Ilkogretim Online - Elementary Education Online, 2021; Vol 20 (Issue 4): pp. 3191-3197
http://ilkogretim-online.org
doi: 10.17051/ilkonline.2021.04.365

Predicting Distributed Denial Of Service Attack With Mining Based Approach

Varun Sharma , Research Scholar: Department of Computer Science, Desh Bhagat University, Mandi Gobindgarh, India

Dr. R.K. Bathla , Professor: Department of Computer Science, Desh Bhagat University, Mandi Gobindgarh, India

E-mail: rajav20@yahool.com, prof.bathla@gmail.com

Abstract

Wireless sensor network provides resources as per requirement of the user. WSN consists of sensors arranged in sequence for sending and receiving signals. WSN is hampered with the attacks such as distributed denial of service, WORM hole attack etc. This work present ensemble-based approach for detecting DDOS attack caused by malicious users. The impact of DDOS attack on the WSN along with need to tackle DDOS attack is discussed. For accomplishing the detection process, ensembles based voting classifiers is designed. Ensemble based algorithms used for demonstrating the DDOS attack includes Regression mechanism that could be linear or nonlinear in nature, Hyperplane dependent SVM that is also termed as multi support vector machine, random forest that is basically used to select branched that have optimal probability of being attackers., naïve bayes and K-nearest neighbour approach for forming cluster of nearest attribute nodes. The classification accuracy of combined classifier is better as demonstrated within the result section.

Keywords: WSN, ensemble of algorithms, voting classifiers, classification accuracy

1. Introduction

Wireless sensor network(WSN) indicates the networks of spatially distributed sensors. These sensors are generally dedicated and meant to provide specific service only. Performance of WSN depends upon many distinct factors including environmental conditions(Humidity, pollution, sound etc)(Muhammad, Hussain and Yousaf, 2015).. As the authenticated users becomes part of WSN so does unauthorized users. Thus, performance of the WSN also impacted by unauthorized access. Unauthorized users may cause multiple attacks within the network and thereby hampering the performance of the system(Singh, Singh and Kumar, 2017). The most common type of attack includes distributed denial of service attack. This attack is caused through distinct mechanisms. Some of these mechanisms includes

- Flooding

With flooding, thousands of packets are dispersed by the attackers over the network. This will cause other users to be in deadlock situations. This means they will not able access the resources and entire system will be in unstable state.

- Protocol attacks

These type of attacks eats the communication channel along with server resources. Thus, server resources will always be in deadlocked state.

- Application layers

Application layer attacks generally caused through cookies, capturing slots and bad bots. This type of attack could generate multiple identity attacks(AAMIR and ZAIDI, 2013). Source may not able to check the correct destination for transmission pf packets. These type of attacks causes the distortion as well disturbance within the network. Extra energy loss could also be caused through this distributed denial of service attack. The next section presents the literature discussing the DDOS attack impacts on the networks along with the mitigation strategies. Section 3 gives the proposed methodology followed to accurately predict DDOS attack. Section 4 gives the performance analysis and result, section 6 gives the conclusion and future scope, last section gives the references.

2. Literature Survey

The literature presents the tabular comparative analysis of techniques used for the detection of DDOS attack along with impact of attack on WSN.

Table 1: Techniques for DDOS attack prediction along with impacts and issues

References	Techniques	Impact	Issues
(Lara and Ramamurthy, 2016)	Network policy-based mechanism for attack detection	Once attack occurs OpenSec system will fail, and resources will be consumed as a result	Energy consumption of sensor is not considered.
(Ganapathy et al., 2013)	Discussed tools and techniques used for intelligent feature selection and classification of intrusion	Intrusion detection in case of multiple identity attacks could be detected.	Energy consumption and packet drop ratio is not considered
(Kumar and Santhi Tilagam, 2011)	Flooding attack detection	Only low-rate attacks could be detected but high rated attacks may cause traffic within the network	Packet drop ratio is high in this case.
(Bukac and Matyas, 2015)	Traffic pattern analysis in case of DOS standalone attack	Distributed attacks could hamper performance of the network and denial of service requests	Lifetime of the network will be reduced but not considered in this literature
(Behal, Kumar	D-Face based	Internet domain	Internet domain-

and Sachdeva, 2018)	technique for detecting the impact of Distributed denial of service attacks	will be impacted with this type of attack	based attack could reduce the packet to base station and should be a part of DDOS attack metric consideration
(Meenakshi, Kumar and Behal, 2021)	Deep learning-based approach for DDOS attack detection	LSTM based mechanism applied detect the impact of resource wastage within WSN	Low classification accuracy of the detection process is an issue
(Nguyen et al., 2021)	Gaussian model collaborated with deep learning for early detection of denial-of-service attack or DDOS	This model evaluates the impact on network resources and also determine the percentage resource consumption	Low classification accuracy of the detection process.

