iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 **Issue:** III **Month of publication:** March 2025

DOI: https://doi.org/10.22214/ijraset.2025.67583

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue III Mar 2025- Available at www.ijraset.com

Employing Methods Such as CNN2D and a Voting Classifier to Identify Ransomware Attacks Using Processor and Disk Usage Data

B. Malavika[1], Mr K. Pavan Kumar[2], M. Sri Srujana[3], J. Pardhiv Nag[4], D.Adiseshu Akash[5]
Dept of CSE, Raghu Institute of Technology

Abstract: ***The research tackles the issue of ransomware detection, spotting the shortcomings of current strategies that rely upon system monitoring and information analysis. The objective is to create a dependable and effective detection mechanism for ransomware running on a virtual machine (VM). The data collection objectives precise processor and disk I/O activities for the complete virtual gadget from the host device. The studies is to develop an efficient detection model by means of making use of machine learning (ML), particularly a random forest (RF) classifier. This approach reduces monitoring burden and lessens the threat of data compromise by ransomware. The recommended method reveals robustness against fluctuations in user workloads, addressing a customary difficulty in ransomware detection. with the aid of eschewing incessant oversight of all techniques on the target pc, the version retains its adaptability to diverse user contexts. The task's efficacy is classed via diverse person workloads and 22 ransomware specimens. This venture gives a realistic and effective answer to the chronic ransomware hassle with a reliable detection mechanism. The project employs particular processor and disk I/O activities along gadget gaining knowledge of to reduce tracking overhead, improve detection pace, and keep adaptability to rising ransomware traces. This observe delivered huge upgrades, combining a Convolutional Neural network 2d (CNN2D) and an ensemble model with a voting classifier to enhance ransomware detection accuracy. The voting classifier, such as various machine learning classifiers, carried out an excellent ninety nine% accuracy in final predictions, illustrating the efficacy of version integration for enhanced detection.***

"Index terms: Deep learning, disk statistics, hardware performance counters, machine learning, ransomware, virtual machines".

I. INTRODUCTION

Ransomware is malicious software program that encrypts documents on a targeted pc or restricts get entry to to the device, rendering the device and its facts inoperable. Cyber-attacks employ ransomware attacks to extort financial sources from victims. actors may additionally hire ransomware attacks to damage the key infrastructure in their fighters. these attacks often contain the exfiltration of victims' data to coerce them into paying a ransom or to sell the records at the darkish internet. In 2022, nearly 70% of corporations skilled ransomware attacks. "Every 11 seconds in 2021, ransomware is predicted to attack a company, consumer or gadget every 2 seconds by 2031". The forecast for financial impact in 2021 is $ 20 billion; In 2031, it is expected to be $ 265 billion. The ransomware attack has been examined by several studies since identification.

Based mainly on hash values created by antivirus software programs for identified ransomware, signature -based detection checks the target file comparison tool. However, polymorphic and metamorphic forms of current ransomware can prevent the detection of the-based signature [4], [5]. Thus, the behavioral or running identity of ransomware during operation improves completely signature -based techniques. Behavioral analysis is a dynamic method that focuses on the movements of ransomware - ie a series of actions that follow the patient's computer infection. despite the fact that virus sports fluctuate substantially, ransomware must execute a particular series of operations to unexpectedly encrypt a maximum variety of statistics documents. latest ransomware versions, like LockBit 2.zero, Darkside, and BlackMatter, encrypt best segments (the initial bytes) of documents to expedite the rendering of many documents unreadable. Therefore, the necessity of fast encryption of user data is expected to reveal not only a legitimate use but also the behavior of ransomware in addition. The theory is that a system has to constantly show various kinds of strange activity during the ransomware attack. Ransomware necessitates get admission to to hard pressure documents and makes use of the CPU for facts encryption, leading to multiplied hobby; accurately skilled gadget-gaining knowledge of algorithms may additionally pick out this heightened hobby.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

The implementation of runtime detection on the goal gadget necessitates ongoing surveillance of numerous methods, components, and subsystems, together with the gathering of occasion-related facts and the analysis of this records for aberrant behavior [7], [8]. Ransomware can also try and difficult to understand its operational behavior via producing supplementary techniques and sports. nonetheless, a device under attack demonstrates heightened activity, which can be recognized thru right analysis. Runtime detection is resource-in depth and intrusive when carried out on the target machine, as figuring out the technique related to ransomware may be difficult, necessitating the tracking of many techniques. furthermore, such monitoring is susceptible to being stopped through ransomware engineered to terminate going for walks processes prior to document encryption.

Specialized registers called hardware performance counters (HPCs) track processor and device events for either individual approaches or the complete gadget. Modern CPUs can perform the range of completed instructions, cache misses, and rancid-chip memory accesses among other events. Usually, performance analysis and device software improvement make use of the data gathered on HPCs. More recently, though, studies have looked at its possibilities for "malware detection [9], [10], [11], [12], [13], [14]". Alam et al. [15] made use of HPC records gathered from every technique available for system walks. Monitoring several techniques is unworkable nevertheless since it affects system performance. Pundir et al. [7] compiled data at the system level, however their analysis limited to a "single Windows Virtual Machine (VM)" workload. Detection accuracy should be much affected by changes in the workload of the VM as well as by expanding the range of walking applications.

II. LITERATURE SURVEY

Malware, along with Trojan Horses, worms, and spyware, poses a considerable danger to the internet. We cited that notwithstanding widespread versions in content signatures amongst malware and its editions, they show off positive behavioral characteristics at a better level that extra appropriately disclose the true intention of the malware. four This paintings examines the method of malware behavior extraction, introduces the formal "Malware behavior feature (MBF)" extraction method, and provides a malware detection algorithm primarily based on malicious conduct functions. We successfully created and carried out the "MBF-based totally malware detection device", and the experimental results reveal its functionality to discover freshly emerged unknown malware.

