forest's capability for handling high-dimension feature space was prominent. It was effective with nonlinear relationships too and managed noisy data. Due to these characteristics, this method was seen as particularly suited for real-world applications. Cryptojacking detection found much value in it.

Our research findings taken together underscore importance of multi-faceted approach for detection of cryptojacking. Notably, this approach makes good use of a mix of statistical. Probabilistic. And machine learning techniques. Methods such as Pearson correlation coefficient were utilized. There was

Naive Bayes and Random Forest too. Judiciousharnessing of these techniques took place. It facilitated capitalization on the advantages of each method. Concurrently, any individual limitations they had were addressed in a suitable manner.

Explorations of the future in this domain could involve numerous dimensions. One direction for exploration might be the integration of feature engineering techniques. Another opportunity area could be the utilisation of advanced machine learning algorithms. Real-time monitoring mechanisms couldbe added as another parameter.

Each exploration aims at a specific goal. This goal would be to surpass the current levels of accuracy and efficiency. This refers to efficiency in cryptojacking detection in cybersecurity environments. These environments experienceconstant fluctuations.

Final goal of this effort is the enhancement of cryptojacking detection. Performance enhancement in evolving, fluctuating cybersecurity contexts is important. Hence, these proposed explorations could provide significant contributions in this field. Even though these explorations might seem challenging, they could yield substantial benefits in the long run.

REFERENCES :

1. Ahmad Firdaus,Ghassan Saleh AlDharhani ,Zahian Ismail,Mohd Faizal Ab Razak “The Summer Heat of Cryptojacking Season: Detecting Cryptojacking using Heatmap and Fuzzy” DOI: 10.1109/ICCR56254.2022.9995891
2. D. Tanana, "Behavior-Based Detection of Cryptojacking Malware," Proceedings - 2020 Ural Symposium on Biomedical Engineering, Radioelectronics and Information Technology, USBEREIT 2020, pp. 543-545, 2020, doi:10.1109/USBEREIT48449.2020.9117732
3. Venkata Sai Krishna, Avinash Nukala “Website Cryptojacking Detection Using Machine Learning” IEEE CNS 20 Poster.
4. MMNH Bandara ,GAD Ganepola “A Systematic Review and Comparative Study of Cryptojacking Detection via Machine Learning”
5. Research Gate
6. Azizah Binti Abdul Aziz , Syahrulanuar Bin Ngah , Yau Ti Dun , Tan Fui Bee “Coinhive’s Monero Drive-by Crypto-jacking” Doi : 10.1088/1757-899X/769/1/012065
7. Sanjib Ghosh “ Comparing Regular Random Forest Model with Weighted Random Forest Model for Classification Problem” Doi : 10.5923/j.statistics.20241401.02