International Journal of Engineering Research and Technology. ISSN 0974-3154, Volume 12, Number 1 (2019), pp. 124-130

An Efficient Routing and Distributed Denial of Service (DDOS) Attack Detection Technique for Centralized Control Plane

Mathiyalagan R[1], Saravana Balaji B[2], Josephine P[3]
[1] *Research Scholar, Visvesvaraya Technological University, Bangalore, India.*
[2] *Assistant Professor, Lebanese French University, Erbil, Iraq.*
[3] *Professor, East Point College of Engg & Tech, Bangalore, India.*

Abstract:

Hybrid IP networks that utilizes both control paradigms such as distributed and centralized. It combines programmability of Software Defined Network (SDN), and fault tolerance of distributed routing protocols, like OSPF. Naturally, a hybrid network arranges SDN to manage prioritized traffic and OSPF to promise relaxed process of finest traffic attempt. Distributed Denial of Services (DDoS) attacks are one of well-known and risky intimidation to the current network. The first threat attacks on control plane communications which exploit from lack of trust guarantees between controllers and switches. The previous system designed a Support Vector Machine (SVM) based classification scheme for attack detection. However it does produced satisfactory results. To overcome this difficulty the proposed work designed a Modified Adaptive Neuro Fuzzy Inference System (MANFIS) based DDOS attack detection scheme. In this work, the routing modification is done by using link-state advertisements (LSAs) generation. Then network partitioning is done with the help of Anarchic Society Optimization (ASO) algorithm. Compute fitness function by using utilization cost of sub graphs. In order to improve the security on control plane the detection of DDoS attacks is an important research topic. The investigate results produced that the designed system attains enhanced performance compared with the previous system in terms of accuracy, precision and recall.

Keywords: Anarchic Society Optimization (ASO), Distributed Denial of Services (DDoS), SDN Partitioning (SDNp), Software Defined Network (SDN).

I. INTRODUCTION

Software Defined Networking (SDN) is a rising system design that attracts individuals' in recent days. In this novel system, its control plane and data plane are isolated. The SDN controller is a legitimately centralized control plane and obtains the responsibility of the flows in the SDN organize [1]. It can adaptable control the flows in the system and allocate discretionary flows to the outgoing links of the SDN nodes [2-3]. A few systems can be abused to guide the distinctive flows to a similar destination to various next hops and the traffic is part on the network layer. The organization of SDN in the system gives a helpful and compelling approach to do the traffic engineering and can enhance the system execution on an expansive scale. Microsoft outlines programming driven WAN interrelate the data centers and accomplishes a huge use of the system throughput [4], which is close ideal. Google additionally use SDN and runs numerous connections at the usage of almost 100% [5]. The upsides of SDN give a motivating force to the change of SDN network. However, completely organizing the SDN in the network is simple employment. It might experience prudent, various leveled and specialized challenges [6]. The SDN has its own particular difficulties of full organization, which makes the full arrangement of SDN troublesome for the time being. Therefore now a day's fractional SDN deployment, i.e, SDN/OSPF hybrid network is introduced.

The word hybrid control plane defines to an inexorably significant network planning, Software-Defined Networking (SDN) and Open Shortest Path First (OSPF) or Intermediate System to Intermediate System (ISIS)– are implemented in the similar routing domain. Open Shortest Path First (OSPF) is a best one among the most used Interior Gateway Protocols (IGPs) lately [7]. In OSPF, each connection is allocated a weight (or a cost) and the most brief ways among the sources and destination are processed as far as these weighted associations inside an Autonomous System (AS). The traffic facilitated to a similar destination at a route node and is consistently separate the following jumps on the comparable cost most brief ways. In the general OSPF organize, each node support OSPF routing protocol. Each router keeps up a sending table and reliably picks the nearest next bounce to forward packets. To acquire high system execution in traffic, need to enhance the OSPF weights to adjust the packet flows. Much work has been done already on hunting down the propelled weight setting under OSPF protocol.

