RES MILITARIS
REVUE EUROPEENNE D ETUDES EUROPEAN JOURNAL OF MILITARY STUDIES
Social Science Journal

UNCOVERING NETWORK VULNERABILITIES THROUGH MACHINE LEARNING

KAMARAPU NARESH[1],VARUN MARAMRAJ[2]
[12]Assistant Professor
Department Of CSE
Sree Chaitanya College of Engineering, Karimnagar

ABSTRACT
Unlike earlier times, there have been notable transformations brought about by the progress made in personal computer and communication technology. While there are many advantages to embracing modern technology for individuals, organizations, and governments, some people are not so fond of it. For example, the availability of information, the security of information transfer methods, the protection of sensitive data, and so forth. Fear-based digital oppression is one of the main issues we are currently facing in light of these issues. Due to a number of groups, including the criminal underworld, professionals, and digital activists, digital dread—which has produced several issues for both persons and organizations—has grown to the point where it may jeopardize open and national security. Intrusion Detection Systems (IDS) were created as a result to avoid online attacks at any costs. Currently, port sweep efforts are distinguished by learning the Support Vector Machine (SVM) computations based on the new CICIDS 2017 dataset with 97.80%, 69.79% accuracy rates attained independently. Alternative algorithms that outperform SVM in terms of accuracy include random forest, convolutional neural network (CNN), and artificial neural network (ANN) (93.29, 63.52, 99.93, and 99.11, respectively).

I. INTRODUCTION

1.1 ABOUT THE PROJECCT

In contrast to the past, advancements in personal computer and communication technologies have brought about significant changes. Although using new technology gives individuals, organisations, and governments enormous benefits, some people are messed up against them. For instance, the security of storage areas for sensitive information, information accessibility, and so on. In light of these problems, digital oppression motivated by fear is one of the biggest problems we face today. Digital dread, which caused a lot of problems for individuals and organisations, has reached a point where it might compromise national and open security due to many groups, including the criminal underworld, professionals, and digital activists. As a result, Intrusion Detection Systems (IDS) were developed to keep a strategic distance from online attacks. Currently, learning the support support vector machine (SVM) calculations were used to distinguish port sweep efforts based on the new CICIDS2017 dataset with 97.80%, 69.79% accuracy rates were achieved separately. We may use various algorithms in place of SVM, such as random forest, CNN, and ANN, which can achieve accuracy values of SVM 93.29, CNN 63.52, Random Forest 99.93, and ANN 99.11.

1.2 MOTIVATION

Although using new technology gives individuals, organisations, and governments enormous benefits, some people are messed up against them. For instance, the security of storage areas for sensitive information, information accessibility, and so on. In light of these problems, digital oppression motivated by fear is one of the biggest problems we face today. Digital dread, which caused a lot of problems for individuals and organisations, has reached a point where it might compromise national and open security due to many groups, including the criminal underworld, professionals, and digital activists. In light of this, intrusion detection systems (IDS) were developed to keep a safe distance from cyberattacks.

2. LITERATURE SURVEY

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

One of the most common methods used by attackers to find services they may use to access systems is port scanning. Services that listen to well-known and less well-known ports are executed on all computers that are linked to a LAN or the Internet through a modem. The following details about the targeted systems may be discovered by the attacker via port scanning: what services are active, whose users control those services, if anonymous logins are supported, and whether certain network services call for authentication. Sending a message to each port individually allows for port scanning. The kind of answer sent tells if the port is utilised and may be tested for other vulnerabilities. Network security specialists like port scanners because they can identify potential security flaws on the targeted system. Using the right tools, port scans can be detected, and the quantity of information about open services can be reduced, just as port scans can be run against your systems. Every system that is accessible to the general public has ports that are open and usable. The goal is to prevent access to locked ports and restrict authorised users' access to open ports.

