Detecting Distributed Denial of Service (DDoS) in SD-IoT Environment with Enhanced Firefly Algorithm and Convolution Neural Network

Sivanesan N. (profnsivanesan@gmail.com)

Vel's College of Science: Vels Institute of Science Technology & Advanced Studies

https://orcid.org/0000-0002-4408-7641

Research Article

Keywords: Internet of Things, Software-Defined networks, Denial of Service, Distributed Denial of services, Convolutional Neural Network, Firefly Algorithm

Posted Date: May 9th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1509704/v1

Detecting Distributed Denial of Service (DDoS) in SD-IoT Environment with Enhanced Firefly Algorithm and Convolution Neural Network

Sivanesan N. [1] K. S. Archana[2]

Research Scholar, School of Computer Science and Engineering, Vels University, Chennai, India.

Asst. Professor, School of Computer Science and Engineering, Vels University, Chennai, India.

Abstract:

In Recent years, Network security becomes essential due to increase in usage of Smart phones and Internet of Things (IoT) devices. An IoT device plays a vital role in day to day life of human being. Such IoT devices are less secured and mostly used under abandoned environment. Recently these devices are widely affected by Distributed Denial of Service (DDoS) attack. DDos is one of the risky threats that destroy the critical network services. The extreme flow of packets in a network results in attack. Single source attack raises the Denial of Service (DoS); on the other hand attack rises from multiple servers referred to as DDoS. Researchers have developed software-defined networks (SDN) to effectively handle IoT equipment. To overcome above issue, we use an updated firefly algorithm to optimize the convolutional neural network (CNN) for detection of DDoS attacks in software-defined Internet of Things (SD-IoT) environment. Experimental result shows that our proposed model achieves 98% accuracy over detection of DDoS attack.

Keyword: Internet of Things, Software-Defined networks, Denial of Service, Distributed Denial of services, Convolutional Neural Network, Firefly Algorithm.

1. Introduction:

The Internet of Things (IoT) is the connectivity of material entities that can see, absorb, and respond to their surroundings using fundamental network protocols. It is the result of advancements in embedded technology, Wireless Sensor Networks (WSNs), and standard networking protocols for allowing communication among smart things [1]. IoT devices are most often used in almost every domain, including smart homes, smart cities, smart grid systems, manufacturing, transportation, healthcare, and smart disaster management systems, where human involvement is problematic. There are numerous issues in IoT networks that necessitate the evolution of traditional internet topology [2]. Network security has recently become especially critical because DDoS [3] poses a serious threat to network safety. DDoS attack becomes common as cyber threats due to increase in IoT devices, complexity and growth of hire attack service [4]. The DDoS attack is considered to be the heaviest in history, with a peak bandwidth of 1.35 Tbps. The number of IoT devices with risk factors has expanded substantially, and by 2025, there will be 24.6 billion linked gadgets [5]. A DDoS attack prevents genuine internet users from gaining access the suspect's services. To accomplish this, the attacker floods the targeted system with trash packets, overloading its processing and storage capacity and eventually causing the system to crash. The botnet army is the most often used way for conducting large-scale attacks [6]. DDoS attacks on IoT devices are becoming more common, causing many IoT devices to malfunction and leaking personal information. As a result of the rapid expansion of IoT, relevant network security measures should be upgraded at the same time.

Above mentioned DDoS issue were minimized by using SDN. Due to the success of SDN in network management and security maintenance, an increasing number of domestic and international researchers have attempted to incorporate their conceptual models into IoT and developed a software-defined internet of things (SD-IoT) framework. The segmentation of the control plane and the forwarding plane is a crucial feature of the SD-IoT framework. The SD-IoT controller often operates on a fast processing platform, enabling security techniques and detection procedures that regular network infrastructure cannot provide [8].

DDoS attacks are generally one of the most challenging malicious activity to detect [9]. DDoS attacks are classified into two types: attacks that use resource bandwidth and attacks that consume system resources. Resource bandwidth attacks employ a large number of zombie servers to swiftly create a massive quantity of traffic that converges on the victim's server and entirely seizes its network bandwidth resources. Because of the numerous permutations of DDoS attacks, identification is becoming extremely difficult. Many DDoS attackers, use mixed protocol packets to attack their victims. To deal with a range of attack strategies, more complete and compelling defensive techniques should be created [10-12]. The conventional signature-based detection mechanisms cannot identify innovative DDoS assaults, whilst the more regularly adopted detection methods based on statistical abnormalities are limited by the detection threshold. To address the shortcomings of statistical anomaly detection approaches, attack detection strategies based on machine learning methods are being investigated. Deep learning algorithms have been acknowledged for their ability to classify DDoS assaults and regular traffic. Deep learning algorithms can extract from the original data flow the features required by a DDoS attack and regular traffic flow. Furthermore, in the past, most attack detection solutions based on deep learning algorithms were implemented in conventional networks and required an excessive amount of resource supply. Present DDoS attack detection technologies, are not built for SD-IoT network offline attack detection. Detection algorithms in real SD-IoT networks must interact with networking traffic flows that have a preset data packet window [13].