With the reliable advancement of network topology, the interminable extension of network business needs, and quick development of the Internet economy in the Internet age, the services of network with imperative business and industry data have been spread to the creation and life of current society [8]. The appearance of DDoS can direct to variations in the related network services, causing enormous financial misfortunes and even causing causing other cataclysmic results [9]. DDoS attacks are one of the serious network security threats facing the Internet. It is a key research point in the security field to identify DDoS attacks precisely and rapidly. Offering protection to control plane using conventional Intrusion Detection System (IDS) to alleviate DoS attack is an important issue. The SDN performance and behavior of network are affected due to DoS attack. Reasonable receivers are not capable to converse with server after formation of DoS attack [10].

SDN has a perpendicularly distributed control plane. Numeral network/sending gadgets on each layer increments as per the Fibonacci arrangement as the thought remember that arrangement improves like undergrowth of a tree (spanning tree) with no loop. They deals with the network by utilizing Fibonacci heap requested tree for balancing load and routing. The designed system is resolvable in polynomial time and produces minimum reaction time when contrasted with the conventional network. In SDN, distributed controllers are introduced to resolve the scalability troubles and reliability problems of network control plane. There is a restriction of distributed controllers, the mapping of switch and controller is arranged statically because of which the load distributed among different controllers isn't even. To take care of this issue, this system is proposed in which the pool of controllers is shrink and grow vigorously according to the traffic on the link and load on each connection is powerfully moved over all controllers.

II. LITERATURE SURVEY

In the ongoing circumstances, a prior research work managed the reasonableness of a finger printing assault prepared for a SDN network Shin and Gu (2013). In this assault, first the designed system is selected to choose if it exploits SDN/Open Flow switches. At that point the SDN is submitted to a DoS attack on the control plane through the data control plane correspondence way and over the data plane through the network device's flow table. A DoS attack point out a demonstration done to make the inaccessibility of a machine or system resources to its designated users. In SDN, the system devices require access to the control plane so as to get traffic control guidelines and the traffic over the system needs access to the network device tables with a specific end goal to determine the traffic administration strategies. Along these lines, the data control plane interface and the network device flow table turn into the purposes of susceptibility for the DoS attack [11].

Open Flow Random Host Mutation (OF-RHM) Jafarian et al., (2012) familiar another answer for avoid the address location. An Open Flow controller is used for the organization of a pool containing virtual IP conveys that are relegated to have inside the system, concealing the honest to goodness IP addresses from the outside world and exhibiting convenient target security. The genuine IP address stays unaltered naturally while the remotely - distinguishable virtual IP addresses are changed in unpredictable at predictable period. The stream table size key for managing OF-RHM is extended as the system session establishment rate and session span expands that, in this manner, could be face for conveying OF-RHM in a colossal system [12].

Liu et al., (2016) outlined a two layer Open Flow switch topology that gets the challenges of stream table size in a different switch, the versatile nature of building security plots in these switches, and a short time later the heap harmony among these switches into thought. Also, it gives shielded plans to evaluating the conformance of these switches in an arrangement for a predominant load modify amid an adjustment in the traffic movement. In correct, the advise technique is shown as a track in an outline, where every node stays for a security framework satisfied arrangement, and each edge symbolizes an unmistakable advance of wellbeing educate. In perspective of this model, a heuristic method for finding ideal refresh track in the time delivered is delineated. Reproductions achieved of the inform configuration show that the as of late present technique is honest to goodness and tried and true under a different extent of conditions. As saw from the creators, prosperity controllers are fundamentally grateful for mechanizing the security plot in huge scale frameworks [13].

Garcia-Alfaro et al., (2011) gave the fundamental handiness of MIRAGE, which is an organization instrument for the examination and sending of setup methodologies over framework security segments, like firewalls, intrusion recognizable proof systems, and VPN routers. Two fundamental functionalities that are consolidated into the present model are investigated: they are (1) a base up appraisal of formally executed framework security strategies and (2) a top down refinement of overall methodologies into sort out security part setups. In both of these cases, MIRAGE offers intra-fragment examination (to recognize the deficiency in single part executions) and between segment examination (to perceive multi- - segment utilization, which have no consistency). MIRAGE also controls the assurance of the security plan topology to ensure the correct execution of every strategy [14].