Crypto-ransomware represents a sizeable threat amongst ordinary malware, as it monetarily extorts sufferers by way of unlawfully encrypting their documents, thereby denying get entry to and protecting their files hostage. This ends in annual losses as much as tens of millions of dollars globally. The proliferation of diverse ransomware variations is increasing, possessing the potential to stay away from several antivirus applications and software program-based "malware detection structures that rely on static execution signatures". [7] Thepaper presents a Ranstop, a "hardware-assisted" method to the first detection of crypto-Ransomware infections at Standard CPU. To track micro-architectonic event patterns, Ranstop tracked known and unknown crypto-ranmware variants using data from hardware indications inside the performance monitoring unit of the contemporary CPU. This paper analyzes micro-architectonic activities in hardware during the performance of several ransomware variants and moderate applications by presenting training "a recurrent neural network utilizing long -term short-term memory (LSTM)" models. By means of the global average pool, LSTM [52.54] and the identification of Ranstop produce a temporal chain to extract internal statistical features from HPC data, thereby decreasing the noise. The data analyzes with high demstration collected more than 20 intervals and within 2 milliseconds from the execution, separated each 100 microom as an early identification mechanism properly and rapidly detects ransomware. This detection rate stops any appreciable damage caused by ransomware. Confirmation against innocuous programs with behavioral equality of crypto-Ranmware also shows Ranstop has an average accuracy of 97% in fifty random testing.

Resurfaced as a common type of malware lately, ransomware is targeted on a wide range of victims—from personal users to businesses—for financial gain. Our primary statement concerning modern-day ransomware detection mechanisms is their inability to supply actual-time early warnings, main to the irreversible encryption of several files, at the same time as "publish-encryption strategies (e.g., key extraction, document restoration) showcase diverse limitations. [27], [28] The contemporary detection mechanisms yield a high charge of false positives", failing to ascertain the authentic reason at the back of file adjustments; especially, they cannot differentiate whether or not a substantial alteration in a record is due to ransomware encryption or a benign consumer -initiated operation, inclusive of encryption or compression. This paper [8] presents "RWGuard, a ransomware detection mechanism designed to identify crypto-ransomware in real-time" on a consumer's gadget with the aid of (1) enforcing decoy techniques, (2) meticulously monitoring walking techniques and the document device for malicious sports, and (3) aside from benign record changes from indicators with the aid of analyzing users' encryption behavior We investigate our device using samples from the 14 maximum large ransomware homes up to now [22], [23], [24], [25], [26].

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

At the same time as introducing most effective around 1. Nine percent overhead, our effects reveal that RWGuard excels in actual-time ransomware detection, reaching zero fake negatives and a minimal fake tremendous price of roughly 1.9%.

The boom of computer systems in any subject is followed through the growth of malware in that field. systems, specifically current cell systems, are inundated with viruses, rootkits, adware, spyware, and several different varieties of malware. notwithstanding the presence of "anti-virus software, malware threats" retain to proliferate because of several methods that may avert such protection. currently, assailants exploit vulnerabilities in antivirus software to infiltrate structures. This studies [9] investigates the viability of constructing a hardware-based totally malware detector with current overall performance counters. information from performance counters can be applied to hit upon "malware, and our detection techniques" show resilience to tiny alterations in "malware programs". consequently, after reading a restrained array of versions within a "malware own family on [33, 36] Android ARM and Intel Linux systems", we are capable of discover numerous variations inside that own family. furthermore, our counseled Hardware adjustments allow malware detector to work safely within the system software, hence facilitating antivirus implementations which can be extra honest and much less liable to mistakes than software-primarily based antivirus answers. the integration of robust and relaxed hardware AV tactics has the ability to enhance on-line virus detection.

current research have demonstrated ability in using microarchitectural execution styles for the detection of malware programs. These detectors are categorized as signature-based since they identify malware by means of software execution pattern (signature) to recognized malware styles. This analysis of [10] presents New type of detector-detectors of malware-based anomalies —that work without requiring malware signatures [9], [10], [11], [12], [13], [14], thereby enabling the detection of a larger range of malware, comprising of new varieties. We build profiles of consistent application execution based on overall performance counter facts using unsupervised device getting to know, then spot great aberrations in behavior coming from malware exploitation. Our results show that near truth detection of attacks on widely used packages such as "Internet Explorer and Adobe PDF Reader on a Windows/x86 platform is possible". We also find the limits and difficult circumstances of this method against opponents aiming to avoid "anomaly-based detection". The proposed detector enhances safety by complementing current signature-based techniques and maybe employed alongside them.

III. METHODOLOGY

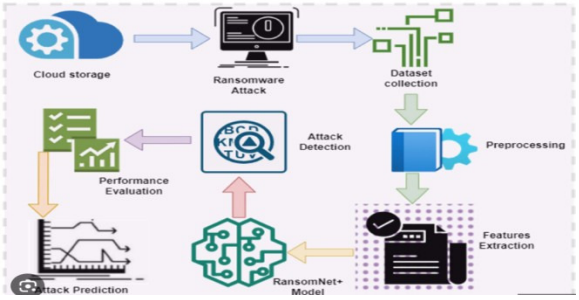
1) Proposed Work

The counseled solution gives a modern technique for "ransomware detection on virtual machines (VMs"). From the host tool, it accumulates specific CPU and disk I/O events for the digital system as entire. To construct a sturdy detection model, gadget getting to know—especially a "random wooded area (RF)"—classifier—is used. This technique aims to reduce the tracking load associated with constant surveillance of all approaches on the aim machine, therefore reducing the opportunity of records corruption via ransomware. It is likewise famous for being strong against changes in personal workload. "With the [52] RF classifier outperforming numerous tested classifiers", the suggested technique lets in speedy detection with elevated accuracy for every acknowledged and unknown ransomware. This paper presents novel changes combining "Convolutional Neural Network 2d (CNN2D)" with Model of the voting classifier to improve ransomware detection accuracy. Like many other classifiers of the device learning, the voting classifier has completed a remarkable 99% accuracy in final predictions, emphasizing the integration value of the version to detect a forward step.