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

Port scanning is a typical task that is quite significant. Computer attackers often use it to describe sites or networks that they are contemplating engaging in hostile activities against. System administrators and other network defence personnel might therefore benefit from seeing port scans as potential precursors to more severe attacks. Network defenders also often utilise it to comprehend and identify vulnerabilities in their own networks. Thus, knowing whether or not a network's defences often do port scanning is of great relevance to attackers. Defenders, on the other hand, often do not want to conceal their port scanning, but attackers do. For the sake of clarity, we shall only refer to the attackers' scanning in the remaining sections of this work. defences attempting to find the scan on the network. On Internet mailing lists and newsgroups, discussions over port scanning's legality and morality often erupt. One wonders whether port scanning faraway networks without the owners' consent is a morally and legally acceptable practise. The majority of jurisdictions are now ambiguous on this. However, we have found that virtually all of the uninvited remote port scans we find in practise turn out to have originated from compromised hosts and are thus extremely likely to be hostile. The administrators of the remote network from which a port scan originated should be informed since in our opinion it is appropriate to consider it at least possibly hostile. The technical issues of how to detect port scans, which are unaffected by the importance that one gives them or by how one chooses to react to them, are the main subject of this study. Additionally, we are concentrating on the issue of using a network intrusion detection system (NIDS) to identify a port scan. We make an effort to consider some of the more blatant strategies an attacker would use to escape detection while maintaining a strategy that is feasible to utilise on busy networks. The rest of this section will describe port scanning, provide many in-depth examples, and go through several methods attackers might attempt to be inconspicuous. The discussion of several earlier port scan detection works is covered in the next section. Following that, we outline the algorithms we want to utilise and provide some very early evidence to support our strategy. Finally, we discuss future directions for this research as well as potential applications. We make the following assumptions about the reader: that they are acquainted with Internet protocols, fundamental concepts of network intrusion detection and scanning, and elementary concepts of probability, information theory, and linear algebra. An attacker may do a port scan for one of two broad reasons: either the primary or secondary goal. The main goal is to collect data on the status and reachability of certain IP address and port (either TCP or UDP) combinations. (While ICMP scans aren't specifically covered in this work, the concepts may obviously be applied to that scenario. The other goal is to overload intrusion detection systems with alarms in an effort to divert or stop network defenders from doing their duties. Since it is simple to identify flood port scans, the focus of this study will mostly be on identifying information collecting port scans. However, a significant concern will be the potential for malevolent information overload design of our algorithm into account. The group of port/IP combinations that the attacker is

RES MILITARIS
REVUE EUROPEENNE D ETUDES EUROPEAN JOURNAL OF MILITARY STUDIES
Social Science Journal

interested in characterising will be referred to as the scan footprint in this article. The script of the scan, which describes the order in which the attacker attempts to investigate the footprint, should be conceptually distinguished from the scan's footprint. The scan's speed, randomness, and other script-related features have no effect on the footprint. The attacker uses the footprint to represent the information collecting needs for her scan, and then she creates a scan script to satisfy those criteria as well as any possible additional non-information gathering requirements (such avoiding detection by an NIDS). At the moment, a horizontal scan is the most used kind of port scan footprint. By this, we imply that a hacker is searching for hosts that expose a certain service in order to use an exploit for it. She then checks all IP addresses within a certain range of interest on the port of interest. Additionally, at the moment, this is mostly carried out sequentially on TCP port 53 (DNS) 2.

Almansob and Lomte used Principal Component Analysis (PCA) and Blameless Bayes with the KDD99 dataset [9].Chithik and Rabbani also employed PCA, SVM, and KDD99 for IDS [10]. The NSL-KDD dataset was used by Aljawarneh et al. in their paper to express their evaluation and exams for their IDS model [11]. Composing inspects demonstrate that IDS [6]–[10] consistently uses the KDD99 dataset.KDD99 was made in 1999 and has 41 highlights. KDD99 is thus outdated and provides no information on modern, novel attack types, such as multiple-day abuses and so on. In this way, we conducted our research using the most recent and cutting-edge CICIDS2017 dataset [12].

Limitations of the current system include: tight regulations, difficulty for non-technical people to utilise, resource restrictions, a need for constant patching, and constant assault.

2.3. Proposed System

The algorithm's key stages are listed below.

1) Every dataset should be normalised.