III. PROPOSED SYSTEM

This part mainly focus on the hypothetical background, time difficulty issue, and methods of the two basic structure blocks in network formulation such as routing updating via LSA creation and network partitioning. SDNp permit advertising topology data modified for each sub-domain, as shown in fig.1: the new topology and its partitioning is shown in fig.1a is not showing in the direction of the nodes in subordinate-domain k (i.e., nodes 1, 2, and 3). As an alternative, the modified analysis give in the direction of these nodes pretends with the purpose of together border nodes a and b have straight links to each and every one other nodes (like shown in Sub fig.1b). This manner, the exit node for every inter-sub-area flow is able to be calculated on a per-destination origin basically with condition the OSPF link weights of the effective associations [15]. For example, setting x and y computes how the traffic in the favour of d presents sub area k. Please consider with the purpose of the OSPF nodes 1-3 are not responsive of the fact with the purpose of they form a sub-area, and consider with the purpose of the limit SDN nodes a and b are usual OSPF neighbors.

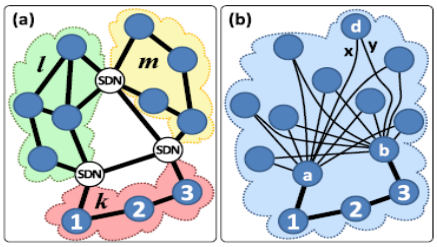

Figure. 1. Two views of the same network: (a) the actual topology and the partitioning of the network, and (b) how it's represented to the nodes in sub-domain *k*

The self-expression of routing in SDN-partitioned networks is controlled by the constraint with the purpose of the custom of routing paths should be steady by means of the presented link metrics. Additional accurately, the routes of each and every one flows, which begins at the OSPF nodes $r_1 \dots r_\alpha$ of the similar sub-area and with the purpose of are destined designed for the similar sub-area-external destination d, based on the metrics $m(b_1, d) \dots m(b_\beta, d)$ in the set of LSAs with the purpose of has been broadcasted for d via the use of border (SDN) routers $b_1 \dots b_\beta$ of with the purpose of sub-area. From the fig.1 it shows with the purposed of traffic flows, which go into the network via a node of sub-area k.

Uniqueness: let us consider that a SDN network which consists of N set of nodes with 'R' set of OSPF nodes and B a set of border nodes by means of $N = R \cup B$. Then the network is divided into K sub-areas is represented as K-partitioned, and the sub-areas are ordered and denoted as $1, 2, \dots K$ in a single manner. Moreover, every node $n \in N$ has a SDN broad unique identifier. Every node r ∈ R goes to a particular sub-area and it is a subset R_k of OSPF nodes designed for every sub-area k. The $|R_k| = \alpha$ nodes in R_k are denoted as $R = (r_1 \dots r_\alpha)$ in a unique manner. Every border nodes in the SDN is denoted as b∈B in the direction of many sub-areas, however for every sub-area k, the direct of its β border nodes $Bk = (b_1 \dots b_\beta)$ is single as well. Moreover, we represent $N_k = R_k \cup B_k$ as the group of each and every one nodes of sub-area k and $\overline{N_k}$ =N\$N_k$as the set of each and every one nodes external in the direction of sub-area k.

Distance: Every link (n_1, n_2) is allocated an integer link metric rate m(n_1, n_2) and m_k(n_1, n_2) represents a dynamic metric (i.e., one with the purpose of be able to be changed with the central organizer) which is promoted in sub-area k. The distance δ(r1, b) metric is described as the sum of each and every one link metrics m(r_1, r_2) + m(r_2, r_3) . . . + m(r_n, b) the length of the n hop least cost path among an OSPF node 'r_1' and an SDN border node 'b' of the similar sub-area. Each and every one distance exclusively based on the first configuration of link metrics on OSPF nodes, and we let us consider with the purpose of the formation of those link metrics is static.

Metric vector: A group of link metrics with the purpose of be able to be broadcasted via the use of the β border nodes of B_k designed for a destination d ∈ $\overline{N_k}$ is represented as metric vector $\vec{m}$= (m(b_1, d, k),m(b_2, d, k), . . . m(b_β, d, k)). The fraction of a metric vector are prepared related to the regulating of the particular border nodes in that sub-area and the i[th] part m(b_i, d, k) is represented as $\vec{m}_i$. A metric vector is optimal, if the commercial of its link metrics lead to a non confusing one path routing algorithm. Two metric vectors $\vec{m} \equiv \vec{m}'$ are equal if they result in the similar routing.