2) System Architecture

This article examines a short detection of ransomware on Windows 10 "Digital Machine" at some unspecified time in the future. In the host system phase, every HPC and I/O disk data are collected. Virtual devices have low to no impact on their general overall performance and are oblivious to facts accumulating and tracking. [24 We select "gadget mastering (ML) algorithms" to analyze records and perceive ransomware in use [52]. Our method could be very strong in defensive customers of virtual computers inside a cloud surroundings. We propose a technique the use of excessive-overall performance computation and disk I/O information obtained from the host system for the most fulfilling ransomware detection. Our technique reduces facts tain with the usage of ransomware supposed to dam such tracking tries and avoids the load of supervising many moves on the aim machine.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



"Fig 1 Proposed architecture"

3) Dataset collection

Records monitoring processor and disk I/O hobby over virtual machine operation make up the HPC dataset obtained below the studies. Carefully compiled to mirror a number of device movements, this dataset offers a strong foundation for training and evaluation of ransomware detection fashions. By manner of inclusion of every mentioned ransomware pattern for version calibration and unknown samples for resilience checking out [16], [17], [18], [19], [20], The HPC data file allows practical modeling probably ransomware behavior in real world computational environments.

	instructions	LLC-stores	L1-icache-load-misses	branch-load-misses	node-load-misses	rd_req	rd_bytes	wr_req	wr_bytes	flush_operations	rd_total_times	wr_total_times	flush_total_times
0	775561160.0	9575.0	257517.0	215949.0	0.0	0	0	8	147456	4	0	3596349	4524778
1	32981037.0	16800.0	797990.0	140417.0	2.0	0	0	0	0	0	0	0	0
2	11049222.0	5302.0	204689.0	55819.0	0.0	0	0	0	0	0	0	0	0
3	4968323.0	5252.0	188982.0	34310.0	0.0	0	0	0	0	0	0	0	0
4	15314480.0	11345.0	601098.0	112428.0	0.0	0	0	0	0	0	0	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...
5995	58694891.0	1827.0	9147.0	468151.0	0.0	0	0	0	0	0	0	0	0
5996	51471771.0	863.0	3740.0	521066.0	0.0	0	0	0	0	0	0	0	0
5997	49304168.0	5280.0	12560.0	480046.0	0.0	0	0	2	28672	1	0	503895	2094439
5998	55666764.0	13175.0	53589.0	529373.0	0.0	0	0	0	0	0	0	0	0
5999	50258111.0	5596.0	58048.0	526161.0	3.0	0	0	0	0	0	0	0	0

"ig 2 Dataset"

4) Data Processing

Data processing converts unrefined facts to meaningful knowledge for organizations. Data scientists usually interact in data processing, include a collection, organization, cleaning, verification, evaluation and transformation of facts into understandable formats, including graphs or articles. records processing may be carried out thru 3 methods: "manual, mechanical, and electronic". Here the goal is to decorate the knowledge price and simplify decisions. This allows companies to make quick strategic decisions and increase operations. automated facts processing technologies, along with laptop software program programming, are pivotal in this context. it can rework enormous volumes of facts, in particular large facts, into great insights for nice control and decision-making.

5) Feature selection

Feature selection is determining for model development the maximum steady, non-redundant, and relevant capabilities. Since the quantity and variety of datasets hold to upward thrust, methodically decreasing dataset size is genuinely crucial. Characteristic preference's predominant objectives are to minimize computing cost of modeling and enhance the overall performance of a predictive model.

A basic factor of characteristic engineering, feature selection is determining the greatest superb capacities for input into system gaining knowledge of algorithms. By omitting redundant or superfluous abilities, feature selection methods help to lessen the range of input variables, so enhancing the set to the ones most relevant to the system discovering version. The foremost benefits of the usage of feature preference earlier rather than allowing the device studying model to determine the relevance of features independently.

6) Algorithms

- "Long short term memory (LSTM)":The long short -term memory (LSTM), which is designed to solve the problem with the disappearing gradient in conventional RNN, is a type of recurring neuralnetwork (RNN). It is especially useful for duties related to time series or sequential patterns since it provides a memory mobile that lets the version capture long-term dependencies in sequential data.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

Given their ability to model and identify temporal dependencies—which could be vital for ransomware detection, since the gathering of system actions and movements is quite important—LSTMs are most likely used in these research. LSTMs improve the potential of the version to stumble on ransomware pastime by spotting diffused styles across time.

LSTM

```
X_train = X_train.values
X_test = X_test.values

#now train LSTM algorithm
X_train1 = np.reshape(X_train, (X_train.shape[0], X_train.shape[1], 1))
X_test1 = np.reshape(X_test, (X_test.shape[0], X_test.shape[1], 1))

lstm_model = Sequential()#defining deep learning sequential object
#adding LSTM layer with 100 filters to filter given input X train data to select relevant features
lstm_model.add(LSTM(32,input_shape=(X_train1.shape[1], X_train1.shape[2])))
#adding dropout layer to remove irrelevant features
lstm_model.add(Dropout(0.2))
#adding another layer
lstm_model.add(Dense(32, activation='relu'))
#defining output layer for prediction
lstm_model.add(Dense(y_train1.shape[1], activation='softmax'))
#compile LSTM model
lstm_model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])
#start training model on train data and perform validation on test data
#train and load the model
#if os.path.exists("model/lstm_weights.hdf5") == False:
    #model_check_point = ModelCheckpoint(filepath='model/lstm_weights.hdf5', verbose = 1, save_best_only = True)
hist = lstm_model.fit(X_train1, y_train1, batch_size = 32, epochs = 10, validation_data=(X_test1, y_test1), verbose=1)
    #f = open('model/lstm_history.pckl', 'wb')
    #pickle.dump(hist.history, f)
    #f.close()
#else:
    # lstm_model.load_weights("model/lstm_weights.hdf5")
#perform prediction on test data
```

"Fig 3 LSTM"

- "Deep Neural Network (DNN):""A Deep Neural network is a category of artificial neural network" characterized by way of numerous hidden layers located among the input and output layers. those networks can research complicated hierarchical representations of records, rendering them suitable for difficult obligations that necessitate characteristic abstraction and representation. Deep Neural Networks may be applied in the task due to their potential to study complex capabilities and relationships inside the accumulated records. In ransomware detection, wherein problematic and nuanced styles may be present, "Deep Neural Networks (DNNs)"Provide a robust framework for characteristic extraction and purchase of high -grade representations [8], [13], [14].