Based on the most recent DDoS detection requirements, this paper proposes a unique detection approach that merges CNN algorithm into SD-IoT controller. The following are some of the most important contributions made by this paper:

1) We presented a security architecture for SD-IoT. This architecture includes Internet of Things infrastructure, IoT switches that link IoT gateways, and an SD-IoT controller.

2) A dataset-independent data packet preprocessing approach that needs detection algorithms to handle flow fragments obtained in preset packet windows. The SD-IoT controller with flexible programmability obtains the packet header from the SD-IoT switch on a regular basis, which significantly reduces the SD-IoT controller's processing overhead.

3) To improve detection accuracy, we propose an improved firefly approach for optimizing neural network architecture.

4) The detection approach in this research employs a CNN algorithm to study potentially malicious traffic before detecting DDoS attacks. The approach presents a greater detection accuracy while also having a minimal processing expenditure.

In [14], DDoS attacks were detected using the entropy of the target IP address. When a packet is received and the switch is unsure what to do with it, it sends a Packet-in message to the SDN controller. The target IP address was contained in packet-in messages, and the controller estimated the entropy of the target IP address. In the controller, configure the sample window size and threshold. It was discovered that a DDoS attack happened when the estimated entropy value is much less than the predefined threshold value. In a policy-based detection technique [15], the network flow investigated is deemed acceptable if the flows identified correspond to a given policy. In contrast, the network traffic being investigated are deemed harmful. In [16], researcher offered the IoT-IDM framework, that was implemented on SDN and included an IoT attack vulnerability management system. It has the ability to detect the victim server and prevent the attack. The Internet of Things (IoT) is a platform that can connect everything and anyplace. Security in the context of IoT is a significant issue [17]. Numerous problems impede the security of IoT devices and their end-to-end connectivity in an IoT context [18].

This paper is structured as follows: section 2. DDoS attacks in proposed SD-IoT framework, section 3. DDoS attack detection model, section 4. Performance evaluation, section 5. Conclusion

2. DDoS attacks in proposed SD-IoT framework

Figure 1 shows the proposed SD-IoT framework that is a broader version of SDN paired with IoT. In the proposed SD-IoT frame we have three layers like Application, control and user infrastructure layers.

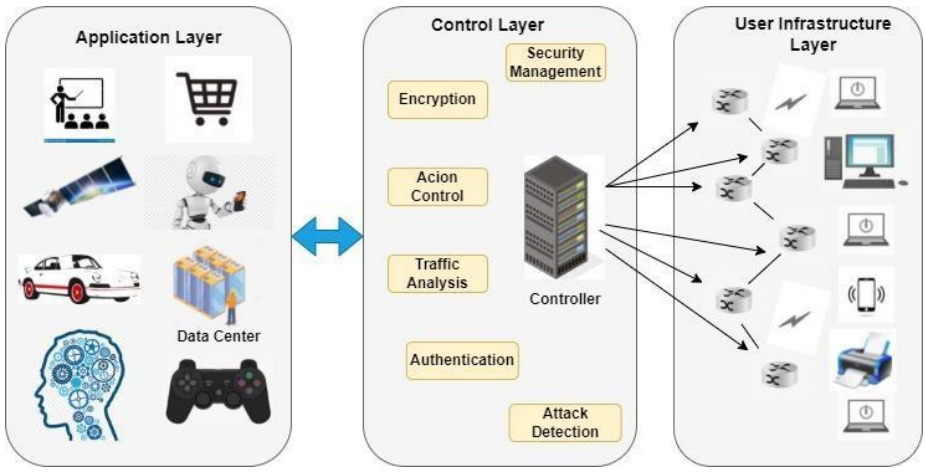

Figure 1. SD-IoT Framework

The user infrastructure layer consists of network equipment as well switches that are supported for SD-IoT framework. In the proposed switches that are used for SDN and gateway used for IoT are independent. Our SD-IoT switches may be used to link IoT drivers and sensor equipment such as personal computers, digital cameras, and smart phones.

The control layer includes the SD-IoT controller. The SD-IoT controller uses the downstream interface to receive topological information from IoT devices, create a global perspective, and then fulfil network management operations on the infrastructure layer such as threat detection, traffic engineering, and load balancing. Simultaneously, this layer offers the API that the application layer can use.