3. Proposed Methodology

The methodology followed is based upon different classifiers including KNN, random forest, SVM, naïve bayes, logistic regression and ensemble based coting classifier. The ensemble-based approach produced better classification accuracy as compared to individual approaches. Fine tuning of classification accuracy also resulted in better learning rate and low error rates. Dataset for demonstration is fetched from Kaggle. The pseudo codes for the approaches is presented in this section.

Table 2: Pseudo code corresponding to different classifiers

KNN
K_N=FindNearestNeighbour(K=7) Fit_KNN_Model(Training_Data) Predictions = Kpredict(Test_Data)
Outcome-Prediction
array([1, 0, 0, 0, 0, 0, 1, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 1, 1, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 1, 1, 1, 0, 0, 1, 0, 0, 0, 1, 0, 1, 1, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0, 0, 1, 0, 1, 1, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 1, 1, 0, 0, 1, 1, 1, 1, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, 1, 1, 1, 1, 1, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 1, 1, 0, 0, 0, 0, 1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 1, 0], dtype=int64)
Logistic Regression
L_R=FindRegressioncofficienct() Fit_LR_Model(Training_Data) Predictions = LRpredict(Test_Data)

```
array([1, 0, 0, 0, 0, 0, 1, 1, 0, 1, 0, 1, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1,
       0, 1, 1, 0, 1, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 1, 1, 1, 0, 0, 0,
       0, 1, 0, 0, 1, 0, 1, 1, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 0, 1, 0, 0,
       1, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1,
       0, 0, 1, 1, 0, 0, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1, 0,
       1, 1, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 1, 0, 1, 1, 1, 0, 0, 0, 1,
       0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 1, 0],
      dtype=int64)
```

SVM

Support_Vector=Findvetcors(kerner="linear")
Fit_SVM_Model(Training_Data)
Predictions = SVMpredict(Test_Data)

```
array([0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
      dtype=int64)
```

Random Forest

R_F=RFClassifer ()
Fit_RF_Model(Training_Data)
Predictions = RFpredict(Test_Data)

```
array([1, 0, 0, 0, 0, 0, 1, 1, 0, 1, 1, 1, 0, 0, 0, 1, 1, 0, 1, 1, 0, 1,
       1, 1, 1, 0, 1, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 1, 1, 1, 0, 0, 0,
       0, 0, 0, 0, 1, 0, 1, 1, 1, 0, 1, 1, 1, 0, 1, 0, 1, 0, 0, 1, 0, 1,
       1, 0, 1, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0,
       0, 1, 1, 1, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1, 0,
       1, 1, 1, 1, 1, 0, 1, 0, 1, 0, 1, 0, 1, 1, 0, 1, 1, 1, 0, 0, 0, 1,
       0, 1, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 1, 0],
      dtype=int64)
```

NB = Naïve_Bayes_NB()
NB.fit(Train_Data)
NB_Predict=NB.predict(test_data)

```
array([0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
      dtype=int64)
```

Almost all the classifiers given the mix result where '1' indicates that attack has been detected and '0' means no attack is detected. The voting classifiers with fine tuning gives the result by accommodating good features of all the classifiers and present the result as per majority. This means that if 3 out of five classifiers give '1' as prediction and 2 classifiers yields ‘0’ for the same data, then voting classifier will give ‘1’ as prediction. The pseudocode for the same is given as under

Table 3: Voting classifier demonstration

Voting Classifier
K_N=FindNearestNeighbour(K=7) L_R=FindRegressioncofficienct() Support_Vector=Findvetcors(kerner=”linear”) R_F=RFClassifer () NB = Naïve_Bayes_NB() V_C = Vote_Classifier(estimators=[('L_R', clf1), ('R_F', clf2), ('NB', clf3),('Support_Vector',clf4),('K_N',clf5)], voting='soft', weights=[2,1,1,2,2]) V_C.fit(Train_Data)

`VotingClassifier(estimators=[('lr', LogisticRegression()),`
`('rf', RandomForestClassifier()),`
`('nb', GaussianNB()),`
`('svc', SVC(kernel='linear', probability=True)),`
`('knn', KNeighborsClassifier(n_neighbors=7))],`
`voting='soft', weights=[1, 1, 2, 2, 1])`

V_C.Predict = VC.predict(test_data) V_C_Predict

`array([1, 0, 0, 0, 0, 0, 1, 1, 0, 1, 0, 1, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1,`
`0, 1, 1, 0, 1, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 1, 1, 1, 0, 0, 0,`
`0, 1, 0, 0, 1, 0, 1, 1, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 0, 1, 0, 0,`
`1, 0, 1, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1,`
`0, 1, 1, 1, 0, 0, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1, 1,`
`1, 1, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 0, 1, 1, 1, 1, 1, 0, 0, 0, 1,`
`0, 1, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 1, 0],`
`dtype=int64)`

4. Performance analysis and result

From the obtained result we conclude that only Support vector machine is generating abnormal result for the presented data. The proposed mechanism however is based on best possible result from all the algorithms. This means in case , 3 or more of the classifiers are generating ‘1’ and last classifier is generating ‘0’ then out voting classifier will generate ‘1’ as overall result.