DNN

```
#train DNN algorithm
y_train1 = to_categorical(y_train)
y_test1 = to_categorical(y_test)
#define DNN object
dnn_model = Sequential()
#add DNN layers
dnn_model.add(Dense(2, input_shape=(X_train.shape[1],), activation='relu'))
dnn_model.add(Dense(2, activation='relu'))
dnn_model.add(Dropout(0.3))
dnn_model.add(Dense(y_train1.shape[1], activation='softmax'))
# compile the keras model
dnn_model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])
#start training model on train data and perform validation on test data
#train and load the model
#if os.path.exists("model/dnn_weights.hdf5") == False:
    # model_check_point = ModelCheckpoint(filepath='model/dnn_weights.hdf5', verbose = 1, save_best_only = True)
hist = dnn_model.fit(X_train, y_train1, batch_size = 32, epochs = 10, validation_data=(X_test, y_test1),  verbose=1)
    # f = open('model/dnn_history.pckl', 'wb')
    # pickle.dump(hist.history, f)
  #  f.close()
#else:
    # dnn_model.load_weights("model/dnn_weights.hdf5")
#perform prediction on test data
```

"Fig 4 DNN"

- "XGBoost":"XGBoost, or extreme Gradient Boosting, is a machine learning algorithm categorized inside the gradient boosting strategies". This produces a weak student ensemble that often follows a sequential pattern; each tree fixes the errors of the previous one, hence producing a strong and elegant model. XGBOOST handles big datasets with efficiency and for classification purposes. XGBoost demonstrates robust predictive talents in ransomware detection by using successfully capturing the many elements of ransomware behaviors, hence facilitating the improvement of an correct detection version [8], [13], [14].

XGBoost

```
#now train XGBoost algorithm
xgb_cls = XGBClassifier(n_estimators=10,learning_rate=0.09,max_depth=2)
xgb_cls.fit(X_train, y_train)
predict = xgb_cls.predict(X_test)
calculateMetrics("XGBoost", predict, y_test)

rf_acc = accuracy_score(predict, y_test)
rf_prec = precision_score(predict, y_test,average='macro')
rf_rec = recall_score(predict, y_test,average='macro')
rf_f1 = f1_score(predict, y_test,average='macro')

storeResults('XGBoost',rf_acc,rf_prec,rf_rec,rf_f1)
```

"Fig 5 Xgboost"

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

- "Random Forest":"Random forest is an ensemble learning technique" that generates numerous decision timber at some stage in the education method. It produces the mode of the classes for classification or the suggest prediction for regression from person bushes. application in the undertaking: Random forest is applied for its capacity to manipulate tricky category challenges. In ransomware detection, where numerous styles may be gift, a By means of combining the features of several selection bushes, random forest area ensemble can improve accuracy and produce a strong and consistent model.

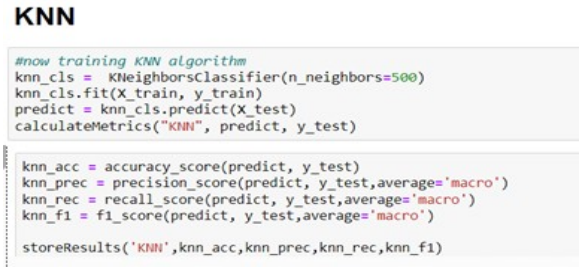
"Fig 6 Random forest"

- "Decision Tree:"A decision tree is a hierarchical architecture whereby every node stands for a choice depending on input capacity. Recursively breaking the dataset into subgroups, it creates main to terminal nodes that offer the very last prediction or category. Decision bushes are appreciated for their simplicity in showing methods of selecting and for their readability. In ransomware detection, they clarify the sequential actions guiding to a choice, so enhancing the knowledge of the main elements affecting the detection result [52].

"Fig 7 Decision tree"

- "K – Nearest Neighbor (KNN):"

Designed for both categorization and regression tasks, "K-Nearest Neighbors (KNN)" is a "supervised machine learning technique". It forecasts its fee using averaging the values of its k nearest friends inside the function space and classifies a brand-new statistics component mostly based on majority balloting from its k nearest friends. KNN's simplicity and efficiency in identifying adjacent patterns in data appeal. KNN provides a flexible way to identify similar patterns in the dataset in ransomware detection, where minor versions could also exist.



"Fig 8 KNN"

- "Support Vector Machine (SVM)":

"Support Vector machine (SVM) is a supervised machine learning technique" hired for classification and regression tasks. It identifies a hyperplane that optimally distinguishes data into distinct lessons or forecasts a non-stop final results, consequently optimizing the margin between training.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

"support Vector machine (SVM)" is applied for its talent in coping with excessive-dimensional data and identifying finest choice boundaries. In ransomware detection, characterized through complex feature areas, SVM offers a sturdy technique for correct classification through delineating distinct decision bounds [52].

"Fig 9 SVM"

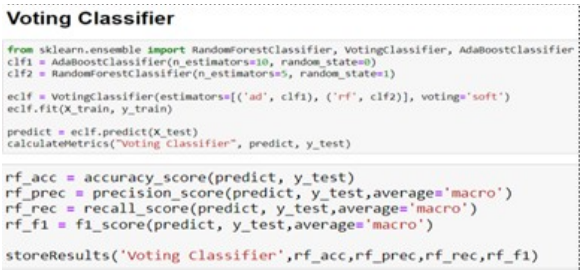
- "2D Convolutional Neural Network (CNN2D):"

Usually used in photo processing, CNN2D is a deep learning tool meant for grid-based statistics analysis. CNN2D is designed to process continuous entry and uses convolutional layers to routinely extract hierarchical features and patterns in this have a view. It is used because it can independently grab complex capabilities from continuous data. CNN2D can detect complex functions in ransomware, where there are vital formulas in sequential and continuous systemic activities, thereby increasing the efficiency of the detection model.

"Fig 10 CNN2D"

- "Voting Classifier:"

Combining predictions from several models, a voting classifier uses average or majority voting. It uses several techniques to increase normal model general performance. This ensemble approach presents a more thorough and trustworthy assessment by combining forecasts from several models, therefore complementing the accuracy and resilience of the ransomware detection system.