On this architecture, the application layer consists of a range of apps that operate in the IoT server and communicate with the SD-IoT controller via a northbound interface. Simultaneously, it is beneficial to developers. SD-IoT developers no longer have to worry about variances in underlying device communication protocols because of the use of a common south interface protocol, which simplifies application development, streamlines application deployment, and lowers network maintenance costs.

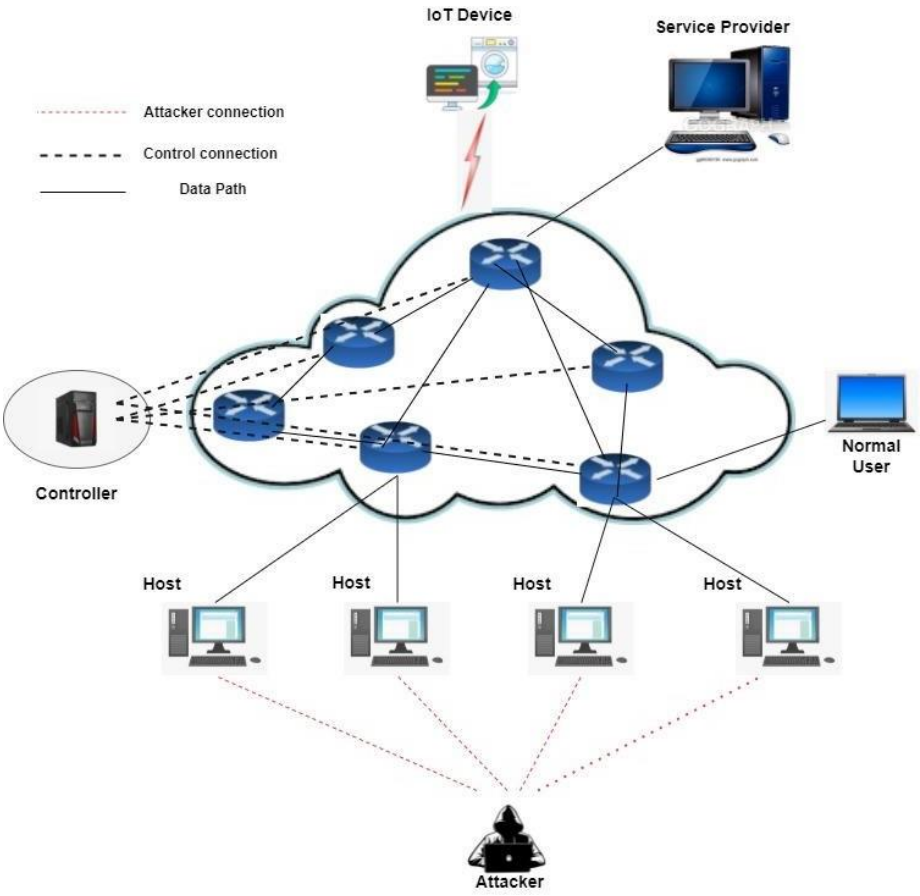

Figure 2. DDoS Attack in SD-IoT

The SD-IoT controller is responsible in our architecture for centralized logical control of IoT devices. The advantages of logical centralized control are configuration and management, but there are also clear disadvantages, such as the system being vulnerable to attack. Our suggested programmable SD-IoT architecture is similar to SDN, which aids in DDoS detection. DDoS attack in SD-IoT framework is shown in Figure 2.

i. SD-IoT switches receive packets from both DDoS attacker and normal user. Attack script generates the attack packets.
ii. Details about SD-IoT packet header are collected in regular basis by using SD-IoT controller.
iii. Outcome of the SD-IoT controller is processed to next step by using SD-IoT switches.

3. Proposed Enhanced Firefly Algorithm with CNN (EFACNN) model for DDoS attack Detection

Enhanced Firefly Algorithm with CNN (EFACNN) model for detecting the DDoS attack is shown in Figure 3.