Exit vector: let us consider that the distance metric from node r via the use of border node b to destination d by means of link metrics $\vec{m}$ broadcasted for destination d as δ(r, b, d, $\vec{m}$). The way out border node e(r, d, $\vec{m}$) ∈ B_k for packets from a node r ∈ R_k in the direction of a destination d ∈ N_k be able to now be computed by the used metric vector $\vec{m}$ with the purpose of was broadcasted for d in sub area k as follows

Quantity vector: The subset Rk(b, $\vec{m}$) ∈ R_k consists of each and every one domain-internal nodes of sub-domain k with the purpose of make use of border node b as way out in case $\vec{m}$ ∈ Mk was broadcasted in sub-domain k. If e(k, $\vec{m}$) is known, we be able to establish how numerous OSPF nodes of R_k use a specific b ∈ B_k as way out.

III.I LSA Generation Algorithm

The injectivity of $E_k \rightarrow Q_k$ has thoughtful focus on the complexity of proposed algorithm with the purpose of creates M_k in network, as the metric vectors are able to be recognized depending on quantity vectors before exit vectors. The investigate space of each and every one metric vectors spans $O(\beta^\alpha)$ (with $|R_k| = \alpha$ and $|B_k| = \beta$), while the investigate space of each and every one quantity vectors only spans $O(\binom{\beta+\alpha-1}{\alpha})$. The algorithm we introduced in the direction of find out each and every one metric vectors be able to be thought of as a tally counter, which constantly advances the part by means of the smallest number of index *i* (i.e., the metric with the minimum number of index *i* in $\vec{m}$) a particular step (i.e., till the point where one more OSPF node stops by means of the according border node b_i as exit), until with the purpose of part has reached the end (i.e., with the purpose of metric is so large with the purpose of *bi* is no longer the exit node designed for some OSPF node). In with the intention of event, with the purpose of element is turned back in the direction of zero and the following part is advanced, and shortly. The operational process of the designed scheme is explained as follows: The first metric vector $\vec{m}_0 = (m_1, \dots, m_\beta) \in M_k$ has values $m_1 = 0$ and $m_x = (x - 1) . (1 + \Delta_k^{\max})$ designed for each and every one $2 \le x \le \beta$ with $\Delta_k^{\max}$ as the maximum difference among some two metric distances in sub-area *k*:

$$\Delta_k^{\max} = \max\{\delta(r, b) - \delta(r', b') | r, r' \in R_k, b, b' \in B_k \quad (1)$$

The quantity vector designed for this metric vector is $\vec{q}$= (α, 0, . . . , 0), as all $|R_k| = \alpha$ OSPF nodes in R_k determination make use of the initial border node as exit. (Let us consider that the notation: the *n*th part of $\vec{a}$ is denoted $\overrightarrow{a_n}$ and the *i*[th] part of a group of vectors *A* is represented as $\vec{a}_i$.) The residual metric vectors are created continuously, such with the purpose of the *z*[th] metric vector $\overrightarrow{mz}$ is a replica of the *y*[th] metric vector $\overrightarrow{my}$ (where $z = y-1$) with its *i*[th] part (i.e., metric) improved by value *v* as discussed in Algorithm 1.

Amongst each and every one representative metric vector (M_k), this proposed method creates moreover a number of equivalents and a small number of non-valid ones, which have to be removed after the algorithm ends.

Algorithm 1: Generation of a Metric Vector

Input : R_k, B_k, $\delta(r,b)$, $\overrightarrow{my}$
Output: $\overrightarrow{mz}$
Step 1: Find out the component index i
For n:=β to 0 do
If $\overrightarrow{my}_n > o$ *then*
i:=n:
end
if i=β then
end algorithm
end
step 2: Find out value v
v:= $\Delta_k^{\max}$
for n:=i+1 to β *do*
for q:=to $\propto$ *do*
if v> $\delta(r_q, b_n)+\overrightarrow{my}_n$ *then*
if v:= $\delta(r_q, b_n)+\overrightarrow{my}_n$ *t+1*
end
end
end
Step 3: Create $\overrightarrow{mz}$
$\overrightarrow{mz}$:= $\overrightarrow{my}$:
$\overrightarrow{mz_i}$:= $\overrightarrow{mz}_i$+v
For n:=i-1 to 0 do
$\overrightarrow{mz_n}$:=0
End