Table 5: Performance Analysis and result in terms of different metrics

Curve_ROC=Obtain_Score(Test_Data) Final_Result = pd.DataFrame([['V_C ']], cols = ['Model', 'Accuracy', 'Precision', 'Recall', 'F1 Score','ROC']) Final_Result = F_append(Final_Results, ignore_index = True)

	Model	Accuracy	Precision	Recall	F1 Score	ROC
0	Logistic Regression	0.720779	0.590164	0.666667	0.626087	0.708333
1	Random Forest	0.694805	0.550725	0.703704	0.617886	0.696852
2	KNN	0.766234	0.655172	0.703704	0.678571	0.751852
3	SVC Linear	0.649351	0.000000	0.000000	0.000000	0.500000
4	NB	0.649351	0.000000	0.000000	0.000000	0.500000
5	Voting Classifier	0.727273	0.590909	0.722222	0.650000	0.726111

Conclusion and Future scope

This work presented the detection of DDOS attack using distinct classifiers including logistic regression, random forest, KNN, SVM, naïve bayes and voting based classifiers. The result section demonstrates that result of logistic regression classifier in the detection process is highest. However still, classification accuracy is below desired levels. To accomplish better result, voting based classifier with fine tuning mechanism is applied. Voting classifier yield better accuracy but improvement is limited. In future, outlier detection mechanism along with missing values handling could be used for better results.

References

AAMIR, M. and ZAIDI, M. A. (2013) 'A Survey on DDoS Attack and Defense Strategies: From Traditional Schemes to Current Techniques', Interdisciplinary Information Sciences, 19(2), pp. 173–200. doi: 10.4036/iis.2013.173.

Behal, S., Kumar, K. and Sachdeva, M. (2018) 'D-FACE: An anomaly based distributed approach for early detection of DDoS attacks and flash events', Journal of Network and Computer Applications, 111, pp. 49–63. doi: 10.1016/j.jnca.2018.03.024.

Bukac, V. and Matyas, V. (2015) 'Analyzing traffic features of common standalone DoS attack tools', Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 9354, pp. 21–40. doi: 10.1007/978-3-319-24126-5_2.

Ganapathy, S. et al. (2013) 'Intelligent feature selection and classification techniques for intrusion detection in networks: a survey', EURASIP Journal on Wireless Communications and Networking, 2013(1), p. 271. doi: 10.1186/1687-1499-2013-271.

Kumar, A. and Santhi Tilagam, P. (2011) 'A Novel Approach for Evaluating and Detecting Low Rate SIP Flooding Attack', International Journal of Computer Applications, 26(1), pp. 31–36. doi: 10.5120/3067-4192.

Lara, A. and Ramamurthy, B. (2016) 'OpenSec: Policy-Based Security Using Software-Defined Networking', IEEE Transactions on Network and Service Management, 13(1), pp. 30–42. doi: 10.1109/TNSM.2016.2517407.

Meenakshi, Kumar, K. and Behal, S. (2021) 'Distributed denial of service attack detection using deep learning approaches', Proceedings of the 2021 8th International Conference on Computing for Sustainable Global Development, INDIACom 2021, pp. 491–495. doi: 10.1109/INDIACOM51348.2021.00087.

Muhammad, S., Hussain, S. and Yousaf, M. (2015) 'Neighbor Node Trust Based Intrusion Detection System for WSN', Procedia - Procedia Computer Science, 63, pp. 183–188. doi: 10.1016/j.procs.2015.08.331.

Nguyen, T. T. et al. (2021) 'Detection of unknown DDoS attacks with deep learning and Gaussian mixture model', Proceedings - 2021 4th International Conference on Information and Computer Technologies, ICICT 2021, pp. 27–32. doi: 10.1109/ICICT52872.2021.00012.

Singh, K., Singh, P. and Kumar, K. (2017) 'Application layer HTTP-GET flood DDoS attacks: Research landscape and challenges', Computers & Security, 65, pp. 344–372. doi: 10.1016/j.cose.2016.10.005.