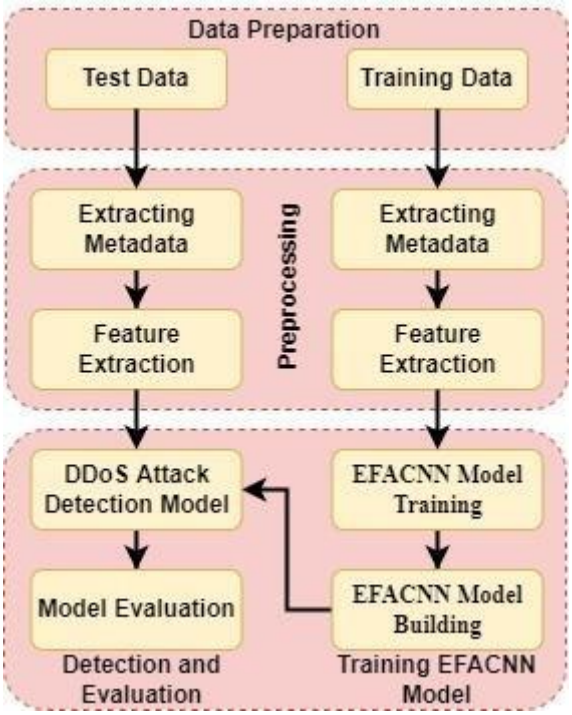

Figure 3. EFACNN model for detecting DDoS attack

3.1 Data preparation and Preprocessing

3.1.1 Data collection

Current data packet header information is collected periodically from SD-IoT switches by the data collection module. Algorithm is created and placed in SD-IoT Controller to circulate the procedures to the SD-IoT switches to collect the packet information. The time interval for collecting the data is set as minimum as possible to avoid the DDoS attack and damage or loss of packets. Similarly less time interval will enhance the interaction between controller and switch. For the proposed work we set 5 seconds as an interval for collecting the packet information.

Algorithm 1 For collecting the packet header information

Input:
Packet.list
Output:

```
1. Packet.list getindex
2. NL←0 (initializing)
3. ϵs ← 0
4. for all packet∈ NDPS do
5.     for index init 0 to NL.length() by 1 do
6.         if packet.list ==NL[index] then
7.             break
8.             if index==NL.length() then
9.                 NL.append(packet.list)
10.                ϵs[index].append(paket.header)
11. End
```

where, NL denotes 5-list network flow, ϵ_s denotes sorted packet that are collected through network flow, NDPS denotes network data packet set.

3.1.2 Preprocessing data packets

Data packets that belongs to same network contains same source and destination IP address and source and destination port address. *S*-vector used to store the packet information such as source and destination IP address. *x* data packets are flowing across the network, from that *m* packets are consider as the packet window. For experimental purpose we have taken nearly 250 packets as window.

Algorithm 2 For collecting the packet header information

Input:
Data flow in network traffic
Output:

```
1. Packet.list getindex
2. Set of window in sub-flow←0 (initializing)
3. Ni ← 0
4. feature vector ← 0
5. for Ni < NL.length do
6.     sfw ← 0
7.     sfwcol ← 0
8.     sfwrow ← 0
```

9. **While** sfw.length()<m **do**
10. $sfw[sfw_{col}, sfw_{row}] \leftarrow \epsilon_s[N_i].shift() \approx transfer\ \epsilon_s[N_i]\ to\ m$
11. if true== $\epsilon_s[N_i].empty\ then \approx move\ to\ next\ flow\ table$
12. $N_i \leftarrow N_i + 1$
13. $sfw_{col} \leftarrow 0$
14. $sfw_{row} \leftarrow sfw_{col} + 1$
15. **for all** $sfw \in set\_of\_window$ **do**
16. **feature_vector(m.features)**
17. **feature_vector**← $Label.[sfw]$
18. **End**

where, sfw denotes the sub-flow window, N_i denotes network flow index.

3.1.3 Feature Extraction

Characteristics of data packets that are gathered from data collection and preprocessing are computed through feature extraction module. These characteristics are closely related to detect the DDoS attack. From the literature we selected 6 characteristics that are closely associated with DDoS attack.

In network, Packet flow has various dissimilarities. Normal network flow carries extensive packets. DDoS attack network flow will generate fake IP address for source and try to communicate in the network to attack the victim host. To avoid such DDoS attack in network we selected Number of packets per network flow (NP_{nf}) as one of the feature.

Another DDoS attack strategy is making the data packets as tiny as possible. Usually normal data packet will be slightly larger in size. Where else attacker tries to reduce the packet size in bytes to attack the victim host as quick as possible. To avoid it Number of bytes per network flow (NB_{nf}) is added as a feature to detect the DDoS attack.

Next feature is Time taken for each data packet to flow in network, Normal data packet that flow in network takes more amount time. In case of attack, abnormal flow of data packet takes place.

Remaining features that are selected to verify the Source and Destination IP address. Since DDoS attack mainly occur with fake source and destination IP address.

3.2 Enhanced Firefly Algorithm

Firefly Algorithm (FA) is inspired by the flashing behavior of the fireflies. The less bright fireflies get attracted towards brighter fireflies taking into account the media around the problem domain.

