Abstract

Distributed Denial of Service (DDoS) attacks are common these days [1]. So it is evident that current industry solutions such as completely relying on Internet Service Provider(ISP) or setting up own DDoS infrastructure are not sufficient in detecting and mitigating DDoS attacks, hence consistent research is needed. Most of the current industry solutions involve setting up centralized expensive hardware system which can analyze the network data packets[1] [2] for probable DDoS attacks. Also each organization or ISP has different systems which are not compatible with other ISP's. In this paper we are going to discuss another a way to detect and mitigate DDoS attack using machine learning tools.

[1] We would call it 'packet'or 'packets'in rest of the paper

Chapter 1

Existing Systems

Distributed Denial of Service (DDoS) attack is the way to jam host network or its resources with large number of data packets, so that host become disabled to serve. The real challenge in detecting and defending DDoS attack is because of its dynamic nature. The source[1] is not a single node or system on the Internet but many, and often distributed over the Internet. Also the source of the packet is often spoofed, which makes more harder to know the actual Internet address of system from where attack is originated because they hide original attack source. On top of that, many times the source itself is not aware that it is compromised and has been used as bot [3] by attacker.

Detecting and mitigating attack at the destination is not very useful as because destination may know that the attack is happening but to stop it happening it will have to block all the incoming traffic including the legitimate traffic, because source address can not be reliable way to know the attack source. To avoid this, firm which provide the networking devices have come up with solutions.

Many of the solutions available in the market or the research that was done is to collect network traffic flow [4] (we would call it just flow) samples at routers(gateways) and feed/send it on the central system to analyze. Central system is a hardware and software infrastructure which is capable of processing and analyzing large flow information.

Some of the major protocols which are widely used for flow collection and analysis are, Internet Protocol Flow Information Export (IPFIX) protocol created by Internet Engineering Task Force (IETF), Ciscos NetFlow [5] and Sflow(Sampled flow) [6]. These protocols have defined standard way to export flow information from router and similar devices. All these flow monitoring protocols gather information and send the consolidated flow information to the centralized server where user can login and do analyses for different purpose such as Security Monitoring,

[1] It is a system/device on the Internet which has an IP address and which is involved in DDoS attack

Bandwidth monitoring, Resource Management, Traffic Analysis, Performance Management. It will have some modules which are specifically used for anomaly/DDoS detection.

E.g. Cisco netflow has flow Exporter, Collector and Analysis modules. Flow exporter is generally router who send flow information to collector module which acts as flow storage. Analysis module then try to find out different patterns in the flow.

These technologies scales well and sufficient to indicate trends in network traffic but they have limitations. 1. They are not cross platform, e.g. Sflow router would not be able to work with Cisco routers. 2. They involve setting up expensive hardware which would act as collector server. 3. In these technologies source address is used for flow analysis but as we know it is not reliable due to IP spoofing in in case of of DDoS attacks.

now we know that router based flow analysis would be useful for anomaly detection but we don't want to set up expensive hardware, we want to have protocol or system which is not compatible with other routers also we want to only rely on destination IP address for flow[2] analysis. So if we could come up with the way by which we could detect anomaly in the traffic at the routers independent of manufacturer, and create a protocol to communicate the attack parameters to allow routers to take decisions then we would be more efficient in detecting and mitigating DDoS attacks.

If we use the learning algorithm at the router and if router could learn the normal behavior of the flow then if there is any anomaly then it will be able to identify it from its previous learning and that change in behavior of the traffic can be communicate to destination network.

With the advance of electronic and the Internet of things, electronic devices are getting equipped with faster processor and fast memories. Router are also not left behind. Only thing that is missing is the storage space. If we use the learning techniques which don't need much storage then we don't have to store the packets instead we would learn from every flow and discard once learning is done. This is necessary because number of entries on the Internet routing table has steadily grown. Now that the table has passed 500,000 routes [7] so storing each and every flow information for these routes could be difficult.

