# AUTOMATIC BOTNET DETECTION BASED ON BIPARTITE GRAPH AND ONE MODE PROJECTION FOR NETWORK FLOW TRACES

[1]Sneha Joshi, 2Dr. Anitha Patil, [3]Prof. Srijita Bhattacharjee.
[123]Department of computer engineering, Pillai HOC College of Engineering and Technology, Rasayani, Maharashtra, India.

***Abstract :*** A continuous increase in the usage of internet has now resulted in an important topic of research in network traffic analysis domain. Due to the growth in the internet traffic, the complexity in the network botnet detection has increased. It has become a progressively crucial task to understand the behavioral patterns of the users who use various net services and network applications. The proposed system presents a unique approach for botnet detection in the internet traffic. This approach is supported by the behavioral graph analysis to check the similarity in the behavior of hosts. The proposed system uses bipartite graphs to model host communications from network traffic. This is followed by building one mode projections of supported bipartite communication graphs. The one mode projection helps to analyze the similarity in the social-behavior communication of end-hosts. The system uses the economical bunch algorithms on the similarity matrices and grouping related to one-mode projection graphs. It performs network awareness on the group of end hosts within the same network prefixes. The proposed system demonstrates results based mostly on real datasets and shows that end-host and application behavior clusters provide distinct traffic options that proves to improve the interpretations of the internet traffic. It demonstrates the sensible edges of exploring behavior similarity in identification of the network behavior, discovering botnets in the network applications, and sleuthing abnormal traffic patterns.

***IndexTerms*** **- Networking, Distributed Denial-of-Service, DDoS, Cyber security, Signal Processing for Network Security.**

## I. INTRODUCTION

Cyber-Security is one of the vital challenge these days. The various forms of attack consist of phishing, website sabotages, etc. There have been cases where it was proved that the terrorist attacks were carried out by compromising cyber security. This makes it even more necessary to protect our digital network. The internet has become a soft target to carry out attacks. It is possible to hide and carry out a broad sort of attacks through internet network.

Denial-of-Service (DoS) attack is usually carried out by many attackers. Therefore, it is classified as an important attack for detection. It is executed by sending multiple bulk of request to one server, so much so that, the receiving capacity of the server becomes less than the amount of requests received. This paralyses the server and makes it unable to serve any request. Now these attacks are executed with the help of bots. Bots can be used to send bulk of requests to a particular server. Sometimes a legitimate user can also send some packets to a server. The detection algorithm should be smart to detect only the attacks carried out by malicious nodes and not by legitimate users. It is important to detect if the attack is carried out by a malicious source, and, if so, there has to be a speedy identification technique to identify the hidden botnet. This type of detection is very important so that there will be a clear classification between a legitimate source and a malicious source and only the malicious source will be detected. Building such algorithm is the main aim of this work.

### A. RELATED WORK

The earliest DoS paradigms (see, e.g., TCP SYN flooding), relied on specific protocols vulnerabilities, and were charac- terised by the repetition of 1 (or few) requests with a large rate [1]. During this state of affairs, the only supply of the attack will be known by computing its remarkably giant request rate. The distributed variants of such attacks exploit primarily the same reasonably vulnerabilities and repetition schemes, except for the fact that the big request rate is currently obtained by aggregating many little individual larva rates. yet, in such attacks, the bots are often still known at a single-user level. Indeed, normal traffic patterns square measure usually characterised by a precise degree of innovation, whereas the repetition theme implicitly emphasizes the larva character. In fact, many helpful inferential strategies are projected for such reasonably DDoS attacks. The literature regarding DDoS attacks is made. With no pretence of completeness, we have a tendency to introduce shortly some recent works on the subject, and that we refer the Reader to the survey in [2] for a additional comprehensive outline. In [3], applied math strategies to identify DDoS attacks square measure projected, wishing on computing entropy and frequency-sorted distribu- tions of elect packet attributes.

The DDoS identification is then supported the detection of anomalies within the characteristics of the packet attributes. In [4], the Authors propose a stratified technique supported macroscopic-level network observation to capture shifts in spatial-temporal traffic patterns, that square measure then accustomed inform a detection system regarding wherever and once a DDoS flooding attack presumably arises in an exceedingly supply network. The work conferred in [5] depends on the appliance of AN entropy detection technique, where the key to spot the DDoS attack is that the randomness of some attributes within the packets headers.