Moving from one less brighter fireflies to brighter fireflies is considered as a single iteration. Best optimal solution is searched by step by step iteration process. In the algorithm, each firefly will be represented as a vector point. Position of the firefly is denoted by candidate solution s_c. where $c = 1,2, \dots . P$. The brightness and attraction of firefly can be represented as shown in Equation (1).

$$f_b = f_{b0} e^{P \forall_{ab}} \tag{1}$$

$$MD_a(\forall) = MD_{a0} e^{\forall_{ab}{}^2} \tag{2}$$

where, f_b denotes the fluorescence firefly brightness over $\forall$ with assuming as zero. *P* denotes the light absorption parameter. Position of firefly will be denoted as ab. MD denotes the maximum degree of attraction. Distance between two firefly can be calculated by using Euclidean distance as mention in equation (3).

$$\forall_{ab} = \|s_a - s_b\| = \sqrt{\sum_{k=1}^{m} (s_{ak} - s_{bk})^2} \tag{3}$$

When less brightness fireflies started moving to brighter fireflies then operational speed can be improved by substituting equation (4) in equation (2).

$$(\forall) = \frac{MD_0}{1+(P*\forall_{ab}{}^2)} \tag{4}$$

Firefly moving from position *a* to position *b* can be calculate by using equation (5)

$$s_a^{n+1} = s_a + \frac{MD_0}{1+P\forall_{ab}^2}(s_a - s_b) + (j - .5) \tag{5}$$

where, *j* denotes the random value between 0 to 1. Number of iteration denoted by *n* and φ denotes the step size.

Till we have seen how firefly moves from one position to another and how the brightest firefly is identified. Usually position of firefly updated based on the attraction of fluorescence brightness and search is made based on random and global attributes. So to improve the accuracy and to obtain optimal solution, we proposed a methodology that updates the step size periodically. with update in position of firefly periodically will decrease the search iteration of fluorescence brightness. Equation (6) describes how to calculate the distance between individual and group center for finding the group diversity initially.

$$d^i = \frac{1}{|S_p|} \sum_{a=1}^{|s_p|} \sqrt{\sum_{b=1}^{m} (s_{ab} - \bar{s}_b)^2} \tag{6}$$

where, d^i denotes the index of diversity. Size of the firefly population is denoted by $|s_p|$. Number of iterations is increased to decrease the linear decrease function ω. Equation (7) used to calculate the ω with two factors, maximum number of iteration (max_i) and current number of iteration ($curr_i$).

$$\omega = \frac{max_i - curr_i}{max_i} \tag{7}$$

3.3 Detection of DDoS Attack

DDoS attack detection is considered as a classification problem since flow of attack network and normal network flow is entirely different. Figure 3 shows the attack detection module work flow in detail for the proposed EFACNN. The CNN layers initially assigned with random weights and bias value. Later based on the output of the first epoch, weights and bias value is update with respect to the error calculated between output obtained and actual output. Learning rate and structure of network are considered as the hyperparameter. The hyperparameter selection plays an important role to increase the accuracy and performance of detecting DDoS attack. Such hyperparameter are selected based on the Enhanced Firefly algorithm that are discussed in previous section. Each firefly are considered as a single hyperparameter. Root mean square is used for measuring the training accuracy of the proposed model.

Proposed work is build on deep learning concept, it consist of input layer that reads the data from network and divides into 5 different list. From this data it extracts the features and forms the 2-D matrix. Next convolution layer consist of kernels which are mapped with the input features obtained from input layer. Size of the kernel is defined as h. with the help of convolution layer useful features are gathered for classifying the DDoS attack request and normal network flow. Linear rectifier function (ReLU) activation function is used in Convolutional Layer and calculated as shown in equation (8).

$$R(Conv(input_{matrix})T_k, bias_k \quad (8)$$

$$R(b) = \max(0, b) \quad (9)$$

where, T_k denotes the Training stage of k^{th} filter.

To down sample the features max pooling layer is used and then output obtained from pooling layer is classified using fully connected layer with sigmoid activation function σ. Sigmoid function will generate the output in the range of 0 to 1. Since we going to classify the attack and normal network flow it flows either 0 or 1.

4. Experimental Evaluation and Performance Assessment

4.1 Experimental Setup

SD-IoT environment is designed by using Software defined network solution for wireless sensor network (SDN-WISE) with help of Contiki Operating System. Experiment is carried out using virtual box platform installed in windows 10 operating system. Ubuntu OS is installed in virtual box with open source SD-IoT controller and switches. Proposed firefly algorithm with CNN is developed using Tensorflow library. Similarly UDP, TCP and ICMP packets for both trusted user and attacker is created by using scapy with the help of python script.