Modified Adaptive Neuro Fuzzy Inference System (ANFIS)

The ANFIS network is one of the types of neural network and it is performed based on the neuro fuzzy network [6]. Because the ANFIS is an adaptive network, element of its nodes are adaptive, which means with the purpose of their outputs based on the parameters fit in to these nodes. The representation of ANFIS is illustrated in fig.2, and the node function in every layer is defined as below.

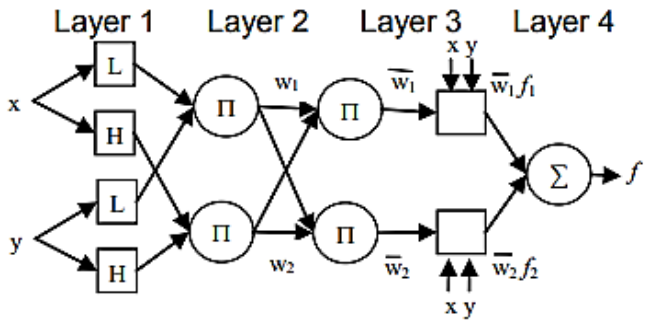

Figure 2: Architecture of Adaptive Neuro Fuzzy Inference System (ANFIS)

Layer 1. This layer is the layer of membership functions with the purpose of consists of adaptive nodes with node functions defined as follows,

$$O_i^1 = \mu_{A_i}(x) \; for \; i = 1,2 \tag{2}$$

Where x and y are denoted as the input nodes, A and B are denoted as the linguistic labels, (x) and (y) are denoted as the membership functions which frequently assume a bell shape by means of the highest and lowest values equivalent to 1 and 0, correspondingly:

$$\mu(x) = \frac{1}{1+(\frac{x-c_i}{a_i})^{2b_i}} \tag{3}$$

Where, a_i , b_i, and c_i are the premise parameters set.

Layer 2: This layer consist of the nodes labeled $\prod$ which multiply incoming data and gives the product out. For instance,

$$O_i^2 = w_i = \mu_{A_i}(x)\mu_{B_i}(y), \; i = 1,2 \tag{4}$$

The output $w_i$denotes the firing strength of a rule. The results of every node denotes the firing strength of a rule

Layer 3: In this layer, the nodes labeled N find the ratio of the ith rule's firing power towards the sum of the entire rules' notice strengths,

$$O_i^3 = w_i = \frac{w_1}{w_1+w_2}, \; i = 1,2 \tag{5}$$

The results of this layer are named the normalized firing strengths.

Layer 4: This layer's nodes are adaptive by means of the subsequent node functions,

$$O_i^4 = w_i f_i = w_i(p_i x + q_i y + r_i) \tag{6}$$

Where w denotes the output of layer 3, and $\{p_i, q_i, r_i\}$ is denoted as the parameter set. These parameters are known as the resultant parameters.

Layer 5: This layer's particular fixed node, labeled $\sum$, works out the final results as the outline of the complete incoming data which is described as follows,

$$O_i^5 = \sum_{i=1} w_i f_i = \frac{\sum_i w_i f_i}{\sum_i w_i} \tag{7}$$

Consequently, an adaptive network with the purpose is functionally related to a Sugeno first-order fuzzy inference system is generated. The ANFIS is optimized by changing the antecedent parameters and consequent parameters consequently with the purpose of a particular objective function are minimized. In order to decrease error and increase results, there is a require of effective training of ANFIS [10]. In this proposed work, Ant Colony Algorithm (ACO) is proposed to optimize the parameters of ANFIS designed for the intention of finding the global optimal solution.