In [6], two new information metrics, the generalized entropy metric and therefore the information distance metric, square measure used to observe lowrate DDoS attacks, by evaluating the difference between legitimate and attack traffic. A mathematical model to look at shrew DDoS attacks (where TCP flows square measure affected to a small fraction of their ideal rate at low attack costs) is introduced in. The Authors propose a technique aimed at capturing the adjustment behaviors of TCP congestion window at the victims facet, so as to judge the interaction be- tween attack patterns and network surroundings. More closely connected to the current work is that the new category of application-layer DDoS attacks, that is recently rising as one of the foremost powerful threats. In such attacks, the malicious traffic patterns square measure disguised as traditional ones by investment the numerous potentialities offered at the appliance layer (for instance, once water sport through an internet site, more and more web-pages square measure probably to be explored as time elapses). By assigning a adequate degree of variability to every individual bots pattern, identification ways supported single-user scrutiny become harmless. Building on such new potentialities, in this work we have a tendency to shall introduce a proper model for DDoS attacks wherever the botnet gets at its disposal a precise emulation dictionary to make the traffic patterns[10].

**B. DDOS Attack using STATIONARY BOTNETS**

The attacker commands and sets up a bot to start the attack. It infects the machine first and then infests an infected code that allows the attacker to manage the machine as an administrator. This attacker now becomes the bot master, who can command and control the server and perform activities like sending and receiving packets from the connected machines. The node thus starts becoming malicious as soon as the code is embedded in it. Once malicious, it works as a larva during the attack. The node that receives a large variety of attacking packets from the malicious hosts is the target node that the attacker wants to actually infect.

1) Attacker: The aggressor configures the bots to initiate an attack. At first, it compromises a machine to put in a malicious code and the gains the management of the machine to join the management server.

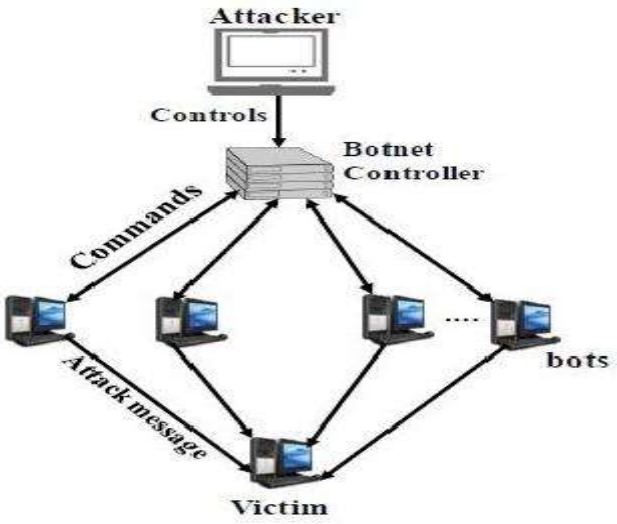



fig. 1. stationary botnet

2)Botnet controller: It works as a botnet command and control server that may send/receive communication commands from the connected parties.

3)Compromised host: Once a machine is compromised by installing malicious code, it works as a larva during a botnet.

4)Victim: It is the target host for the aggressor that receives a large variety of attacking packets from the compromised hosts.

Botnets are an important aspect to carry out a DDoS attack. The history of botnet started from Internet Relay Chat (IRC). IRC was a text-based chat system. The IRC system was made with a good mind. It was meant for services such as sharing of messages, body support, games and an alternative to chatter. Attackers realized its potential and started to exploit its power by executing malicious activities. They started to launch attacks through internet. According to Arbor Networks, once, six C&C servers utilized over twenty five thousand infected machines. They attacked the CMS systems. This technique uses thousand infected hosts that executes co-occurring attacks with the help of a bot-master has made. Recently an attack was detected that used botnets with redoubled complexness. Figure below shows an example of DDoS attack generated by the exploitation of a botnet.

**A. Botnet Characteristics**

Botnets are classified according to the C&C system that is used for communication. The C&C system is responsible for the communication between the malicious node and the botmaster. The mostly used C&C system unit for communication are the centralized and distributed mechanisms. These system have some limitations. Peer-to-Peer botnet is recently introduced which is

very difficult to defend. Understanding the malicious code and enhancing the code is essential to tackle the botnet attack. It is essential to study the behavior of the botnet architectures, topologies, and protocols used for attacks.

**B. Botnet MODEL**

Agent handler model, IRC botnet model and web-based model are three different models that are used to carry out a Botnet-based DDoS attack.

The Agent Handling Model consists of four threads, namely, assailant, handlers, agents and the victim. The initiator among these four threads is the handler. It is responsible to infest the other host and gain control over them. Once the host is compromised, the handlers infests the machine with malicious code. This allows the handler to carry out the DDoS attack. It becomes tough to trace an attack if the assailant uses the help of the handler to infest a machine. This is because, during detection, the handler is detected, and the main attacker is set free. The third thread are the agents. These agents are basically the code itself. They are the actual cause of the DDoS attack. The attacker uses the weakness of the protocols like UDP and ICMP to conduct such DDoS attacks. The assailant uses multiple handlers to carry out attack. These masks the actual attacker. The last thread are the victims. These are the machines over which the attack is actually performed.