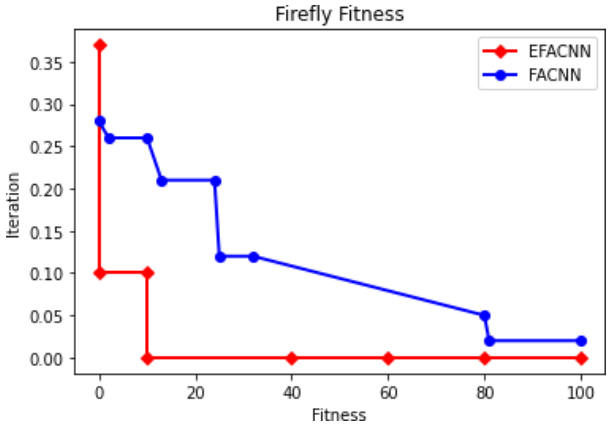

Figure 4. Firefly fitness improvement

Network topology designed using mininet with 6 SD-IoT switches, 1 controller and 50 devices connected into it for detecting DDoS attack is shown in Figure 4. The devices that are connected with the network include wired and wireless equipment. sw_1 ... . sw_6 denotes SD-IoT switches and hs_1 ... ... hs_{50} denotes host that are connect with the network. For experimental purpose

hs_2 , hs_4 and hs_6 that are connected with sw_2 are considered as attack host. That sends fake request continuously to the hs_{40}. Performance of detecting DDoS attack is measured by using precision, recall and F1-score metrics. Table 1. Describes the data that are used for experimental purpose.

Table 1. Sample data used for Detecting DDoS Attack

Purpose	Trusted user	DDoS Attacker
Training	177856	137837
Testing	123742	87864

From the total number of data, percentage of data that are recognized correctly is computed as accuracy of detecting DDoS attack.

$$Acc\% = \frac{True\ positive + True\ Negative}{True\ Positive + True\ Negative + False\ Positive + False\ Negative} \quad (10)$$

Total number of attack packets that are actually determined by the model is denoted as precision.

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive} \quad (11)$$

From the total number of attack that are estimated by the model as DDoS attack is denoted by recall

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative} \quad (12)$$

F1 score is the mean of precision and recall.