[2] In our paper we would consider flow only in sense of destination address

Chapter 2

DDoS Detection and Mitigation

One of the way to detect the DDoS attack is to check for anomalous behavior in the network traffic. This can be done at different points. Either at the destination server or at the router. Once the attack is detected by observing the behavior of the traffic to mitigate it we would have to block all the packets which are causing the attack. One of the way to identify those packets is by their source address.

Chapter 3

Network Functioning

A switch creates a network and router connects networks. A router links computers to the Internet through other routers. Routers are the backbone of the network who helps to forward packet from one point to other point on the Internet. Every packet traveling on Internet has to go through router [8].Router knows where the packet is destine hence it could serve as first point of knowledge about the change in the flow information for a destine network. Each router has interfaces to which hosts or other network are connected. So router is aware to whom it is connected. Router uses protocol to communicate and by that they gather knowledge about other networks or router on the Internet. ICMP [9] is one of the most frequently use protocol by routers to communicate.

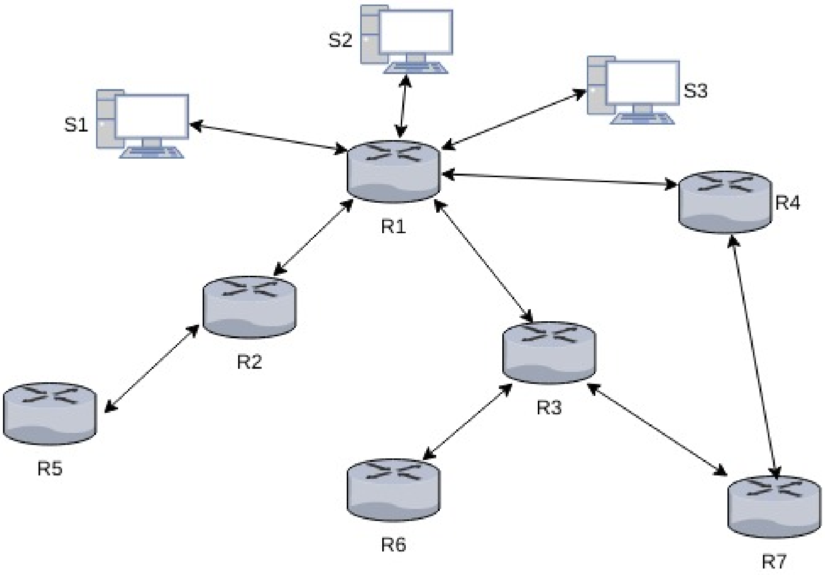
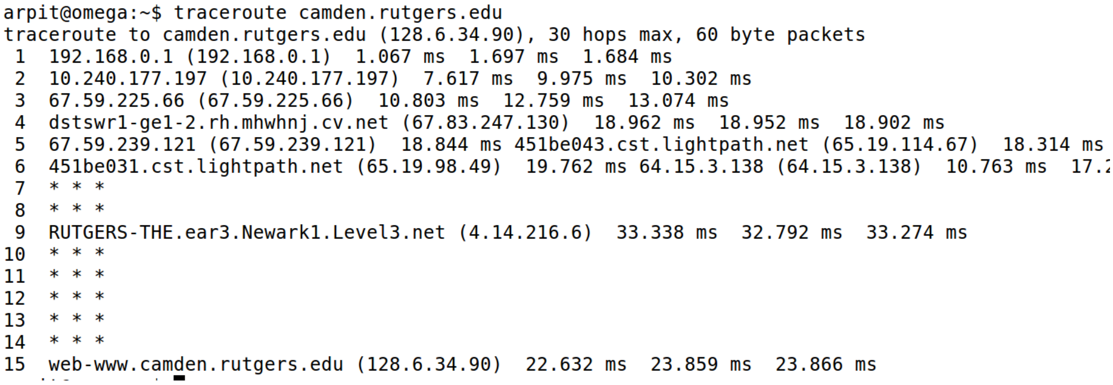


Figure 3.1: Network Example

Lets illustrated this using an example. In the above figure we can see that host S1, S2, S3 are connected to router R1. Router R1 is connected to Internet through router R2, R3, R4 , thus every packet that would be reaching to S2 would come from either of these three routers. All three routers are locates at different geographical region. Most of the websites are regional, either county, state or national (If we leave out few global websites) and hence they are mostly accessed from those region it is meant for. E.g. Rutgers University website would be accessed mostly from the eastern region of United States and that too mostly from the New Jersey State or the Philadelphia region.