Botnet are also used to carry out attacks in network and web-server. Different types of botnets have inherited existence throughout the recent years. Some of the important bots that have to be studied are the Agobot, SDBot, RBot, SpyBot and the Conficker.

Agobot is a malicious code written in C++/assembly language. To infest such bots, attacker uses the password protected IRC interface, remote update and removal of the put in bots, execution of programs and commands for DDoS attacks. It also uses the port scanning to find and infect different hosts. SDBot has backdoor capabilities and knowledge stealing techniques. It sails though the network to exploiting its vulnerabilities and weakness. RBot is also a family of backdoor attackers that uses remote administration utility programs. Once they gain access, It gives access to a remote user to manage it over the Internet. A malicious users are thus capable to manage the associated infected systems. This sometimes can be done without the knowledge or permission of the systems administrator. These bots uses the backdoors remotely to perform an attack such as information stealing on the victim's machine. SpyBot is the successor of SDBot. It connects to the IRC server and performs action as commanded by the botmaster. The Conficker is one of the powerful botnet that has attacked a lot of users in the government and the business space. It emphasizes on the errors in the Windows package and plants an attack on the system passwords.

## II. SOME STATIONARY BOTNETS

In recent times, most refined attacks on networks or Web servers are launched by botnets. many sorts of botnets have inherit existence throughout the past few years. In this section, we introduce an inventory of botnets used for launching various sorts of attacks.

1)Agobot : it's a multi-threaded larva written in C++/assembly language by a German software engineer celebrated as Axel past Gembe. Agobot could be a pioneer IRC larva used for attack generation. Some nec- essary options of Agobot are: (i) password protected IRC shopper management interface, (ii) remote update and removal of the put in bots, (iii) execution of programs and commands for DDoS attacks and (iv) port scanning to search out and infect different hosts.

2)SDBot : This larva is provided with many backdoor capabilities and knowledge stealing routines. It will propagate through network shares and exploited vulner- abilities.

3)RBot : This Ruby IRC larva creates an oversized family of backdoors - remote administration utility programs. Once the backdoors area unit with success put in, it permits a remote user to access and management it over a network or the Internet. a far off user with malicious intentions is also able to management associate infected pc, sometimes while not the knowledge or consent of the systems legitimate user(s) and uses the backdoors remotely to perform a spread of actions, equivalent to stealing information, corporal punishment commands on the affected machine or accessing different machines on a local network.

4)Spybot : Spybot could be a successor of SDBot. It con- nects to a delegated IRC server and receives commands from the botmaster through selected channels.

5)Conficker : it's one amongst the foremost powerful and most effective pc worms that has infected ample users in government and business in over two hundred countries since 2003. It uses flaws within the Windows package and mounts lexicon attacks on administrator passwords to propagate whereas forming a botnet. it's not trivial to counter as a result of it combines many advanced malware techniques.

## III. LITERATURE SURVEY

### A. Mitigation of random query string DoS via gossip: S. Ferretti and V. Ghini,

They eplain the DDoS attacks and gift a categoryification of DDoS attacks and characteristics of every class of DDoS attacks. They then introduce botnet technology, the primary help of recent DDoS attacks, and up to date trends within the launching of varied forms of DDoS attacks using botnets. A discussion of mobile botnets and ancient PC-based botnets is given, followed by temporary comparison between the 2. they supply a close discussion of botnet-based DDoS defense approaches and strategies. Though, within the past twenty years, a decent range of defense solutions are introduced to counter DDoS attacks with increasing

sophistication, still there square measure many vital problems and analysis challenges that square measure open and nevertheless to be addressed .

**B. J. Luo, X. Yang, J. Wang, J. Xu, J. Sun, and K. Long, On a mathematical model for low-rate shrew DDoS**

In this paper we tend to propose a mathematical model for explanation shrew attacks result and revealing the attacks properties. Such model permits US to explore totally different attack patterns that may be used by Associate in Nursing ag- gressor, so simpler defense strategies is designed consequently. totally different from previous works, the projected model takes under consideration the specific behaviors of TCPs con- gestion window adaptation mechanism, so it will comprehen- sively judge attack result from each attack pattern and network surroundings. The simulation results indicate that the relative error of our model remains around 10% for many attack patterns and network environments, whereas the benchmark model in previous works (which ignored congestion window adaptation mechanism) includes a mean relative error of sixty nine.57% and a most of 186.17%. In addition, our model uncovers some novel properties of shrew attack, equivalent to the minimum DoS amount for given peak length (T L ) and also the international minimum peak length (Lmin). The former reveals the very best outturn degradation is caused by shrew attack, whereas the latter shows very cheap attack price to launch a thriving attack. With these analytic deductions, we tend to square measure ready to discover the approach of calibration the attack parameters to enhance the attack result in an exceedingly given network environment, and also the approach of reconfiguring the network resources (e.g., the information measure of bottleneck link, the buffer size, etc.) to mitigate shrew DoS with given attack pattern. Moreover, we tend to square measure ready to style an easy defense strategy based on our theoretical results, i.e., mitigating shrew attack by temporarily increasing the bottleneck buffer size. Simulation demonstrates that this strategy will recover nearly 1/2 the bottleneck transmission rate for general attack patterns, and to breach it the aggressor should exploit high-rate knowledge streams. At last, our model indicates that some conclusions in [1] turn out to be rather inaccurate in some cases (e.g., we find that [1] seriously underestimated the lurking nature of shrew attack). one among the results is that a lot of existing defense strategies, particularly those that were designed and valid on the premise of these inaccurate conclusions, perhaps after all unable to defend against shrew attack effectively.