Usually the ants select the path depending on the pheromone updating on it. The pheromone updating on a path is associated with the number of the ants transitory all the way through the path, it creates the system straightforward to hold into local optimum. In order to solve this problem and increase the convergence speed of the ACO the proposed system, new pheromone updating is carryout. At each iteration, the ants which have established the shortest path in the iteration are labeled as selected ants. The path selection step of ACO is determined by general ants. For the *k*-th examining ant the probability P_{ij}^{k} (t) of iteration *t* to shift from *i*-th grid to *j*-th grid is represented via the use of the following equation.

$$P_{ij}^{k}(t) = \left\{ \begin{array}{c} \frac{\eta_{ij}^{\beta}(t)}{\tau_{ij}^{\alpha}(t)} \\ \hline \sum_{s \in allowed_k} \frac{\eta_{is}^{\beta}(t)}{\tau_{is}^{\alpha}(t)} \end{array} , j \in allowed_k \right\} \quad (8)$$
$$0, otherwise$$

Where,

$allowed_k$-is denoted as the set of the left behind possible states of the *k*-th investigative ant

$\alpha, \beta -$ is denoted as the Weight of $\tau_{ij}(t)$ and η_{ij} (t) on the transition probability, correspondingly

η_{ij} *is denoted as the* Local heuristic value of visibility , it is represented by the following equation

$$\eta_{ij}(t) = \frac{1}{d_{ij}} \quad (9)$$

The elite ant's pheromone update step is described as

$$\tau_{ij}(t+n) = (1-\rho) * \tau_{ij}(t) + \Delta\tau_{ij}(t) + \alpha + \Delta\tau_{ij}^{best}(t) \quad (10)$$

Where,

$\Delta\tau_{ij}(t)$- is denoted as the pheromone increment of usual ants

$\Delta\tau_{ij}^{best}(t)$ is denoted as the best pheromone growth of elite ants which open on the path<*i*, *j*> in the current iteration

$$\Delta\tau_{ij}^{best}(t) = \left\{ \begin{array}{c} \frac{Q}{L_{best}}, i,j \in L_{best} \\ 0, otherwise \end{array} \right\} \quad (11)$$

Where, Q is denoted as the constant which described the sum of the pheromone increases

$L_{best} -$ is denoted as the best length of the path with the purpose of the *k*-the lite ant has accepted in the present iteration.

In this work the objective function or fitness is computed via the use of root mean squared error (RMSE) among actual results and predictive results,

The suitable classifier is chosen in order to create the detection algorithm based on the series of feature values of samples and the unlabeled feature values of samples are categorized by using the classifier.

IV. EXPERIMENTAL RESULTS

The proposed DDOS attack detection mechanisms are executed in the Network Simulator tool version 2(NS2). The proposed Modified ANFIS based attack detection and existing Support Vector Machine (SVM) based attack detection methods compared in terms of accuracy, precision and recall. In this proposed system, the normal traffic is generated during the training sample phase; It includes TCP traffic, UDP traffic, and ICMP traffic.

Accuracy (A)

Accuracy (A) is the proportion of correctly classified packets as ordinary and attackers in the packet list.

A= (Accurately classified packets /total packets)*100 (12)

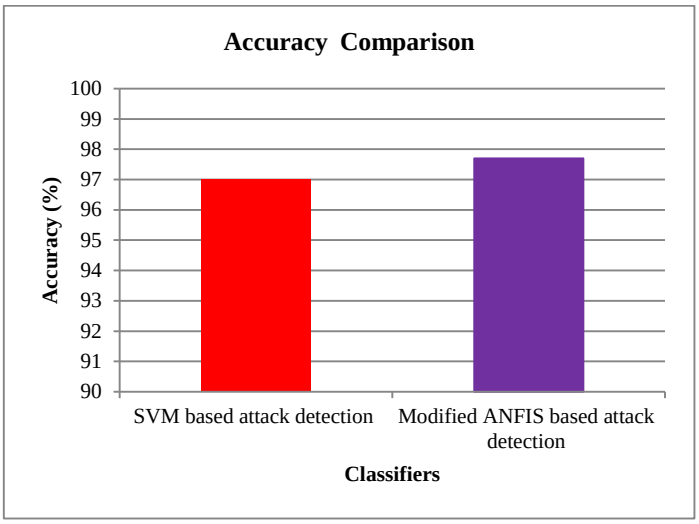

Figure 3: Accuracy comparison

The accuracy of the proposed Modified ANFIS based attack detection and existing SVM based attack detection methods are shown in fig.3. In x-axis attack detection methods are taken and accuracy is taken as y-axis. Based on the outcomes, it proves that the presented Modified ANFIS classifier provides greater accuracy outcomes of 97.7%, while the SVM based attack detection methods yields recall outcomes of 97%. It proves that the presented Modified ANFIS classifier achieves better accuracy compared to existing methods.