"Fig 11 Voting classifier"

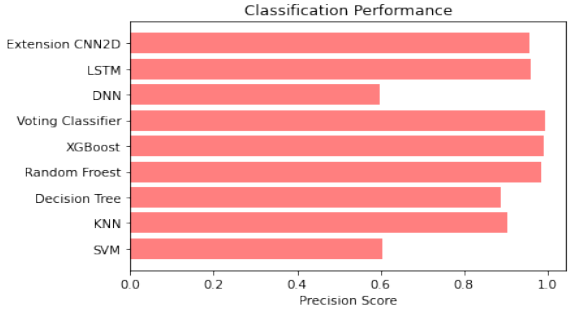
IV. EXPERIMENTAL RESULTS

1) *Precision:* Precision assesses the share of appropriately classified cases among those identified as fine. therefore, the formulation for calculating precision is expressed as:

"Precision = True positives/ (True positives + False positives) = TP/(TP + FP)"

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

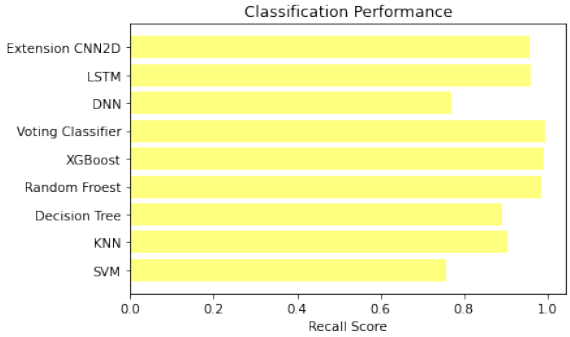
$$\text{Precision} = \frac{True\ Positive}{True\ Positive + False\ Positive}$$



"Fig 12 Precision comparison graph"

2) *Recall:* Recall is a metric in machine learning that measures the ability of the model to identify all relevant instances of a particular class. It is the ratio of correctly predicted positive observations to overall real positives and provides information on the completeness of the model when capturing the instances of the class.

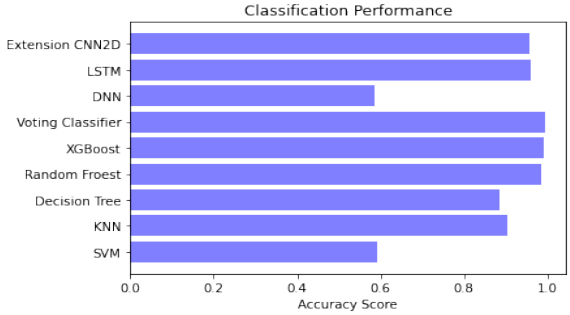
$$Recall = \frac{TP}{TP + FN}$$



"Fig 13 Recall comparison graph"

3) *Accuracy:* Accuracy is the ratio of accurate predictions in a class take a look at, assessing the overall precision of a model's predictions.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

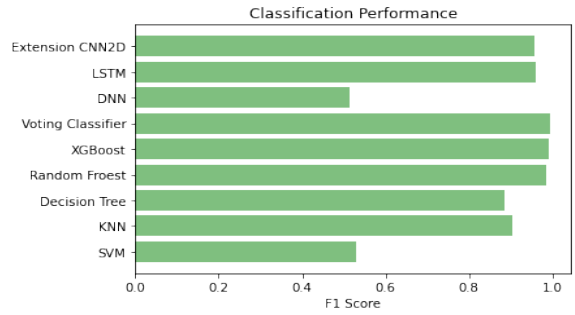


"Fig 14 Accuracy graph"

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

4) *F1 Score:* F1 - Score is a metric of evaluation of machine learning that measures the accuracy of the model. It combines the score and induction of the model. The accuracy metric calculates how many times the model has created the correct prediction throughout the data file.

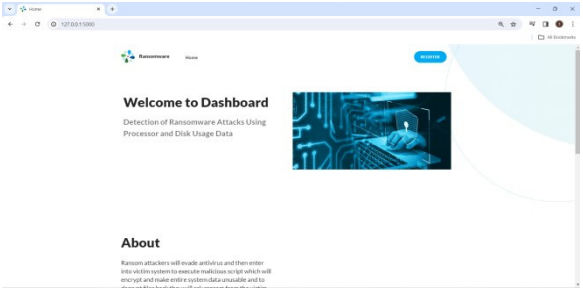
$$\text{F1 Score} = 2 * \frac{\text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} * 100$$



"Fig 15 F1Score"

ML Model	Accuracy	Precision	Recall	F1-score
SVM	0.593	0.605	0.758	0.529
KNN	0.904	0.905	0.905	0.904
Decision Tree	0.885	0.887	0.891	0.885
Random Forest	0.985	0.985	0.985	0.985
XGBoost	0.992	0.991	0.992	0.992
Extension Voting Classifier	0.995	0.995	0.995	0.995
DNN	0.585	0.597	0.769	0.513
LSTM	0.960	0.961	0.961	0.960
Extension CNN2D	0.956	0.957	0.957	0.956

"Fig 16 Performance Evaluation"

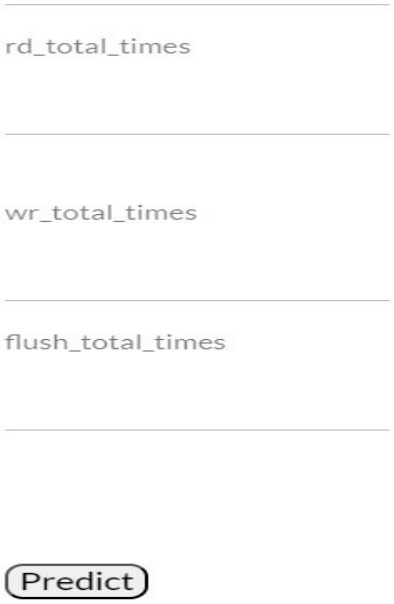


"Fig 17 Home page"

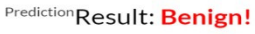
"Fig 18 Signin page"

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

"Fig 19 Login page"



"Fig 20User input"



"Fig 21 Predict result for given input"

V. CONCLUSION

The undertaking effectively implements a new approach for detection of ransomware, the use of virtualization era, powerful performance counter and IO information to improve accuracy while decreasing the impact on machine performance. Research conducts complete experiments to evaluate numerous gadget studying algorithms, together with "SVM, KNN, decision -making tree, random forest, XGBOOSTCH, DNN and LSTM, suggesting that random forest area and XGBOOST" constantly expect splendid accuracy in predicting ransomware sports. It examines the effectiveness of deep look at models, specifically DNN and LSTM, which provides a giant insight into their average performance in comparison to the standard machine of algorithms, improving the kind of prediction methodologies. The effort complements the cybersecurity community with the aid of freeing a

publicly to be had dataset sourced from many packages, promoting collaboration and permitting lecturers to evaluate their ransomware detection techniques.