**C. Botnet in DDoS Attacks: Trends and Challenges, Nazrul Hoque, Dhruba K Bhattacharyya and Jugal K Kalita**

They eplain the DDoS attacks and gift a category verification of DDoS attacks and characteristics of every class of DDoS attacks. They then introduce botnet technology, the primary assistant of contemporary DDoS attacks, and up to date trends within the launching of varied sorts of DDoS attacks victimization botnets. A discussion of mobile botnets and ancient PC-based botnets is given, followed by temporary comparison between the 2. they supply an in depth discussion of botnet-based DDoS defense approaches and strategies. Though, within the past twenty years, an honest range of defense solutions are introduced to counter DDoS attacks with increasing sophistication, still there ar many vital problems and analysis challenges that ar open and however to be self- addressed.

**D. DDoS Attack Detection Algorithms Based on Entropy Computing, Liying Li, Jianying Zhou, and Ning Xiao**

In this paper, they justify the DDoS detection algorithms supported entropy monitoring, and planned 2 improved en- tropy detection approaches: additive entropy and time-based strategies. They additionally conducted experiments for the normal entropy detection technique and also the additive entropy detection technique, severally. From the take a look at results, they might see that our additive entropy detection technique has smart detection capability. for various network environments, the way to set up the edge worth may be a key purpose, that influences the detection potency. within the time- based entropy detection technique, they introduced a brand new conception of your time cumulating. By setting a systems tolerable detection time, DDoS detection are often dole out while not giving a typical threshold worth.

**E. Monitoring the Macroscopic Effect of DDoS Flooding Attacks, Jian Yuan and Kevin Mills**

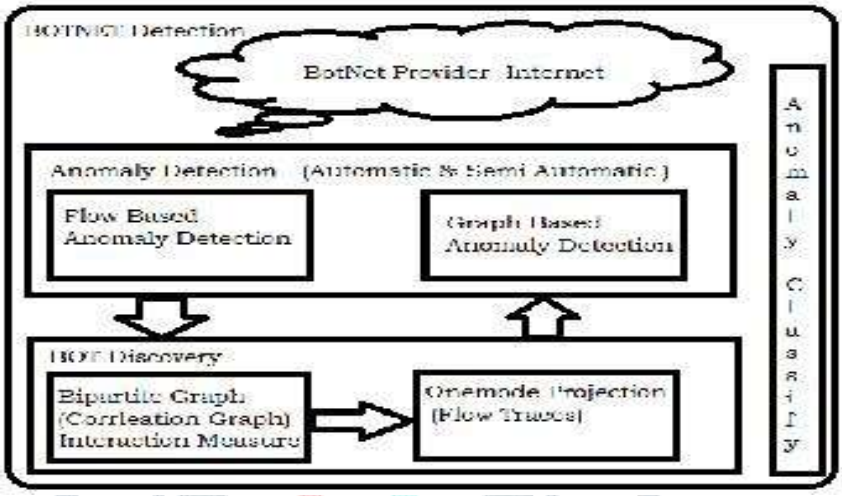
They planned a way for early detection of DDoS flooding attacks by watching gross (network-wide) effects. They ex- permeated with completely different attack modes: constant rate, increasing rate, natural-network-congestion-like, pulsing, TCP-targeted, and subgroup attacks. They found that these attacks, that have the apparent result of causation network congestion, reveal themselves through shifts in spatial-temporal patterns that exhibit an equivalent signature: congestion at the victim network. Their associate degree analysis technique reveals the time and placement of an attack while not observations from the suffering victim. They additionally realize the dynamic nature of elect attack varieties (e.g., subgroup and pulsing attacks) is also advantageous for our analysis technique as a result of exaggerated correlation iatrogenic by changes in attack traffic keeps the load of the attack victim salient for extended periods. Their results recommend that gross level watching can be each sensible and useful for triggering a lot of targeted detection and filtering in transit or supply network.

