MACHINE LEARNING TECHNIQUES FOR CYBER ATTACK DETECTION

[1]ONTELA KAVYASRI, [2] S.VIJAY KUMAR
[1]MCA Student, [2] Assistant Professor
Department Of MCA
Sree Chaitanya College of Engineering, Karimnagar

ABSTRACT

Contrasted with the past, improvements in PC and correspondence innovations have given broad and propelled changes. The use of new innovations give incredible advantages to people, organizations, and governments, be that as it may, messes some up against them. For instance, the protection of significant data, security of put away information stages, accessibility of information and so forth. Contingent upon these issues, digital fear based oppression is one of the most significant issues in this day and age. Digital fear, which made a great deal of issues people and establishments, has arrived at a level that could undermine open and nation security by different gatherings, for example, criminal association, proficient people and digital activists. Along these lines, Intrusion Detection Systems (IDS) has been created to maintain a strategic distance from digital assaults. Right now, learning the bolster support vector machine (SVM) calculations were utilized to recognize port sweep endeavors dependent on the new CICIDS2017 dataset with 97.80%, 69.79% precision rates were accomplished individually. Rather than SVM we can introduce some other algorithms like random forest, CNN, ANN where these algorithms can acquire accuracies like SVM – 93.29, CNN – 63.52, Random Forest – 99.93, ANN – 99.11.

1. INTRODUCTION

Political and economic actors are increasingly using sophisticated cyber-warfare to disrupt, destroy, or suppress information content in computer networks. There is a requirement to assure network protocol resilience against incursions by powerful attackers who can even control a percentage of the network's parties. Both passive (eavesdropping, nonparticipation) and active (jamming, message dropping, corruption, and forging) assaults can be launched by the controlled parties. Intrusion detection is the system which continuously monitoring events in a computer system or network, analysing them for signals of potential problems, and, in many cases, preventing unwanted access. This is usually performed by automatically gathering data from a range of systems and network for potential security issues. Traditional intrusion detection and solutions, such as firewalls, access controlling mechanisms, and encryptions, have significant flaws when it comes to properly defending networks and systems against more complex assaults such as denial of service. Furthermore, most systems based on such

methodologies have a high rate of false positive and false negative detection, as well as a lack of ability to react to changing harmful behaviour. Several Machine Learning (ML) approaches have, however, been applied to the challenge of intrusion detection in the last decade in the hopes of boosting detection rates and adaptability. These methods are frequently employed to maintain attack information bases current and thorough. Cyber-security and defence against a variety of cyber-attacks has recently become a hot topic. The fundamental reason for this is the phenomenal expansion of computer technology. a large number of relevant apps used by people or groups for personal or commercial purposes, particularly after the Internet of Things was accepted (IoT). The cyber-threats wreak havoc and generate significant financial losses on a huge scale networks. Hardware and software solutions that are already in place Firewalls, user authentication, and data encryption mechanisms are all examples of security measures. Not enough to address the anticipated demand problem, and Unfortunately, the computer network's multiple computers were unable to be protected. Cyber-threats. These traditional security arrangements aren't working. Sufficient as a protection as a result of the more rapid and rigorous evolution of intrusion detection systems Only the access from the firewall is controlled. The term "network to network" refers to the inability of two networks to communicate with each other. Networks. However, it does not send out any alerts in the event of an emergency. As a result, it is self-evident that accurate defence must be developed. Intrusion detection approaches based on machine learning system (IDS) for the security of the system In general, an encroachment A detection system (IDS) is a programme or system that detects something.

Infectious activities and policy breaches in a network or system system. An IDS detects anomalies and inconsistencies. During the course of daily activities, behaviour on a network is observed. In a network or system that detects security threats or assaults.