$$F1\ score = \frac{2 * precision * Recall}{Precision + Recall} \quad (13)$$

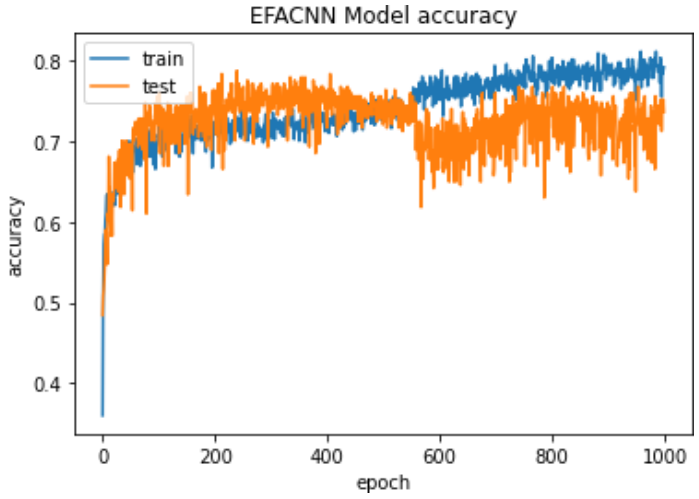

Figure 5. EFACNN model Accuracy

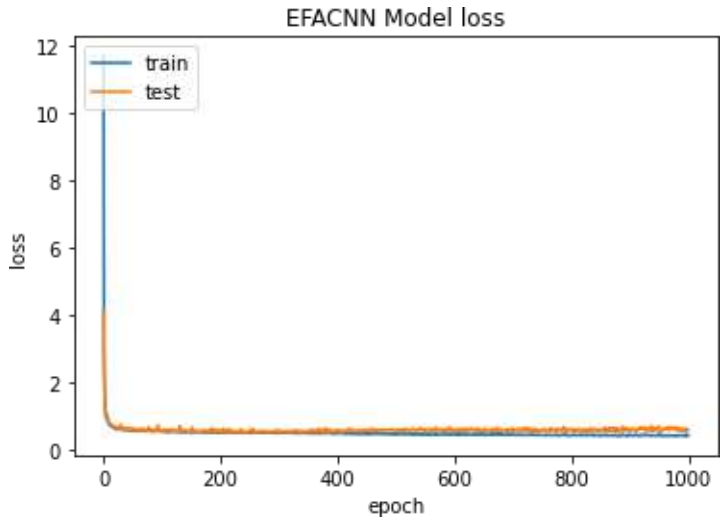

Figure 6. EFACNN model loss

4.2 Evaluation of EFACNN model

In the proposed model Enhanced firefly algorithm with CNN is used with two dimensional layers. Firefly is used for optimizing the input features and hidden layer node. Size of firefly is set to 60, maximum epoch is set to 1000 and step factor is set to 0.6. Figure 4 shows the improvement in fitting the firefly and Figure 5 and 6 denotes the accuracy and loss for the proposed EFACNN model. The depth of the CNN layer plays an important role in improving the accuracy. For the experimental purpose we used 4 EFACNN model with different number of layers. Parameter used for all 4 models are described in Table 2.

Table 2. Details of parameter used in 4 different EFACNN model

Parameters	Model 1	Model 2	Model 3	Model 4
No.of Layers	2 Conv + 2 pooling + 2 Fully Connected Layer	2 Conv + 2 pooling + 3 Fully Connected Layer	3 Conv + 2 pooling + 2 Fully Connected Layer	3 Conv + 3 pooling + 2 Fully Connected Layer
Activation Function	Sigmoid and ReLU			
Size	60			
epoch	1000			
Step factor	0.6			

Figure 7. shows the performance of proposed model is compared with models that are developed in [19] and [20].

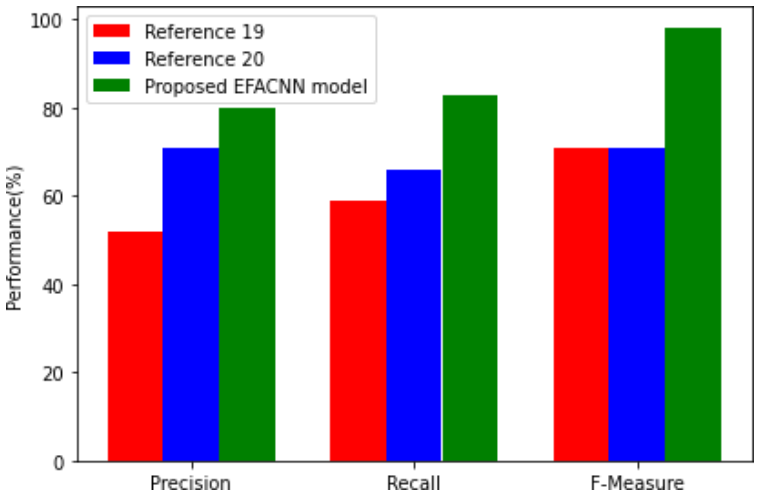

Figure 7. Comparison between proposed model Vs existing models

5. Conclusion

In network security, DDoS attack still considered to be an important treat. In the proposed work we designed a SD-IoT framework that contains switches, controller and IoT devices. For selecting and extracting efficient features from the input data enhanced firefly algorithm is used. By using firefly algorithm we reduces the time taken for detecting the DDoS attack. Since firefly algorithm will update the position regularly for each 5 seconds which reduce the time for searching. CNN is used for classifying the attack network flow and normal network flow. We used 4 different CNN model with different set of layers and parameters that improves the accuracy of detecting DDoS attack. Performance of proposed algorithm is measured by using precision, recall and F1 score metrics. The proposed model shows nearly 98% of accuracy when compared with existing methodology.

References

[1] Gubbi J, Buyya R, Marusic S, Palaniswami M. Internet of things (IoT): a vision, architectural elements, and future directions. Future Gener Comput Syst 2013;29 (7):1645–60.

[2] Neshenko N, Bou-Harb E, Crichigno J, Kaddoum G, Ghani N. Demystifying IoT security: an exhaustive survey on IoT vulnerabilities and a first empirical look on internet-scale IoT exploitations. IEEE Commun Surv Tutor 2019;21(3):2702–33.

[3] I. Cvitic, D. Perakovi ´ c, B. Gupta, K. K. R. Choo, Boosting-based DDoS detection in internet of ´ things systems, IEEE Int. Things J., 2021 (2021). doi: 10.1109/JIOT.2021.3090909.

[4] F. Song, Z. Ai, H. Zhang, I. You, S. Li, Smart collaborative balancing for dependable network components in cyber-physical systems, IEEE Trans. Ind. Inf., 17 (2021), 6916–6924. doi: 10.1109/TII.2020.3029766.