**F. Low-Rate DDoS Attacks Detection and Trace back by Using New Information Metrics, Yang Xiang, Ke Li, and Wanlei Zhou**

They propose 2 new and effective info metrics for low-rate DDoS attacks detection: generalized entropy and knowledge distance metric. The experimental results show that these metrics work effectively and stably. They outmatch the normal engineer entropy and Kull back Leibler distance approaches, severally, in detection anomaly traffic. specially, these metrics will improve (or match the assorted requirements of)

the systems detection sensitivity by effectively adjusting the worth of order of the generalized entropy and knowledge distance metrics. because the planned metrics will increase the knowledge distance (gap) between attack traffic and legit traffic, they'll effectively sight low-rate DDoS attacks early and scale back the false positive rate clearly. The planned info distance metric overcomes the properties of uneven of each KullbackLeibler and knowledge divergences. moreover, the planned scientific discipline traceback theme supported info metrics will effectively trace all attacks till their own LANs (zombies). lastly, our planned info metrics will considerably improve the performance of low-rate DDoS attacks detection and scientific discipline traceback over the normal approaches.

## IV. SYSTEM MODULES




Agent handler model, IRC botnet model and web-based model are three different models that are used to carry out a Botnet-based DDoS attack.

The Agent Handling Model consists of four threads, namely, assailant, handlers, agents and the victim. The initiator among these four threads is the handler. It is responsible to infest the other host and gain control over them. Once the host is compromised, the handlers infests the machine with malicious code. This allows the handler to carry out the DDoS attack. It becomes tough to trace an attack if the assailant uses the help of the handler to infest a machine. This is because, during detection, the handler is detected, and the main attacker is set free. The third thread are the agents. These agents are basically the code itself. They are the actual cause of the DDoS attack. The attacker uses the weakness of the protocols like UDP and ICMP to conduct such DDoS attacks. The assailant uses multiple handlers to carry out attack. These masks the actual attacker. The last thread are the victims. These are the machines over which the attack is actually performed.

Botnet are also used to carry out attacks in network and web-server. Different types of botnets have inherited existence throughout the recent years. Some of the important bots that have to be studied are the Agobot, SDBot, RBot, SpyBot and the Conficker.

Agobot is a malicious code written in C++/assembly language. To infest such bots, attacker uses the password protected IRC interface, remote update and removal of the put in bots, execution of programs and commands for DDoS attacks. It also uses the port scanning to find and infect different hosts. SDBot has backdoor capabilities and knowledge stealing techniques. It sails though the network to exploiting its vulnerabilities and weakness. RBot is also a family of backdoor attackers that uses remote administration utility programs. Once they gain access, It gives access to a remote user to manage it over the Internet. A malicious users are thus capable to manage the associated infected systems. This sometimes can be done without the knowledge or permission of the systems administrator. These bots uses the backdoors remotely to perform an attack such as information stealing on the victim's machine. SpyBot is the successor of SDBot. It connects to the IRC server and performs action as commanded by the botmaster. The Conficker is one of the powerful botnet that has attacked a lot of users in the government and the business space. It emphasizes on the errors in the Windows package and plants an attack on the system passwords.

Automatic botnet detection by bot discovery and anomaly detection is divided into two modules likewise,

1)Botnet Discovery

Bipartite Graph
One Mode Projection

2)Anomaly Detection

Flow Based Anomaly Detection
Graph based anomaly Detection

In automatic botnet discovery proposed system implements Bot discovery by construction of bipartite graph for internet network resources communication to identify network flow traces.

1)Bipartite Graph:-

Traffic analysis is the process of intercepting and ex- amining messages in order to deduce information from patterns in communication. It can be performed even when the messages are encrypted and cannot be de- crypted. A bipartite graph is a graph whose vertices can be divided into two disjoint sets and (that is, and are
each independent sets) such that every edge connects a vertex in to one in. Vertex set and is often denoted as partite sets.

2)One projection: -

is an extensively used method for compressing infor- mation about bipartite networks, since the one-mode projection is always less informative than the original bipartite graph, an appropriate method for weighting net- work connections is often required. Optimal weighting methods reflect the nature of the specific network, con- form to the designers objectives and aim at minimizing information loss.

3)Anomaly Detection:-

In anomaly detection proposed a method for detect- ing the compromised nodes. This is based on ideas from community detection in social networks. However, this system devised a refined modularity measure that is suitable for botnet detection. Anomaly detection is performed on network flow traces between source and destination communication resources.

Flow based detection:-

This approach is used to estimating the histogram of quantized flows; the first anomaly detection method quantizes flow-level data (e.g., Cisco NetFlows) and monitors the histogram of quantized flows. approach is to quantize flows and to treat them as independent observations of a discrete random variable. We first cluster network addresses (part of the flow charac- teristics) into a manageable number of clusters.