The task efficaciously incorporates Flask as the internet framework and SQLite for person registration and authentication, supplying an intuitive interface that permits users to input data, undergo preprocessing, and acquire predictions from the educated version, hence improving practical use.

VI. FUTURE SCOPE

The following examine should acknowledge the assessment of the effectiveness of the advocated approach in the detection of recent and emerging versions of ransomware, due to the fact the modern-day view of the mixture of identified and unknown ransomware. Extension of research to evaluate the impact of a couple of workload of consumers on ransomware detection [7] would carry enormous understanding, which could improve the confirmed flexibility of a unique workload. The voting classifier, the extension of the technology, showed a great performance with an accuracy of 99% for ransomware detection. A thorough checking out at the the front stop using the function values confirms its electricity and efficiency in regular detection and remedy of ransomware threats. The overall performance and evaluation of warned era inside the contexts of the real international should provide an empirical insight into its efficiency in detecting attacks on ransomware in stay production structures. The venture may want to improve its ransomware detection competencies through exploring the incorporation of supplementary data resources or attributes into the device gaining knowledge of model. Engagement with cybersecurity experts and businesses gives a chance to corroborate findings and beautify the proposed technique utilizing realistic abilities and insights.

REFERENCES

[1] SR Department. (2022). Ransomware victimization rate 2022. Accessed: Apr. 6, 2022. [Online]. Available: https://www.statista.com/statistics/204457/businesses-ransomware-attack-rate/
[2] D. Braue. (2022). Ransomware Damage Costs. Accessed: Sep. 16, 2022. [Online]. Available: https://cybersecurityventures.com/globalransomware-damage-costs-predicted-to-reach-250-billion-usd-by-2031/
[3] Logix Consulting. (2020). What is Signature Based Malware Detection. Accessed: Apr. 3, 2023. [Online]. Available: https://www.logixconsulting.com/2020/12/15/what-is-signature-based-malware-detection/
[4] W. Liu, P. Ren, K. Liu, and H.-X. Duan, ''Behavior-based malware analysis and detection,'' in Proc. 1st Int. Workshop Complex. Data Mining, Sep. 2011, pp. 39–42.
[5] (2021). Polymorphic Malware. Accessed: Apr. 3, 2023. [Online]. Available: https://www.thesslstore.com/blog/polymorphic-malware-andmetamorphic-malware-what-you-need-to-know/
[6] M. Loman. (2021). Lockfile Ransomware's Box of Tricks: Intermittent Encryption and Evasion. Accessed: Nov. 16, 2021. [Online]. Available: https://news.sophos.com/en-us/2021/08/27/lockfile-ransomwares-box-oftricks-intermittent-encryption-and-evasion/
[7] N. Pundir, M. Tehranipoor, and F. Rahman, ''RanStop: A hardwareassisted runtime crypto-ransomware detection technique,'' 2020, arXiv:2011.12248.
[8] S. Mehnaz, A. Mudgerikar, and E. Bertino, ''RWGuard: A real-time detection system against cryptographic ransomware,'' in Proc. Int. Symp. Res. Attacks, Intrusions, Defenses. Cham, Switzerland: Springer, 2018, pp. 114–136.
[9] J. Demme, M. Maycock, J. Schmitz, A. Tang, A. Waksman, S. Sethumadhavan, and S. Stolfo, ''On the feasibility of online malware detection with performance counters,'' ACM SIGARCH Comput. Archit. News, vol. 41, no. 3, pp. 559–570, Jun. 2013.
[10] A. Tang, S. Sethumadhavan, and S. J. Stolfo, ''Unsupervised anomalybased malware detection using hardware features,'' in Proc. Int. Workshop Recent Adv. Intrusion Detection. Cham, Switzerland: Springer, 2014, pp. 109–129.
[11] S. Das, J. Werner, M. Antonakakis, M. Polychronakis, and F. Monrose, ''SoK: The challenges, pitfalls, and perils of using hardware performance counters for security,'' in Proc. IEEE Symp. Secur. Privacy (SP), May 2019, pp. 20–38.
[12] S. P. Kadiyala, P. Jadhav, S.-K. Lam, and T. Srikanthan, ''Hardware performance counter-based fine-grained malware detection,'' ACM Trans. Embedded Comput. Syst., vol. 19, no. 5, pp. 1–17, Sep. 2020.
[13] B. Zhou, A. Gupta, R. Jahanshahi, M. Egele, and A. Joshi, ''Hardware performance counters can detect malware: Myth or fact?'' in Proc. Asia Conf. Comput. Commun. Secur., May 2018, pp. 457–468.
[14] [S. Aurangzeb, R. N. B. Rais, M. Aleem, M. A. Islam, and M. A. Iqbal, ''On the classification of microsoft-windows ransomware using hardware profile,'' PeerJ Comput. Sci., vol. 7, p. e361, Feb. 2021.
[15] M. Alam, S. Bhattacharya, S. Dutta, S. Sinha, D. Mukhopadhyay, and A. Chattopadhyay, ''RATAFIA: Ransomware analysis using time and frequency informed autoencoders,'' in Proc. IEEE Int. Symp. Hardw. Oriented Secur. Trust (HOST), May 2019, pp. 218–227.
[16] K. Thummapudi, R. Boppana, and P. Lama, ''HPC 41 events 5 rounds,'' Harvard Dataverse, 2022, doi: 10.7910/DVN/MA5UPP.
[17] K. Thummapudi, R. Boppana, and P. Lama, ''IO 41 events 5 rounds,'' Harvard Dataverse, 2022, doi: 10.7910/DVN/GHJFUT.
[18] K. Thummapudi, R. Boppana, and P. Lama, ''HPC 5 events 7 rounds,'' Harvard Dataverse, 2022, doi: 10.7910/DVN/YAYW0J.
[19] K. Thummapudi, R. Boppana, and P. Lama, ''Io 5 events 7 rounds,'' Harvard Dataverse, 2022, doi: 10.7910/DVN/R9FYPL.
[20] K. Thummapudi, R. Boppana, and P. Lama, ''Scripts to reproduce results,'' Harvard Dataverse, 2023, doi: 10.7910/DVN/HSX6CS.
[21] M. Rhode, P. Burnap, and A. Wedgbury, ''Real-time malware process detection and automated process killing,'' Secur. Commun. Netw., vol. 2021, pp. 1–23, Dec. 2021.