Using traceroute we can find out how many hope away the destination is. Each hope is the router on the Internet. Following is one of the captured traceroute for Rutgers University website.



```
arpit@omega:~$ traceroute camden.rutgers.edu
traceroute to camden.rutgers.edu (128.6.34.90), 30 hops max, 60 byte packets
 1  192.168.0.1 (192.168.0.1)  1.067 ms  1.697 ms  1.684 ms
 2  10.240.177.197 (10.240.177.197)  7.617 ms  9.975 ms  10.302 ms
 3  67.59.225.66 (67.59.225.66)  10.803 ms  12.759 ms  13.074 ms
 4  dstswr1-ge1-2.rh.mhwhnj.cv.net (67.83.247.130)  18.962 ms  18.952 ms  18.902 ms
 5  67.59.239.121 (67.59.239.121)  18.844 ms 451be043.cst.lightpath.net (65.19.114.67)  18.314 ms
 6  451be031.cst.lightpath.net (65.19.98.49)  19.762 ms 64.15.3.138 (64.15.3.138)  10.763 ms  17.2
 7  * * *
 8  * * *
 9  RUTGERS-THE.ear3.Newark1.Level3.net (4.14.216.6)  33.338 ms  32.792 ms  33.274 ms
10  * * *
11  * * *
12  * * *
13  * * *
14  * * *
15  web-www.camden.rutgers.edu (128.6.34.90)  22.632 ms  23.859 ms  23.866 ms
```

Figure 3.2: Trace Route: All the routers in the path to destination

We can see that there are about 14 routers (if we don't consider the home router 192.168.0.1) to reach to the camden.rutgers.edu. This trace route is taken from a location in the New Jersey State.

Chapter 4

Suggested New Approach

From figure 3 and 3 we know that routers are located at different geographical locations and also we know that there is pattern in which the particular destination website is getting accessed from the different regions. Some of the service providers such as GeoIP or Google can find out the location from where the traffic is coming in the network for a given destination, but that is approximate based on the source IP, which in the case of DDoS attack is unreliable information because packet source address are often spoofed. So it could be difficult to know the actual geographical location from which packet has come. Only routers can provide the correct geographical information about the source of the packet.

In the normal scenario there would be some definite pattern in which the website is getting accessed from different regions and this pattern can be learned over the time with learning algorithms at the routers. Thus finding this pattern in the flow at the router can form the basis of our analysis. When ever there is deviation from the normal pattern of the flow for a particular destination then that change in pattern as well as the geographical region can be communicated to the destination network. Destination network on receiving that information can decide based on the region and how much is the deviation from the normal to decide whether it want the reporting router to discard or forward the traffic for the destination. This is selective process in which traffic from other routers remain unaffected