2. LITERATURE SURVEY

R. Christopher, "Port scanning techniques and the defense against them," SANS Institute, 2001.

Port Scanning is one of the most popular techniques attackers use to discover services that they can exploit to break into systems. All systems that are connected to a LAN or the Internet via a modem run services that listen to well-known and not so well-known ports. By port scanning, the attacker can find the following information about the targeted systems: what services are running, what users own those services, whether anonymous logins are supported, and whether certain network services require authentication. Port scanning is accomplished by sending a message to each port, one at a time. The kind of response received indicates whether the port is used and can be probed for further weaknesses. Port scanners are important to network security technicians because they can reveal possible security vulnerabilities on the targeted system. Just as port scans can be ran against your systems, port scans can be

detected and the amount of information about open services can be limited utilizing the proper tools. Every publicly available system has ports that are open and available for use. The object is to limit the exposure of open ports to authorized users and to deny access to the closed ports.

S. Staniford, J. A. Hoagland, and J. M. McAlerney, "Practical automated detection of stealthy portscans," Journal of Computer Security, vol. 10, no. 1-2, pp. 105–136, 2002.

Portscanning is a common activity of considerable importance. It is often used by computer attackers to characterize hosts or networks which they are considering hostile activity against. Thus it is useful for system administrators and other network defenders to detect portscans as possible preliminaries to a more serious attack. It is also widely used by network defenders to understand and find vulnerabilities in their own networks. Thus it is of considerable interest to attackers to determine whether or not the defenders of a network are portscanning it regularly. However, defenders will not usually wish to hide their portscanning, while attackers will. For definiteness, in the remainder of this paper, we will speak of the attackers scanning the network, and the defenders trying to detect the scan. There are several legal/ethical debates about portscanning which break out regularly on Internet mailing lists and newsgroups. One concerns whether portscanning of remote networks without permission from the owners is itself a legal and ethical activity. This is presently a grey area in most jurisdictions. However, our experience from following up on unsolicited remote portscans we detect in practice is that almost all of them turn out to have come from compromised hosts and thus are very likely to be hostile. So we think it reasonable to consider a portscan as at least potentially hostile, and to report it to the administrators of the remote network from whence it came. However, this paper is focussed on the technical questions of how to detect portscans, which are independent of what significance one imbues them with, or how one chooses to respond to them. Also, we are focussed here on the problem of detecting a portscan via a network intrusion detection system (NIDS). We try to take into account some of the more obvious ways an attacker could use to avoid detection, but to remain with an approach that is practical to employ on busy networks.

3. EXISTING SYSTEM

Blameless Bayes and Principal Component Analysis (PCA) were been used with the KDD99 dataset by Almansob and Lomte [9].Similarly, PCA, SVM, and KDD99 were used Chithik and Rabbani for IDS [10]. In Aljawarneh et al's. Paper, their assessment and examinations were conveyed reliant on the NSL-KDD dataset for their IDS model [11] Composing inspects show that KDD99 dataset is continually used for IDS [6]–[10].There are 41 highlights in KDD99 and it was created in 1999. Consequently, KDD99 is old and doesn't give any data about cutting edge new assault types, example, multi day misuses and so forth. In this manner we utilized a cutting-edge and new CICIDS2017 dataset [12] in our investigation.

DISADVANTAGES

1) Strict Regulations

2) Difficult to work with for non-technical users

3) Restrictive to resources

4) Constantly needs Patching

5) Constantly being attacked

4. Proposed System

important steps of the algorithm are given in below. 1) Normalization of every dataset. 2) Convert that dataset into the testing and training. 3) Form IDS models with the help of using RF, ANN,

Advantages

• Protection from malicious attacks on your network.

• Deletion and/or guaranteeing malicious elements within a preexisting network.

• Prevents users from unauthorized access to the network.

• Deny's programs from certain resources that could be infected.

• Securing confidential information

5. **IMPLEMENTATION**

1. Dataset Description

Because machine learning methods are used in applications for diverse network security tasks, extensive raw data are required to monitor network activities and discriminate between usual and suspicious traffic. Several efforts to generating network datasets have been carried out throughout the years. Most machine learning experiments have evaluated their work simulated or actual network data, just a few of the datasets that have been publically available, despite the fact that many of them are still private owing to security concerns.