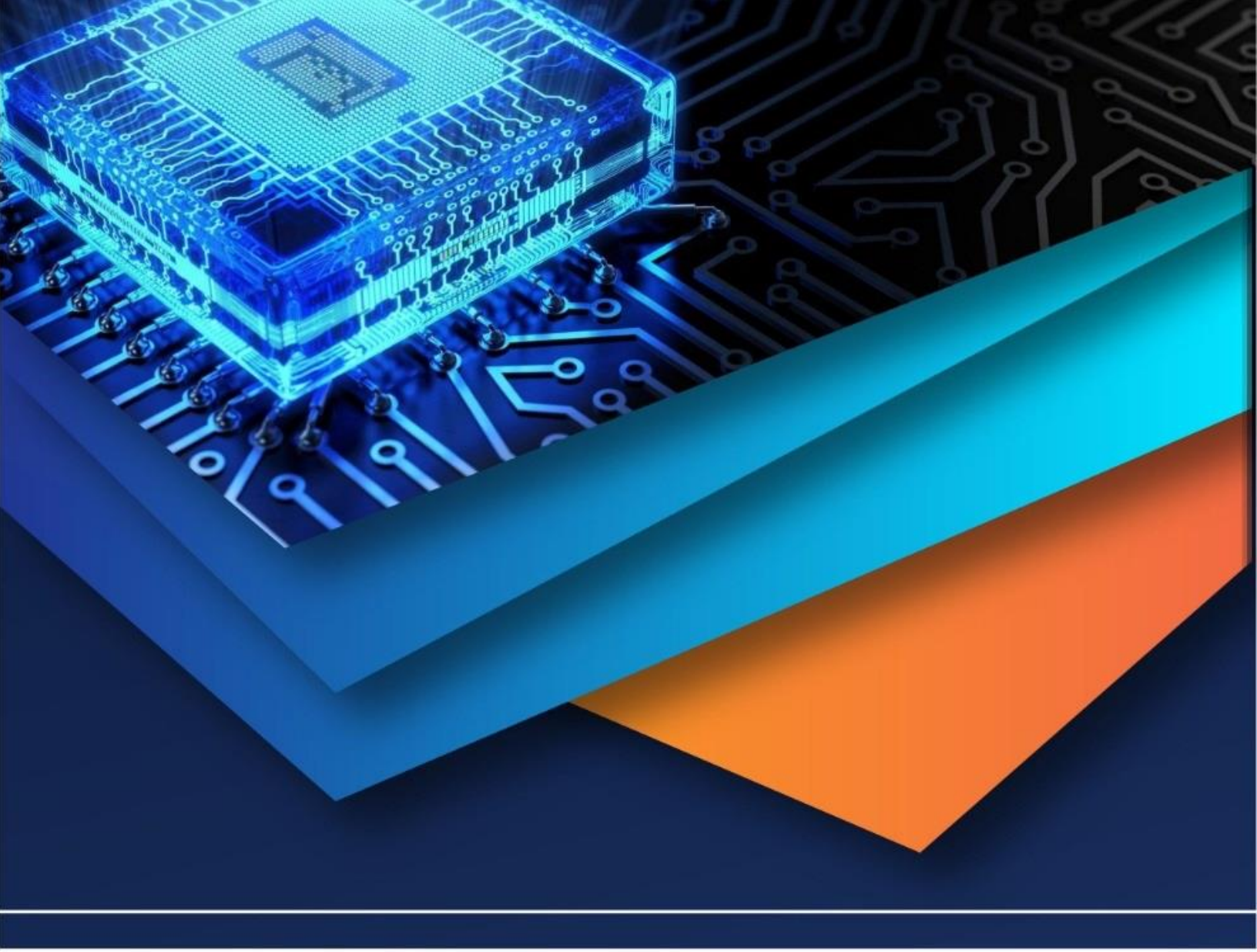
©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