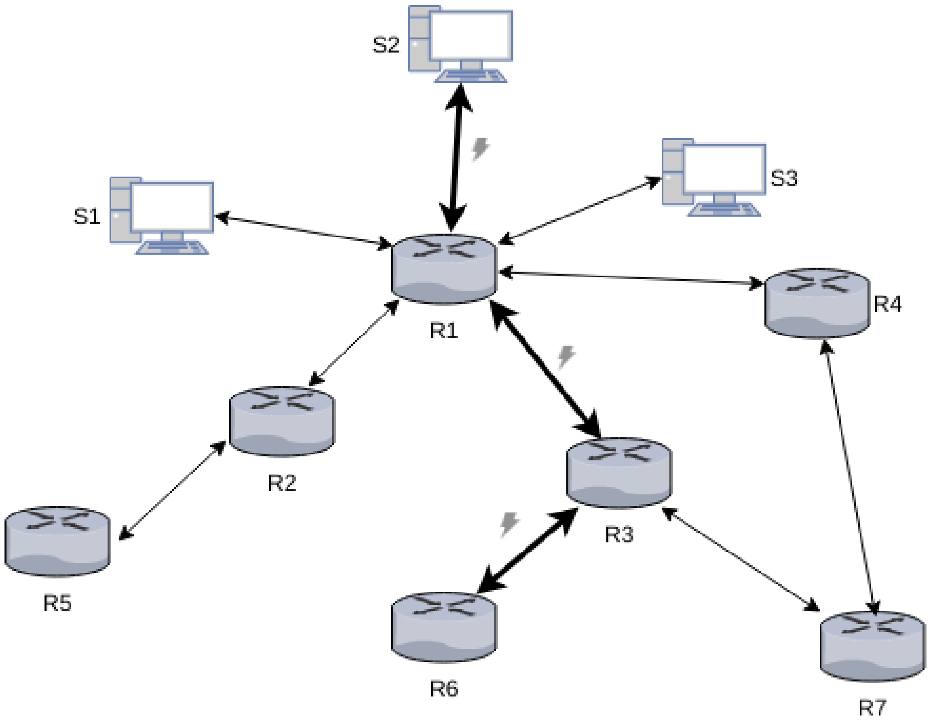


Figure 4.1: DDoS Attack path

In the above figure attack initiated from the region where router R6 is located and from R6 data packets reach to victim from R3 and R1. If we could detect attack at R6 itself then we can discard packets at R6 which are heading towards S2, while traffic from other routers remain unaffected.

To achieve this we would measure the flow during a time window (e.g 300 sec) whose size would be fixed at the beginning. These time windows can be combined to form a period which is a portion of a day during which traffic is measured. Once we have flow information we can apply learning techniques on each flow iteratively to gain deeper knowledge about normal behavior of the flow.

In the proposed system, each router will itself act as a analyzer. Each packet will be analyzed and flow statistic would be created based on the destination IP address. Based on the statistic we would cluster destination IP address using input feature vector. Clustering is the process of examining a collection of points, and grouping the points into clusters according to some distance measure.The goal is that points in the same cluster have a small distance from one another, while points in different clusters are at a large distance from one another [10].

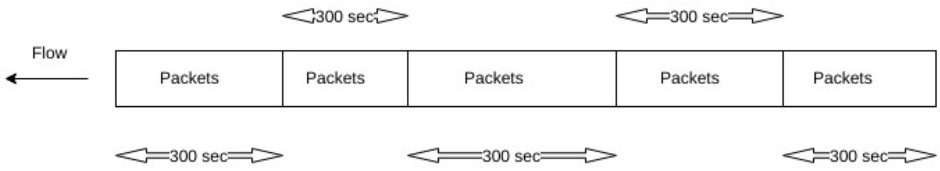


Figure 4.2: Data Flow Segments

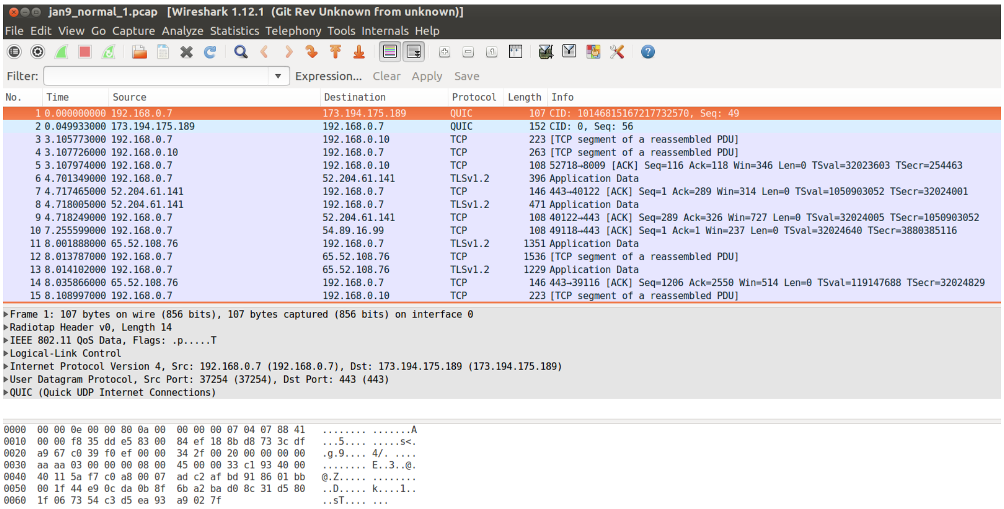
Once cluster is learned it would be used as bench mark for all future flow. Router would constantly keep clustering destination IP and if there is deviation in the normal traffic at router for a destination then that would affect the clustering and would cause destination to be placed in different cluster, cluster with more packet frequency. This change in clustering would be reported to destination network, which then decide on regulating the traffic