Despite the fact that various datasets have been created, actual IoT and network traffic statistics that cover novel Botnet situations are rare.

2. Feature Extraction

Flow based features were extracted from network traffic data using the CICFlowMeter [25]. It is a CIC-enabled network traffic flow generator that generates several network traffic features. It study the pcap file and develops a document with extracted features, as well as a csv file with the database. By extracting additional features of the data set, this method was primarily intended to increase the guessing skills of class dividers.

3. Data Preprocessing

Database is converted into a machine suitable for ML using pre-processing, data processing methods. This process also includes data cleaning, which involves removing external or suspicious data that may compromise the accuracy and efficiency of database.

4. Feature Selection

It is critical to reduce the number of features used in training and testing algorithms in order to build a lightweight security solution suitable for IoT systems [13]. As a feature selection strategy, we employed the Random Forest algorithm. It showed to be a useful

tool for lowering dataset dimensionality. The model trains and responds faster when the input data features are reduced from over 80 network traffic features to seven. For the whole dataset, the relevance weights of the characteristics

ALGORITHMS

1. Random Forest

Random Forest is a classifier that contains a no.of decision trees for the various data sets provided and takes measure to improve the prediction accuracy of that data.

Working of Random Forest Algorithm:

Step 1: First, start with selection of samples from given dataset.

Step 2: After that, this will create a decision tree for each sample. The result of the forecast on each decision tree will then be available.

Step 3: Each projected outcome will voted on this round.

Step 4: Lastly, select the predictable result with the most votes as the final prediction result.

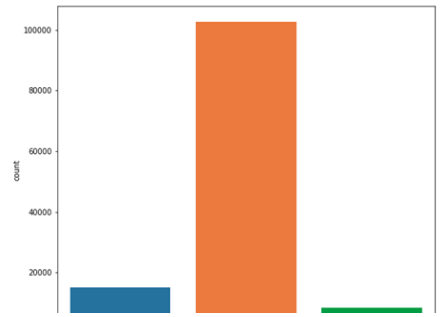
2. ANN

MLP stands for multilayer perceptron and type of feedforward artificial neural network. Artificial neural networks are a type of machine learning method that is inspired by how the human brain learns and derives new information. An MLP has three layers. For training, MLP employs the unsupervised learning approach of backpropagation.

6. SCREEN SHOTS

Data EDA

```
# Protocol type distribution
plt.figure(figsize=(9,8))
sns.countplot(x="protocol_type", data=train)
plt.show()
```



Model Building

```
train_X=train_new[cols]
train_y=train_new['attack_class']
test_X=test_new[cols]
test_y=test_new['attack_class']
```

ML Deploy

Logistic Regression

```
# Building Models
from sklearn.linear_model import LogisticRegression
logreg = LogisticRegression(random_state=0,solver='lbfgs',multi_class='multinomial')
logreg.fit( train_X, train_y)
logreg.predict(train_X)   #by default, it use cut-off as 0.5
```

```
list( zip( cols, logreg.coef_[0] ) )
```

```
logreg.intercept_
```

```
logreg.score(train_X,train_y)
```

Decision Trees

```
train_X.shape
```

```
param_grid = {'max_depth': np.arange(2, 12),
              'max_features': np.arange(10,15)}
```

```
train_y.shape
```

```
from sklearn.model_selection import GridSearchCV
from sklearn.tree import DecisionTreeClassifier, export_graphviz, export
tree = GridSearchCV(DecisionTreeClassifier(), param_grid, cv = 10,verbose=1,n_jobs=-1)
tree.fit( train_X, train_y )
```

```
tree.best_score_
```

```
tree.best_estimator_
tree.best_params_
```

```
train_pred = tree.predict(train_X)
```

```
print(metrics.classification_report(train_y, train_pred))
```

```
test_pred = tree.predict(test_X)
```