Graph based detection:-

Graph based approach is used to estimating the degree distribution of interaction of node interaction graph .The second anomaly detection stage where, first identify a set of highly interactive nodes, which are referred to as pivotal nodes.

A graph based approach that processes packet-level data, each packet as an interaction record between the source and the destination. First group packets into windows based on their timestamps.

**A. Relevant mathematics associated with the System:**

System S is defined as S = Input, Process, Output

S=s, e, X, Y, Fme, DD, NDD,

Where,

s = Start of the Application 1. Create network 2. Analysis of Internet behavior
e = End of the Application
To analysis of the botnet detection analysis
X = Input of the program.
Source Node= S1, S2Sn be the set of source host Destination Node= D1, D2.Dn be the set destination node.
Number of Cluster= c1, c2,,cn

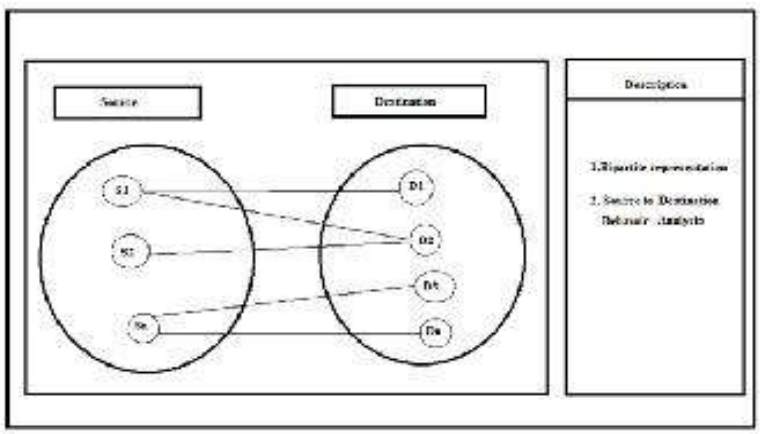


fig. 3. venn diagram representation

Y = Output of the program.
Result of internet botnet detection with bipartite graph.
X, Y U
Let, G is the Set of Bipartite Graph.
G= U, V, E
Where, G is the Graph and their element set.
U, V=these are the independent two disjoint sets of Graph G
E= Edges between communicating nodes of network;
NN=Nearest Neighbor classification in cluster
Fme = F1, F2, F3, F4, F5
Where,
F1= Network Creation.
F2= Bipartite graph designing.
F3= One mode Projection
F4= Cluster formation
DD= since Fme gives expected result, therefore problem is deterministic i.e NP Complete
SPACE COMPLEXITY:
The space complexity depends on Presentation and visu- alization of discovered patterns of Package.
NDD= TIME COMPLEXITY
Check No. of cluster between two disjoint set available in the network= n
If (n > 1) then retrieving of result of behavior analysis of internet traffic. So the time complexity of this algorithm is O(nn)
= Failures and Success conditions.
Failures:
1)Huge Travel Package dataset can lead to more time consumption to get the information.
2)Hardware failure.
3)Software failure.
Success:
1)Retrieval of packages and personalized Recommen- dation to tourist Datasets.
2)User gets result very fast according to their personal information provided.

## V. ANOMALY DETECTION

In this section, the work proposes two approaches for anomaly detections. There are two approaches to detect an anomaly. In the first approach, Net Flow is files are used. The approach quantifies the flow. In the second approach, pcap files are used. It aggregates the packets to graphs. It leverages the LDP for the graph degree distribution. The GLRT is projected to choose the foremost acceptable graph. The first approach provides a framework for network traffic. The setup of the approach is to quantize flows and to treat them as freelance observations of a definite variable. A flow has features such as the cluster label of the source destination IP address, Source port number, Destination port number and the duration of the flow. Let ref denote the probability vector calculated from some reference flows Fref and let Iflow( ) = D( jj ref ): the LDP proposes the following anomaly detector: Iflow(f) = 1(Iflow( f ) "))

where " is a detection threshold.

## VI. BOTNET DISCOVERY

Network anomaly detection technique can exclusively report whether or not there is an anomaly. The proposed botnet discovery technique use a slithering window to network traffic. It monitors them to store all the anomalies. The botnet detection algorithm becomes positive only when the number of detected anomalies is more than the threshold. For best results, the system unifies the output of both the approaches. An interaction record (ts; i; j) could be a tuple of timestamp, source and destination IP, and it represents that node i and j move at time ts (regardless of direction). For the first approach, an abnormal window is shown as collection of interaction records. The reason is because every flow is simply reborn to an interaction record. In the second approach, an interaction graph is shown as a collection of records because every edge corresponds to an interaction record. In each strategy, a suspect is depicted as a collection of records. The input of proposed botnet discovery technique could be a sequence of anomalies.