Chapter 5

Implementation

Botnet(Network of bots) plays important role in DDoS attack. Bots are the compromised system controlled by attacker for launching attack. Those can be anywhere on the Internet they are not bounded by geographical boundaries. The major DDoS attacks are the volume based attacks. So for this paper we would be considering only the volume based attack for the simplicity.

Wireshark, an open source application, was used for capturing Internet packet. Packets are captured for 5 hr on a single router for the demonstration purpose. Once the packets are captured they are saved as pcapng file which is a Wireshark file format for captured packets. Wireshark capture every detail of the packets but we don't need all of the information, we would only be interested in the IP packets.


jan9_normal_1.pcap [Wireshark 1.12.1 (Git Rev Unknown from unknown)]
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Filter:
Expression...
Clear
Apply
Save

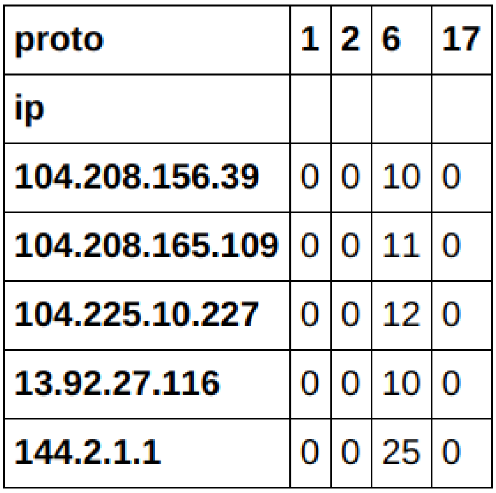
```
No.  Time        Source           Destination      Protocol Length Info
   1 0.000000000 192.168.0.7      173.194.175.189  QUIC       107 CID: 10146815167217732570, Seq: 49
   2 0.049933000 173.194.175.189  192.168.0.7      QUIC       152 CID: 0, Seq: 56
   3 3.105773000 192.168.0.7      192.168.0.10     TCP        223 [TCP segment of a reassembled PDU]
   4 3.107726000 192.168.0.10     192.168.0.7      TCP        263 [TCP segment of a reassembled PDU]
   5 3.107974000 192.168.0.7      192.168.0.10     TCP        108 52718→8009 [ACK] Seq=116 Ack=118 Win=346 Len=0 TSval=32023603 TSecr=254463
   6 4.701349000 192.168.0.7      52.204.61.141    TLSv1.2    396 Application Data
   7 4.717465000 52.204.61.141    192.168.0.7      TCP        146 443→40122 [ACK] Seq=1 Ack=289 Win=314 Len=0 TSval=1050903052 TSecr=32024001
   8 4.718005000 52.204.61.141    192.168.0.7      TLSv1.2    471 Application Data
   9 4.718249000 192.168.0.7      52.204.61.141    TCP        108 40122→443 [ACK] Seq=289 Ack=326 Win=727 Len=0 TSval=32024005 TSecr=1050903052
  10 7.255599000 192.168.0.7      54.89.16.99      TCP        108 49118→443 [ACK] Seq=1 Ack=1 Win=237 Len=0 TSval=32024640 TSecr=3880385116
  11 8.001888000 65.52.108.76     192.168.0.7      TLSv1.2   1351 Application Data
  12 8.013787000 192.168.0.7      65.52.108.76     TCP       1536 [TCP segment of a reassembled PDU]
  13 8.014102000 192.168.0.7      65.52.108.76     TLSv1.2   1229 Application Data
  14 8.035866000 65.52.108.76     192.168.0.7      TCP        146 443→39116 [ACK] Seq=1206 Ack=2550 Win=514 Len=0 TSval=119147688 TSecr=32024829
  15 8.108997000 192.168.0.7      192.168.0.10     TCP        223 [TCP segment of a reassembled PDU]
```

```
▶Frame 1: 107 bytes on wire (856 bits), 107 bytes captured (856 bits) on interface 0
▶Radiotap Header v0, Length 14
▶IEEE 802.11 QoS Data, Flags: .p.....T
▶Logical-Link Control
▶Internet Protocol Version 4, Src: 192.168.0.7 (192.168.0.7), Dst: 173.194.175.189 (173.194.175.189)
▶User Datagram Protocol, Src Port: 37254 (37254), Dst Port: 443 (443)
▶QUIC (Quick UDP Internet Connections)
```

```
0000  00 00 0e 00 00 80 0a 00  00 00 00 07 04 07 88 41   ........ .......A
0010  00 00 f8 35 dd e5 83 00  84 ef 18 8b d8 73 3c df   ...5.... .....s<.
0020  a9 67 c0 39 f0 ef 00 00  34 2f 00 20 00 00 00 00   .g.9.... 4/. ....
0030  aa aa 03 00 00 00 08 00  45 00 00 33 c1 93 40 00   ........ E..3..@.
0040  40 11 5a f7 c0 a8 00 07  ad c2 af bd 91 86 01 bb   @.Z..... ........
0050  00 1f 44 e9 0c da 0b 8f  6b a2 ba d0 8c 31 d5 80   ..D..... k....1..
0060  1f 06 73 54 c3 d5 ea 93  a9 02 7f                  ..sT.... ...
```