Random Forest

```
from sklearn.ensemble import RandomForestClassifier
pargrid_rf = {'n_estimators': [50,60,70,80,90,100],
              'max_features': [2,3,4,5,6,7]}
```

```
from sklearn.model_selection import GridSearchCV
gscv_rf = GridSearchCV(estimator=RandomForestClassifier(),
                       param_grid=pargrid_rf,
                       cv=10,
                       verbose=True, n_jobs=-1)

gscv_results = gscv_rf.fit(train_X, train_y)
```

```
gscv_results.best_params_
```

```
gscv_rf.best_score_
```

```
radm_clf = RandomForestClassifier(oob_score=True,n_estimators=80, max_features=5, n_jobs=-1)
radm_clf.fit( train_X, train_y )
```

```
radm_test_pred = pd.DataFrame( { 'actual':  test_y,
                                 'predicted': radm_clf.predict( test_X ) } )
```

Support Vector Machine (SVM)

```
from sklearn.svm import LinearSVC
svm_clf = LinearSVC(random_state=0, tol=1e-5)
svm_clf.fit(train_X,train_y)
```

```
print(svm_clf.coef_)
print(svm_clf.intercept_)
print(svm_clf.predict(train_X))
```

```
from sklearn.svm import SVC
from sklearn.pipeline import make_pipeline

model = SVC(kernel='rbf', class_weight='balanced',gamma='scale')
```

```
model.fit(train_X,train_y)
```

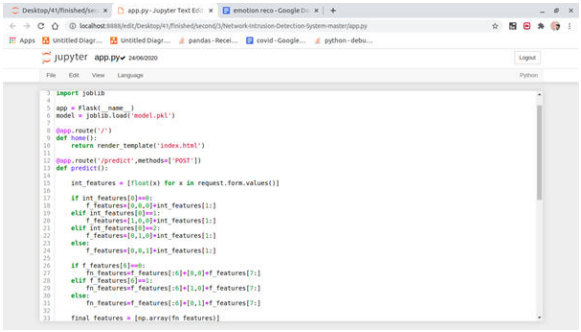
```
from sklearn.model_selection import GridSearchCV
param_grid = {'C': [1, 10],
              'gamma': [0.0001, 0.001]}
grid = GridSearchCV(model, param_grid)

grid.fit(train_X,train_y)
```

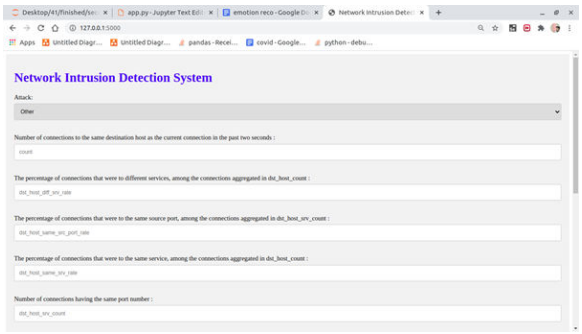
```
print(grid.best_params_)
```

From the score accuracy we concluding the DT & RF give better accuracy and building pickle file for predicting the user input

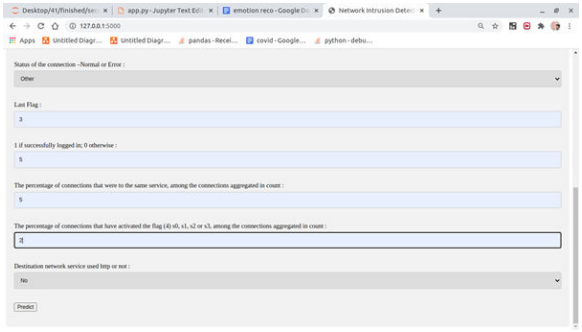
Application



Localhost - in cmd python app.py



Enter the input



Predict attack -

7. CONCLUSION

Right now, estimations of help vector machine, ANN, CNN, Random Forest and profound learning calculations dependent on modern CICIDS2017 dataset were