### A. Identification of important nodes:

Detecting bots directly is non-trivial. Instead, detecting the leaders (botmasters) or targets is easier as a result of they are additional interactive than traditional nodes. Botmasters need to command and control their bots to keep up the botnet, and bots actively move with victims in an exceedingly typical DDoS attack, even before the attack whereas they probe the targets. each leaders and targets, henceforward brought up as important nodes, ar extremely interactive. Suppose the full number of nodes that seem in an exceedingly is n. Let Gijk be the number of interactions between node i and node j in anomaly Sk. Then,

$$e_i = (1=jAj) \sum_{=1}^{jAj\,n} \sum_{j=1}^{ij} G_k \; ; \; i = 1; \; :; \; :; \; n$$

represents the amount of interaction of node i with all other nodes in A and will be called the total interaction measure of node i. We next define pivotal nodes.

(Pivotal nodes): The set of pivotal nodes is N = fi : ei > Tg, where T is a threshold.

We now focus on interactions between pivotal nodes and the remaining nodes. Pivotal nodes are either botmasters or attack targets; in either case their interactions with the bots are likely to be correlated. This section presents a technique that detects a community of highly correlated and highly interactive nodes. Compared to similar approaches in community detection, e.g., the leader-follower algorithm, our method takes advantage of not only temporal features (amount of interaction) but also correlation relationships. These relationships are characterized
using a graph, whose definition is presented next.

1) Construction of the Social Correlation Graph:

For $i = 1; ::::; n$, let $X_i$ represent the number of interactions between node i and pivotal nodes. For each anomaly $S_k A$, we obtain a sample of $X_i$ as $x_i^k = P_{j N} G_k^{ij}$:

Let $X_i = (1=jAj) \sum_{kj=1j}^{A} x_i^k$ be the sample mean and $(X_i) = (1= (A, 1)) \sum_{k=1}^{A} {}^{j} (x_i^k - X_i)^2$ be the sample standard deviation of $X$ for all i. The sample correlation coefficient is defined as

$$(X_i; X_j) = \frac{\sum^{jAj} {}_{k=1}[(x_i^k - X_i)(x_j^k - X_j)]}{(jAj\ 1)\ (X_i)\ (X_j)}$$

with the convention that (Xi; Xj) = 0 if (Xi) = 0 or (Xj) = 0

## VII. ALGORITHMS:

### A.Bipartite Communication

Input: - Load dataset
Output:-Group of similar source and destination hosts

1)Construct bipartite graph of host communication from flow traces;
2)Generate one mode projection of bipartite graph and its weighted adjacency matrix for end host in prefix and the obtain similarity matrix in prefix.
3)Construct diagonal matrix for adjacency matrix.
4)Compute laplacian matrix with adjacency matrix- degree of host matrix.
5)Normalize the row of matrix for clustering.
6)Perform clustering for source and destination end host for communication.
7)Assign the original IP address for clustering for row according to laplacian matrix.
8)Clustered IP address from available source and destination addresses.

### B.Classification Algorithm

Input: - Input source and destination communication with application category dataset.

Output: - cluster graph with category
If traces from available dataset do hostAddress=retrieveIPHost(dataset) Category=classifyEndHost(hostAddress);
clusterList(category)
While categoryList do of
clusterArray=addHostCategory(host Address, category) g
plotNetwork (clusterArray); Return clustered graph

## VIII. ANALYSIS

The Distributed Denial of Service attacks are attacks that can be caused by a botnet to compromise the security of the network. This attack can be targeted on the application layer and thus affect the interaction of the application layer with other layers which will crumble the system. This makes it very difficult to identify the real cause of the attack and also makes it troublesome to

detect the node that may have initiated the attack. This project proposes a unique model that uses a clustering algorithm to cluster all the nodes defined in the network. It then uses the one mode projection to detect the relation between each node and their communication links. The outcome is evaluated using multiple graphs that track the delay, throughput and precision levels of the proposed system. The clustering and the projection methods help to reveal the unknown indirect communication links in the network which would have either wise remained unexplored and would have been out of radar to analyze and detect any botnet activity.