Figure 5.1: Wireshark Tool: snippet of captured packets

To extract data from the captured packets data extraction program *data_extraction* is written in Python. Flow is logically treated as IP packets captured over 300 second time and that is considered as one sample flow. Because of of the limited resource each frame is written to the file with time and sample number. Once we have extracted information form all the packets and sampled in different files then we can use them to create feature table. Another program *IP_clustering* is used for clustering the samples. Our data sample would look like below



proto	1	2	6	17
ip				
104.208.156.39	0	0	10	0
104.208.165.109	0	0	11	0
104.225.10.227	0	0	12	0
13.92.27.116	0	0	10	0
144.2.1.1	0	0	25	0

Figure 5.2: Flow Information for given destinations

Number in the table represents the frequency of the packet of the particular protocol destine for particular destination. For our clustering algorithm, frequency of the protocol would serve as feature vector. For the sake of simplicity we would be dealing with TCP and UPD flood attacks. Attack is engineered using LOIC (Low Orbit Ion Canon), a free tool which is even used by attackers in the real world. Flow based model is build as it is more reliable and fast because packet analyzing is often difficult due to size and encryption also destination port could not a reliable information in detecting attacks as attacker use different ports.

There are many algorithms which can be used to learn from the data, they can be categorized as Supervised or Unsupervised.

According to Tom Mitchell (1998) A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience.

Let f be the function which we need to guessed from input vector X = x^1, x^2, ..., x^m and h be the hypothesis about the function f. $h \in H$ and $f \in H$, where H is class of functions. Both f and h can be vectors. We select h based on a training set, Ξ, of m input vector examples. In Supervised leaning we know the values of f for the m samples in the training set Ξ. We assume that if we can find a hypothesis, h, that closely agrees with f for the members of Ξ, then this hypothesis will be a good guess for f when Ξ is large. In Unsupervised learning, we simply have a training set of vectors without function values for them. The problem in this case, typically, is to partition the training set into subsets, Ξ_1, ,Ξ_R, in some appropriate way. [10]

Supervised algorithm such as One Class Support Vector Machine(SVM) could be efficient to identify the anomalies in the data but it is very process and memory intensive so they can be used only for the highest cluster IP address. Unsupervised learning algorithms are fast and consume less data so they would be useful for our analysis at router level.

k-means clustering is one of the most efficient algorithm for creating clusters. In k-means, problem is to find set of k points ($k \in \mathbb{N}$) in $\mathbb{R}^d$ $(\mu_1, \mu_2, ..., \mu_k)$, called centers, such that mean square distance $(argmin_k||x^i{-}\mu_k||^2)$ from each data point from the set of n data points in $d - dimensionl$ space $\mathbb{R}^d$ to its nearest center from k centers is minimum. Lloyd's algorithm is most popular heuristic for k-means clustering.