introduced relatively. Results show that the profound learning calculation performed fundamentally preferable outcomes over SVM, ANN, RF and CNN. We are going to utilize port sweep endeavors as well as other assault types with AI and profound learning calculations, apache Hadoop and sparkle innovations together dependent on this dataset later on. All these calculation helps us to detect the cyberattack in network. It happens in the way that when we consider long back years there may be so many attacks happened so when these attacks are recognized then the features at which values these attacks are happening will be stored in some datasets. So by using these datasets we are going to predict whether cyberattack is done or not. These predictions can be done by four algorithms like SVM, ANN, RF, CNN this paper helps to identify which algorithm predicts the best accuracy rates which helps to predict best results to identify the cyberattacks happened or not.

FUTURE SCOPE

In enhancement we will add some ML Algorithms to increase accuracy

REFERENCES

[1] K. Graves, Ceh: Official certified ethical hacker review guide: Exam 312-50. John Wiley & Sons, 2007.

[2] R. Christopher, “Port scanning techniques and the defense against them,” SANS Institute, 2001.

[3] M. Baykara, R. Das¸, and I. Karado ˘gan, “Bilgi g ¨uvenli ˘gi sistemlerinde kullanilan arac¸larin incelenmesi,” in 1st International Symposium on Digital Forensics and Security (ISDFS13), 2013, pp. 231–239.

[4] S. Staniford, J. A. Hoagland, and J. M. McAlerney, “Practical automated detection of stealthy portscans,” Journal of Computer Security, vol. 10, no. 1-2, pp. 105–136, 2002.

[5] S. Robertson, E. V. Siegel, M. Miller, and S. J. Stolfo, “Surveillance detection in high bandwidth environments,” in DARPA Information Survivability Conference and Exposition, 2003. Proceedings, vol. 1. IEEE, 2003, pp. 130–138.

[6] K. Ibrahimi and M. Ouaddane, “Management of intrusion detection systems based-kdd99: Analysis with lda and pca,” in Wireless Networks and Mobile Communications (WINCOM), 2017 International Conference on. IEEE, 2017, pp. 1–6.

[7] N. Moustafa and J. Slay, “The significant features of the unsw-nb15 and the kdd99 data sets for network intrusion detection systems,” in Building Analysis Datasets and Gathering

Experience Returns for Security (BADGERS), 2015 4th International Workshop on. IEEE, 2015, pp. 25–31.

[8] L. Sun, T. Anthony, H. Z. Xia, J. Chen, X. Huang, and Y. Zhang, “Detection and classification of malicious patterns in network traffic using benford’s law,” in Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), 2017. IEEE, 2017, pp. 864–872.

[9] S. M. Almansob and S. S. Lomte, "Addressing challenges for intrusion detection system using naive bayes and pca algorithm," in Convergence in Technology (I2CT), 2017 2nd International Conference for. IEEE, 2017, pp. 565–568.

[10] M. C. Raja and M. M. A. Rabbani, "Combined analysis of support vector machine and principle component analysis for ids," in IEEE International Conference on Communication and Electronics Systems, 2016, pp. 1–5.

[11] S. Aljawarneh, M. Aldwairi, and M. B. Yassein, "Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model," Journal of Computational Science, vol. 25, pp. 152–160, 2018.

[12] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization." in ICISSP, 2018, pp. 108–116.

[13] D. Aksu, S. Ustebay, M. A. Aydin, and T. Atmaca, "Intrusion detection with comparative analysis of supervised learning techniques and fisher score feature selection algorithm," in International Symposium on Computer and Information Sciences. Springer, 2018, pp. 141–149.

[14] N. Marir, H. Wang, G. Feng, B. Li, and M. Jia, "Distributed abnormal behavior detection approach based on deep belief network and ensemble svm using spark," IEEE Access, 2018.

[15] P. A. A. Resende and A. C. Drummond, "Adaptive anomaly-based intrusion detection system using genetic algorithm and profiling," Security and Privacy, vol. 1, no. 4, p. e36, 2018.