[22] A. Kharraz and E. Kirda, ''Redemption: Real-time protection against ransomware at end-hosts,'' in Proc. Int. Symp. Res. Attacks, Intrusions, Defenses. Cham, Switzerland: Springer, 2017, pp. 98–119.
[23] F. Mbol, J.-M. Robert, and A. Sadighian, ''An efficient approach to detect torrentlocker ransomware in computer systems,'' in Proc. Int. Conf. Cryptol. Netw. Secur. Springer, 2016, pp. 532–541.
[24] K. Lee, S. Lee, and K. Yim, ''Machine learning based file entropy analysis for ransomware detection in backup systems,'' IEEE Access, vol. 7, pp. 110205–110215, 2019.
[25] C. J. Chew and V. Kumar, ''Behaviour based ransomware detection,'' in Proc. Int. Conf. Comput. Their Appl., in EPiC Series in Computing, vol. 58. 2019, pp. 127–136.
[26] S. Homayoun, A. Dehghantanha, M. Ahmadzadeh, S. Hashemi, and R. Khayami, ''Know abnormal, find evil: Frequent pattern mining for ransomware threat hunting and intelligence,'' IEEE Trans. Emerg. Topics Comput., vol. 8, no. 2, pp. 341–351, Apr. 2020.
[27] A. Kharaz, S. Arshad, C. Mulliner, W. Robertson, and E. Kirda, ''UNVEIL: A large-scale, automated approach to detecting ransomware (keynote),'' in Proc. IEEE 24th Int. Conf. Softw. Anal., Evol. Reengineering (SANER), Feb. 2017, pp. 757–772.
[28] A. Kharraz, W. Robertson, D. Balzarotti, L. Bilge, and E. Kirda, ''Cutting the gordian knot: A look under the hood of ransomware attacks,'' in Proc. Int. Conf. Detection Intrusions Malware, Vulnerability Assessment. Cham, Switzerland: Springer, 2015, pp. 3–24.
[29] A. Continella, A. Guagnelli, G. Zingaro, G. De Pasquale, A. Barenghi, S. Zanero, and F. Maggi, ''ShieldFS: A self-healing, ransomware-aware filesystem,'' in Proc. 32nd Annu. Conf. Comput. Secur. Appl., Dec. 2016, pp. 336–347.
[30] M. Shukla, S. Mondal, and S. Lodha, ''POSTER: Locally virtualized environment for mitigating ransomware threat,'' in Proc. ACM SIGSAC Conf. Comput. Commun. Secur., Oct. 2016, pp. 1784–1786.
[31] N. Scaife, H. Carter, P. Traynor, and K. R. B. Butler, ''CryptoLock (and drop it): Stopping ransomware attacks on user data,'' in Proc. IEEE 36th Int. Conf. Distrib. Comput. Syst. (ICDCS), Jun. 2016, pp. 303–312.
[32] D. Sgandurra, L. Muñoz-González, R. Mohsen, and E. C. Lupu, ''Automated dynamic analysis of ransomware: Benefits, limitations and use for detection,'' 2016, arXiv:1609.03020.
[33] P. Zavarsky and D. Lindskog, ''Experimental analysis of ransomware on windows and Android platforms: Evolution and characterization,'' Proc. Comput. Sci., vol. 94, pp. 465–472, Jan. 2016.
[34] T. McIntosh, J. Jang-Jaccard, P. Watters, and T. Susnjak, ''The inadequacy of entropy-based ransomware detection,'' in Proc. Int. Conf. Neural Inf. Process. Cham, Switzerland: Springer, 2019, pp. 181–189.
[35] Z. A. Genc, G. Lenzini, and D. Sgandurra, ''On deception-based protection against cryptographic ransomware,'' in Proc. Int. Conf. Detection Intrusions Malware, Vulnerability Assessment Cham, Switzerland: Springer, 2019, pp. 219–239.
[36] S. Song, B. Kim, and S. Lee, ''The effective ransomware prevention technique using process monitoring on Android platform,'' Mobile Inf. Syst., vol. 2016, pp. 1–9, Mar. 2016.
[37] K. Cabaj, M. Gregorczyk, and W. Mazurczyk, ''Software-defined networking-based crypto ransomware detection using HTTP traffic characteristics,'' Comput. Electr. Eng., vol. 66, pp. 353–368, Feb. 2018.
[38] R. Moussaileb, B. Bouget, A. Palisse, H. Le Bouder, N. Cuppens, and J.-L. Lanet, ''Ransomware's early mitigation mechanisms,'' in Proc. 13th Int. Conf. Availability, Rel. Secur., 2018, pp. 1–10.
[39] Z. A. Genc, G. Lenzini, and P. Y. Ryan, ''No random, no ransom: A key to stop cryptographic ransomware,'' in Proc. Int. Conf. Detection Intrusions Malware, Vulnerability Assessment. Cham, Switzerland: Springer, 2018, pp. 234–255.
[40] M. M. Ahmadian, H. R. Shahriari, and S. M. Ghaffarian, ''Connectionmonitor & connection-breaker: A novel approach for prevention and detection of high survivable ransomwares,'' in Proc. 12th Int. Iranian Soc. Cryptol. Conf. Inf. Secur. Cryptol. (ISCISC), Sep. 2015, pp. 79–84.
[41] E. Kolodenker, W. Koch, G. Stringhini, and M. Egele, ''PayBreak: Defense against cryptographic ransomware,'' in Proc. ACM Asia Conf. Comput. Commun. Secur., Apr. 2017, pp. 599–611.
[42] M. S. Kiraz, Z. A. Genc, and E. Ozturk, ''Detecting large integer arithmetic for defense against crypto ransomware,'' Cryptology ePrint Arch., Rep., vol. 558, p. 2017, Jan. 2017.
[43] (2022). What Systems Have You Seen Infected by Ransomware? Accessed: Apr. 3, 2023. [Online]. Available: https://www.statista.com/ statistics/701020/major-operating-systems-targeted-by-ransomware/
[44] (2022). Linux Profiling With Performance Counters. Accessed: Apr. 3, 2023. [Online]. Available: https://perf.wiki.kernel.org/ index.php/MainPage
[45] (2022). Likwid Performance Tools. Accessed: Apr. 3, 2023. [Online]. Available: https://hpc.fau.de/research/tools/likwid/
[46] K. Keahey, J. Anderson, Z. Zhen, P. Riteau, P. Ruth, D. Stanzione, M. Cevik, J. Colleran, H. S. Gunawi, C. Hammock, J. Mambretti, A. Barnes, F. Halbach, A. Rocha, and J. Stubbs, ''Lessons learned from the chameleon testbed,'' in Proc. USENIX Annu. Tech. Conf., Jul. 2020, pp. 219–233.
[47] Alexa Top Websites. Accessed: Sep. 13, 2021. [Online]. Available: https://www.alexa.com/topsites
[48] Ninite. Accessed: Apr. 3, 2023. [Online]. Available: https://ninite.com
[49] API. (2023). Libvirt: Virsh Tool Manual. Accessed: Apr. 3, 2023. [Online]. Available: https://libvirt.org/manpages/virsh.html
[50] (2021). Virusshare. Accessed: Nov. 16, 2021. [Online]. Available: https://virusshare.com
[51] Intel. (2023). System Programming Guide. Accessed: Apr. 3, 2023. [Online]. Available: https://www.intel.com/content/dam/www/ public/us/en/documents/manuals/64-ia-32-architectures-softwaredeveloper-vol-3b-part-2-manual.pdf
[52] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, ''Scikit-learn: Machine learning in Python,'' J. Mach. Learn. Res., vol. 12, pp. 2825–2830, Oct. 2011.
[53] J. Benesty, J. Chen, Y. Huang, and I. Cohen, ''Pearson correlation coefficient,'' in Noise Reduction in Speech Processing. Berlin, Germany: Springer, 2009, pp. 1–4.
[54] H. Jin, Q. Song, and X. Hu, ''Auto-Keras: An efficient neural architecture search system,'' in Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, Jul. 2019, pp. 1946–1956.
[55] Wikipedia. (2021). Sensitivity and Specificity. Accessed: Apr. 3, 2023. [Online]. Available: https://en.wikipedia.org/wiki/ Sensitivityandspecificity

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

[56] Hat Enterprise. (2023). Block I/O Tuning. [Online]. Available: https://access.redhat.com/documentation/en-us/redhatenterpriselinux/7/html/virtualizationtuningandoptimizationguide/sectvirtualizationtuningoptimizationguide-blockio-techniques

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)