Precision (P)

It is known as the quantity of accurately predictable records over the whole predicted records for a particular class. By using the confusion matrix, M, precision for each class, j, could be explained in this form:

$$P_j = \frac{TP_j}{TP_j + FP_j} \times 100 \quad (13)$$

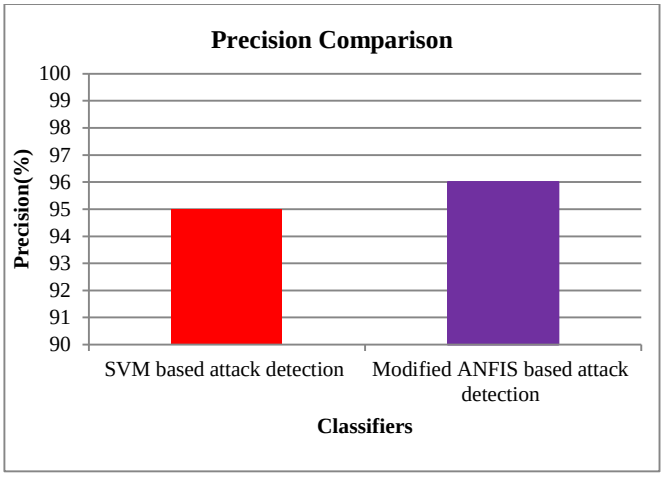

Figure 4: Precision comparison

Fig.4 displays the precision outcomes evaluation of DDOS based attack detection techniques such as SVM based attack detection and Modified ANFIS based attack detection. In x-axis attack detection methods are taken and precision is taken as y-axis. According to the outcomes, it proves that the presented Modified ANFIS classifier gives greater precision outcomes of 96%, while the SVM classifier yields precision outcomes of 95%. It proves that the presented Modified ANFIS classifier contains greater detection of attacker compared to other approaches.

Recall

The quantity of precisely predicted packets over all the packets attained for a specific class. Utilizing the confusion matrix,, M, recall for each class, j, can be referred as follows:

$$R_j = \frac{TP_j}{TP_j + FN_j} \times 100 \qquad (14)$$

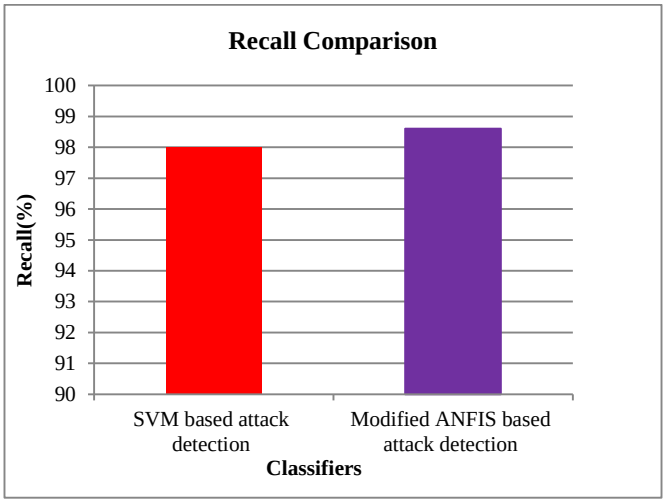

Figure 5: Recall comparison

Fig.5 displays the recall of proposed Modified ANFIS based attack detection and existing SVM based attack detection methods. In x-axis attack detection methods are taken and recall is taken as y-axis. According to the outcomes, it proves that the presented Modified ANFIS classifier gives greater recall outcomes of 98.6%, while the SVM based attack detection methods yields recall outcomes of 98%. It proves that the presented Modified ANFIS classifier achieves better recall compared to existing methods.