2) Create training and testing datasets using that dataset.

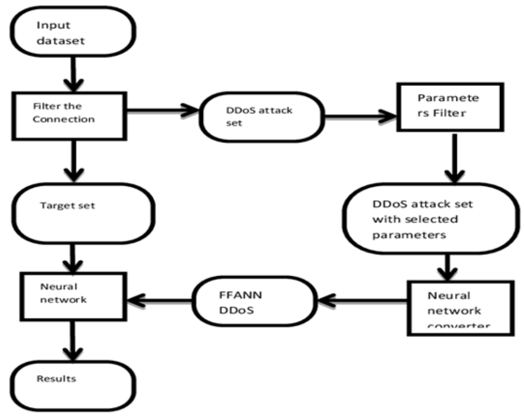
3) Use the RF, ANN, CNN, and SVM algorithms to create IDS models.

4) Assess the performances of each model.

Advantages

- Defence against harmful network assaults.
- Removal of harmful components from an already-existing network and/or their guarantee.
- Prevents people from accessing the network without authorization.
- Block programmes from accessing resources that could be contaminated.
- Protecting sensitive information

2.4 BLOCK DIAGRAM



3. SYSTEM ANALYSIS AND DESIGN

3.2. SOFTWARE REQUIREMENTS

• Python idel 3.7 version (or)
• Anaconda 3.7 (or)
• Jupiter (or) Google colab

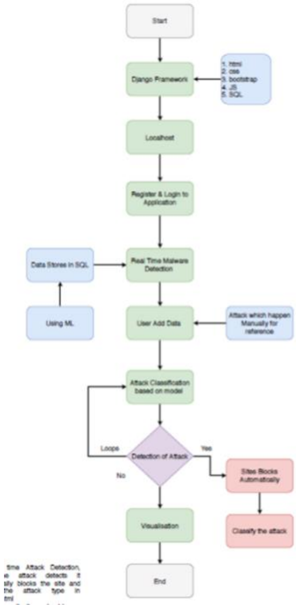
3.3. HARDWARE REQUIREMENTS

• Operating system : windows, linux
• Processor : minimum intel i3
• Ram : minimum 4 gb
• Hard disk : minimum 250gb

3.4. SYSTEM DESIGN

The technique or art of specifying a system's architecture, parts, modules, interfaces, and data in order to meet predetermined criteria is known as system design. It may be considered the application of systems theory to the process of product development. The fields of systems analysis, systems architecture, and systems engineering have some overlap and synergy.

3.4.1 SYSTEM ARCHITECTURE



1. RESULTS AND DISCUSSIONS

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
```

```
import itertools
import seaborn as sns
import pandas_profiling
import statsmodels.formula.api as sm
from statsmodels.stats.outliers_influence import variance_inflation_factor
from patsy import dmatrices
```

```
/usr/local/lib/python3.6/dist-packages/statsmodels/tools/_testing.py:19: FutureWarning: pandas.util.testing is dep
recated. Use the functions in the public API at pandas.testing instead.
  import pandas.util.testing as tm
```

```
from sklearn import datasets
from sklearn.feature_selection import RFE
import sklearn.metrics as metrics
from sklearn.svm import SVC
from sklearn.linear_model import LogisticRegression
from sklearn.feature_selection import SelectKBest
from sklearn.feature_selection import chi2, f_classif, mutual_info_classif
```

```
train=pd.read_csv('/content/drive/My Drive/kdd/NSL_Dataset/Train.txt',sep=',')
test=pd.read_csv('/content/drive/My Drive/kdd/NSL_Dataset/Test.txt',sep=',')
```

DATA PREPARATION

```
In [6]: columns=["duration","protocol_type","service","flag","src_bytes","dst_bytes","land",
"wrong_fragment","urgent","hot","num_failed_logins","logged_in",
"num_compromised","root_shell","su_attempted","num_root","num_file_creations",
"num_shells","num_access_files","num_outbound_cmds","is_host_login",
"is_guest_login","count","srv_count","serror_rate", "srv_serror_rate",
"rerror_rate","srv_rerror_rate","same_srv_rate", "diff_srv_rate","srv_diff_host_rate","dst_host_count","dst_host_sr
"dst_host_diff_srv_rate","dst_host_same_src_port_rate",
"dst_host_srv_diff_host_rate","dst_host_serror_rate","dst_host_srv_serror_rate",
"dst_host_rerror_rate","dst_host_srv_rerror_rate","attack", "last_flag"]
```