Chapter 6

Algorithms

Algorithm 1 Lloyd's algorithm k-means algorithm

```
$x^1$, $x^2$, ..., $x^m$ is training set, where $x^i \in \mathbb{R}^d$, $i = 1, 2, .., m$
Randomly initialize k cluster centroids $\mu_1$, $\mu_2$, ..., $\mu_k \in \mathbb{R}^d$
repeat
    for $i \leftarrow 1, m$ do
        $c^i = argmin_k||x^i - \mu_k||^2$                          ▷ Cluster Assignment
    end for
    for $k \leftarrow 1, K$ do
        $\mu_k = \frac{1}{n}[x^{(k_1)} + x^{(k_2)} + ... + x^{(k_n)}] \in R$        ▷ Move Centroid
    end for
until convergence
```

Algorithm 2 Algorithm for DDoS attack Detection

```
Divide traffic into minor time window (e.g. 300 sec).
Filter data based on the required features.
Run Clustering algorithm on the data. (Use elbow method to decide on number of clusters).
Find the probability of certain IP lying in a particular cluster.
Once the clusters are decided our model is ready.
now to test model we would cluster and label each IP and if there is change in the cluster.
If No change → traffic is normal.
If change change → traffic not normal.
```

Once the change in the normal behavior has been reported then router communicate to the destination network. Destination network collect that information to know the source of the attack. This information can be used to know the nature of the attack.

6.1 Conclusion

A novel way to identify and mitigate DDoS attack is discussed in the paper. With the advance of NVF(Network Virtual Functional) it would be easy to push the learning algorithms to the routers and thus making them efficient in detecting the attacks. Also ICMP protocol can be used to communicate location and attack information in between the routers.

Bibliography

[1] Derek Kortepeter. Destructive ddos attacks increasing at a rapid rate. http://techgenix.com/ddos-attacks-increasing/, December 2017. [Online; accessed 22-August-2017].

[2] HowStuffWorks. What is a packet? https://computer.howstuffworks.com/question525.htm. [Online; accessed 20-January-2018].

[3] Ken Dunham and Jim Melnick. *Malicious Bots: An Inside Look into the Cyber-Criminal Underground of the Internet.* CRC Press, August 2008.

[4] Network Working Group. Traffic flow measurement: Architecture. https://www.ietf.org/rfc/rfc2722.txt, October 1999.

[5] Cisco. Introduction to cisco ios netflow. https://www.cisco.com/c/en/us/products/collateral/ios-nx-os-software/ios-netflow/prod_white_paper0900aecd80406232.html, May 2012. [Online; accessed 22-August-2017].

[6] sFlow. Using sflow. http://www.sflow.org/using_sflow/index.php. [Online; accessed 16-November-2017].

[7] Omar Santos. The size of the internet global routing table and its potential side effects. https://supportforums.cisco.com/t5/network-infrastructure-documents/the-size-of-the-internet-global-routing-table-and-its-potential/ta-p/3136453, May 2014. [Online; accessed 22-August-2017].

[8] Cisco. What is a network switch vs. a router? https://www.cisco.com/c/en/us/solutions/small-business/resource-center/connect-employees-offices/network-switch-what.html. [Online; accessed 16-November-2017].

[9] Network Working Group. Icmp. https://tools.ietf.org/html/rfc792, Septmber 1981. [Online; accessed 16-November-2017].

[10] Jeffrey D. Ullman Jure Leskovec, Anand Rajaraman. *Mining of Massive Datasets.* Stanford University, 2014.