V. CONCLUSION

In this proposed work, hybrid SDN/OSPF control plane launch the centralized control over the distributed routing protocol by splitting the topology into sub-domains with SDN-enabled border nodes. The Link-State Advertisements (LSAs) generation based routing adaptation is performed to achieve an efficient packet transmission. The Anarchic Society Optimization (ASO) algorithm is used to compute network partitioning with the help of utilization cost of sub graphs. In order to perform DDoS attack detection in control plane, Modified Adaptive Neuro Fuzzy Inference System (ANFIS) algorithm is utilized. From the results show that the designed system produced better performance matched with the previous system in terms of accuracy, precision and recall.

REFERENCES

[1] Guo, Y., Wang, Z., Yin, X., Shi, X., Wu, J. and Zhang, H., 2015, December. Incremental deployment for traffic engineering in hybrid SDN network. In Computing and Communications Conference (IPCCC), 2015 IEEE 34th International Performance (pp. 1-8). IEEE.

[2] Vissicchio, S., Vanbever, L. and Bonaventure, O., 2014. Opportunities and research challenges of hybrid software defined networks. ACM SIGCOMM Computer Communication Review, 44(2), pp.70-75.

[3] Hong, D.K., Ma, Y., Banerjee, S. and Mao, Z.M., 2016, March. Incremental deployment of SDN in hybrid enterprise and ISP networks. In Proceedings of the Symposium on SDN Research (p. 1). ACM.

[4] Y. Jimenez, C. Cervello-Pastor, A. Garcia, On the controller placement for designing a distributed sdn control layer, in: Networking Conference, 2014 IFIP, 2014, pp. 1–9. doi:10. 1109/IFIPNetworking.2014.6857117

[5] M. Caria, A. Jukan, M. Hoffmann, "A Performance Study of Network Migration to SDN-enabled Traffic Engineering," Globecom 2013, Atlanta, USA, December 2013

[6] Vissicchio, L. Vanbever, O. Bonaventure, "Opportunities and Research Challenges of Hybrid Software Defined Networks," ACM SIGCOMM CCR 44(2), pp.70-75, April 2014

[7] Yan, Q., Yu, F.R., Gong, Q. and Li, J., 2016. Software-defined networking (SDN) and distributed denial of service (DDoS) attacks in cloud computing environments: A survey, some research issues, and

challenges. IEEE Communications Surveys & Tutorials, 18(1), pp.602-622.

[8] Kandoi, R. and Antikainen, M., 2015, May. Denial-of-service attacks in OpenFlow SDN networks. In Integrated Network Management (IM), 2015 IFIP/IEEE International Symposium on (pp. 1322-1326). IEEE.

[9] Bawany, N.Z., Shamsi, J.A. and Salah, K., 2017. DDoS attack detection and mitigation using SDN: methods, practices, and solutions. Arabian Journal for Science and Engineering, 42(2), pp.425-441.

[10] Schuchard, M., Mohaisen, A., Foo Kune, D., Hopper, N., Kim, Y. and Vasserman, E.Y., 2010, October. Losing control of the internet: using the data plane to attack the control plane. In Proceedings of the 17th ACM conference on Computer and communications security (pp. 726-728). ACM.

[11] Shin, S and Gu, G, (2013). Attacking software-defined networks: A first feasibility study. In Proceedings of the second ACM SIGCOMM workshop on Hot topics in software defined networking, pp. 165-166.

[12] Jafarian, J.H, Al-Shaer, E and Duan, Q, (2012). Open flow random host mutation: transparent moving target defense using software defined networking. In Proceedings of the first workshop on Hot topics in software defined networks, pp. 127-132.

[13] Liu, J, Li, Y, Wang, H, Jin, D, Su, L, Zeng, L and Vasilakos, T, (2016). Leveraging software-defined networking for security policy enforcement. Information Sciences, vol. 327, pp. 288-299.

[14] Garcia-Alfaro, J, Cuppens, F, Cuppens-Boulahia, N and Preda, S, (2011). MIRAGE: a management tool for the analysis and deployment of network security policies. Data Privacy Management and Autonomous Spontaneous Security, pp. 203-215.

[15] Guo, Y., Wang, Z., Yin, X., Shi, X. and Wu, J., 2014, October. Traffic engineering in SDN/OSPF hybrid network. In Network Protocols (ICNP), 2014 IEEE 22nd International Conference on (pp. 563-568). IEEE.