```
In [7]: train.columns=columns
        test.columns=columns
```

```
In [8]: train.head()
```

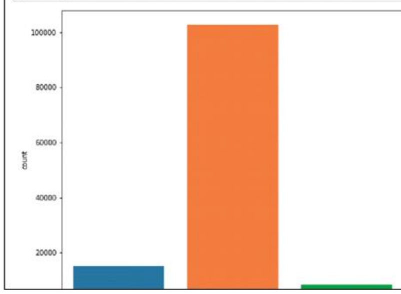
Out[8]:

	duration	protocol_type	service	flag	src_bytes	dst_bytes	land	wrong_fragment	urgent	hot	num_failed_logins	logged_in	num_compromised	root_s
0	0	udp	other	SF	146	0	0	0	0	0	0	0	0	
1	0	tcp	private	S0	0	0	0	0	0	0	0	0	0	
2	0	tcp	http	SF	232	8153	0	0	0	0	0	1	0	
3	0	tcp	http	SF	199	420	0	0	0	0	0	1	0	
4	0	tcp	private	REJ	0	0	0	0	0	0	0	0	0	

```
In [9]: test.head()
```

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)
```

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)
```

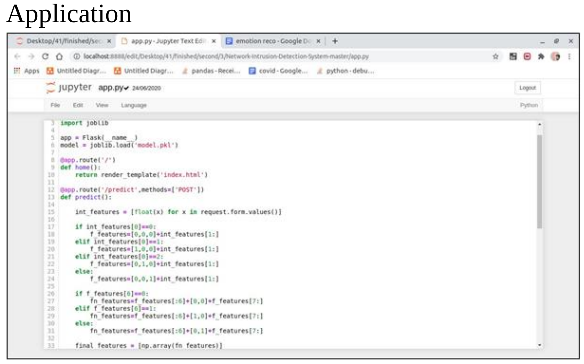
```
gscv_results.best_params_
```

```
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 )
```

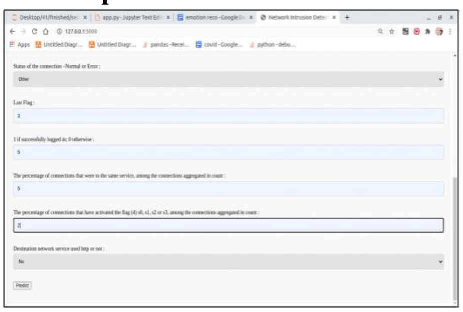
```
radm_test_pred = pd.DataFrame( { 'actual':  test_y,
                            'predicted': radm_clf.predict( test_X ) } )
```

Application



Localhost - in cmd python app.py

Enter the input



Predict attack -

CONCLUSION

Based on the most recent CICIDS2017 dataset, support vector machine, ANN, CNN, Random Forest, and deep learning computations have all just been made available. The outcomes show that the deep learning algorithm produced far better results than SVM, ANN, RF, and CNN. We intend to use this dataset as the basis for various AI and deep learning-based computations in the future, along with Apache Hadoop and sparkle-based assaults, like as port sweeps. These computations' outcomes help to identify network-based cyberattacks. It is possible that there were several attacks over a long period of time. Once these attacks are discovered, the features at which they occurred are recorded in different databases. Thus, we will forecast whether or not a cyberattack has already happened using these data sets. SVM, ANN, RF, and CNN are the four algorithms that can perform these forecasts. This study helps determine which algorithm produces the most accurate results when assessing the validity of cyberattacks.

FUTURE SCOPE

The focus of future attempts to counteract the dynamic nature of cyberattacks will be on enhancing the precision of threat predictions generated by combining a variety of machine learning techniques.

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.