[5] F. O. Catak, Two-layer malicious network flow detection system with sparse linear model based feature selection, J. Nat. Sci. Found. Sri Lanka, 46 (2018), 601–612. doi: 10.4038/jnsfsr.v46i4.8560.

[6] Bertino E, Islam N. Botnets and internet of things security. Computer 2017;50(2):76–9.

[7] K. K. Karmakar, V. Varadharajan, S. Nepal, U. Tupakula, SDN-enabled secure IoT architecture, IEEE Int. Things J., 8 (2021), 6549–6564. doi: 10.1109/JIOT.2020.3043740.

[8] P. Mishra, A. Biswal, S. Garg, R. Lu, M. Tiwary, D. Puthal, et al., Software defined internet of things security: properties, state of the art, and future research, IEEE Wireless Commun., 27 (2020), 10–16. doi: 10.1109/MWC.001.1900318.

[9] F. O. Catak, A. F. Mustacoglu, Distributed denial of service attack detection using autoencoder and deep neural networks, J. Intelli. Fuzzy Syst., 37 (2019), 3969–3979. doi: 10.3233/JIFS-190159.

[10] F. Song, Y. Zhou, Y. Wang, T. Zhao, I. You, H. Zhang, Smart collaborative distribution for privacy enhancement in moving target defense, Inf. Sci., 479 (2019), 593–606. doi: 10.1016/j.ins.2018.06.002.

[11] De Donno M, Dragoni N, Giaretta A, Spognardi A. DDoS-capable IoT malwares: comparative analysis and mirai investigation. Secur Commun Netw 2018;2018: 7178164. https://doi.org/10.1155/2018/7178164.

[12] Cviti´c I, Perakovi´c D, Periˇsa M, Husnjak S. An overview of distributed denial of service traffic detection approaches. Promet Traffic Transp 2019;31(4):453–64.

[13] Chandola V, Banerjee A, Kumar V. Anomaly detection: a survey. ACM Comput Surv (CSUR) 2009;41(3):1–58.

[14] S. M. Mousavi, M. St-Hilaire, Early detection of DDoS attacks against SDN controllers, in 2015 International Conference on Computing, Networking and Communications (ICNC), (2015), 77–81. doi: 10.1109/ICCNC.2015.7069319.

[15] N. Dayal, P. Maity, S. Srivastava, Z. Khondoker, Research trends in security and DDoS in SDN, Secur. Commun. Networks, 9 (2016), 6386–6411. doi: 10.1002/sec.1759.

[16] M. Nobakht, V. Sivaraman, R. Boreli, A host-based intrusion detection and mitigation framework for smart home IoT using OpenFlow, in 2016 11th International Conference on Availability, Reliability and Security (ARES), (2016), 147–156. doi: 10.1109/ARES.2016.64.

[17] Perakovi´c D, Periˇsa M, Cviti´c I, Husnjak S. Model for detection and classification of DDoS traffic based on artificial neural network. Telfor J 2017;9(1):26–31.

[18] Chen W, Yeung DY. Defending against TCP SYN flooding attacks under different types of IP spoofing. In: Proceedings of the international conference on networking, international conference on systems and international conference on mobile communications and learning technologies (ICNICONSMCL'06). IEEE; 2006. pp. 38-38.

[19] P. Xiao, W. Y. Qu, H. Qi, Z. Y. Li, Detecting DDoS attacks against data center with correlation analysis, Comput. Commun., 67 (2015). doi: 10.1016/j.comcom.2015.06.012.

[20] V. Phan, N. Bao, M. Park, A novel hybrid flow-based handler with DDoS attacks in software-defined networking, in 2016 Intl IEEE Conferences on Ubiquitous Intelligence Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People, and Smart World Congress (UIC/ATC/ScalCom/CBDCom/IoP/SmartWorld), (2016), 350–357. doi: 10.1109/UIC-ATCScalCom-CBDCom-IoP-SmartWorld.2016.0069.

[21] F. Song, M. Zhu, Y. Zhou, I. You, H. Zhang, Smart collaborative tracking for ubiquitous power IoT in edge-cloud interplay domain, IEEE Int. Things J., 7 (2020), 6046–6055. doi: 10.1109/JIOT.2019.2958097.

[22] Bhushan K, Gupta BB. Distributed denial of service (DDoS) attack mitigation in software defined network (SDN)-based cloud computing environment. J Ambient Intell Humaniz Comput 2019;10(5):1985–97.

[23] S. Selvakumar, M. Mohanapriya,"Securing Cloud Data in Transit using Data Masking Technique in Cloud Enabled Multi Tenant Software Service". Indian Journal of Science and Technology, Vol 9(20), DOI: 10.17485/ijst/2016/v9i20/89782, May 2016.