| Name of botnet | Year | No of machines used | Architecture | Attack type | Protocol used | Platform | Reference |
|---|---|---|---|---|---|---|---|
| GTbot | 1986 | | Centralized | DDoS | IRC | Windows | [34] |
| EggDrop | 1993 | | Centralized | | IRC | Window's/Linux | [45] |
| Agobot | 2002 | | Centralized | DDoS | IRC/ HTTP | Windows | [23] |
| SDBot | | | Centralized | DoS | IRC | Windows | [24] |
| RBot | 2003 | | Centralized | DoS | IRC | Window's/Linux | [25] |
| Sinit | 2003 | | P2P | | IRC/HTTP | Windows/ Linux | [35] |
| Bagle | 2004 | 230,000 | Centralized/P2P | Spam sending | IRC | Windows | [36] |
| Phatbot | 2004 | | P2P | | HTTP/IRC | Windows | [36] |
| SpamThru | 2006 | 12,000 | P2P | Sending spam | IRC | Windows | [29] |
| Nugache | 2006 | 160,000 | P2P | | IRC | Windows | [31] |
| Rxbot | 2006 | | Centralized | | IRC | Windows | [37] |
| Spybot | | | Centralized | DoS | IRC | | [26] |
| Rustock | 2006 | 150,000 | Centralized | Sending spam/ DDoS | HTTP | Windows | [32] |
| MegaD | 2007 | 500,000 | P2P | DDoS | | Windows | [28] |
| Srizbi | 2007 | 400,000 | Centralize | DoS/DDoS | HTTP | Window's/Linux | [29] |
| Storm | 2007 | 160,000 | P2P | DDoS | IRC | Windows | [31] |
| Conficker | 2008 | 10.5 millions | P2P | Buffer overflow | HTTP | Windows | [27] |
| Torpig | 2008 | 180,000 | Centralized | Phishing/ man-in-the-middle | IRC | Windows | [30] |
| Onum | 2008 | | Centralized | Sending spam | IRC/ HTTP | Windows | [32] |
| Asprox | 2008 | 15,000 | Centralized | SQL injection | HTTP | Windows | [38] |
| Robax | 2008 | 185,000 | Centralized | | HTTP/UDP | Windows | [36] |
| Kraken | 2008 | 400,000 | Centralized | | HTTP | Windows | [40] |
| Waledac | 2009 | 90,000 | P2P | Sending spam | IRC | Window's/Linux | [41] |
| Cutwail | 2009 | | P2P | DDoS | IRC | Windows | [33] |
| Donbot | 2009 | 125,000 | Centralized | DDoS | TCP | Windows | [42] |
| Festi | 2010 | 25,000 | Centralized | DoS/email spam | HTTP | Windows | [43] |
| TDL-4 | 2011 | 4.5 million | P2P | DDoS/DoS | | Windows | [44] |

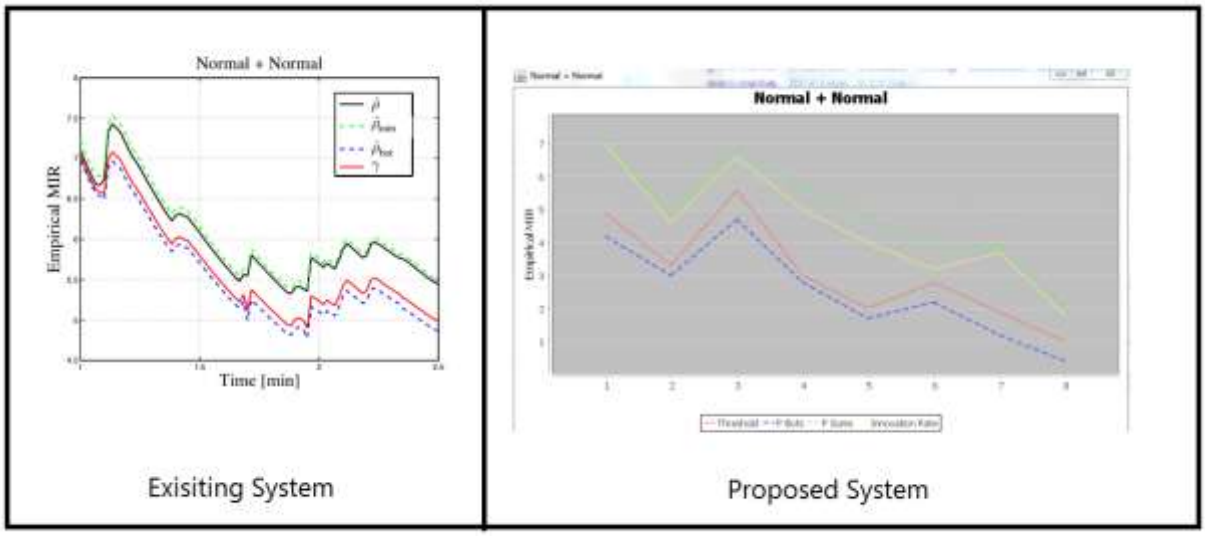
Fig. 4. Comparison Chart

**Graphical Analysis**

A. Normal + Normal:-

Normal + Normal addresses the case of a pair of normal users. The MIR stays (slightly) below the upper bound, meaning that a certain degree of correlation exists. However, the MIR stands clear above the threshold, as prescribed by (29), and confirming the validity of the BIC.

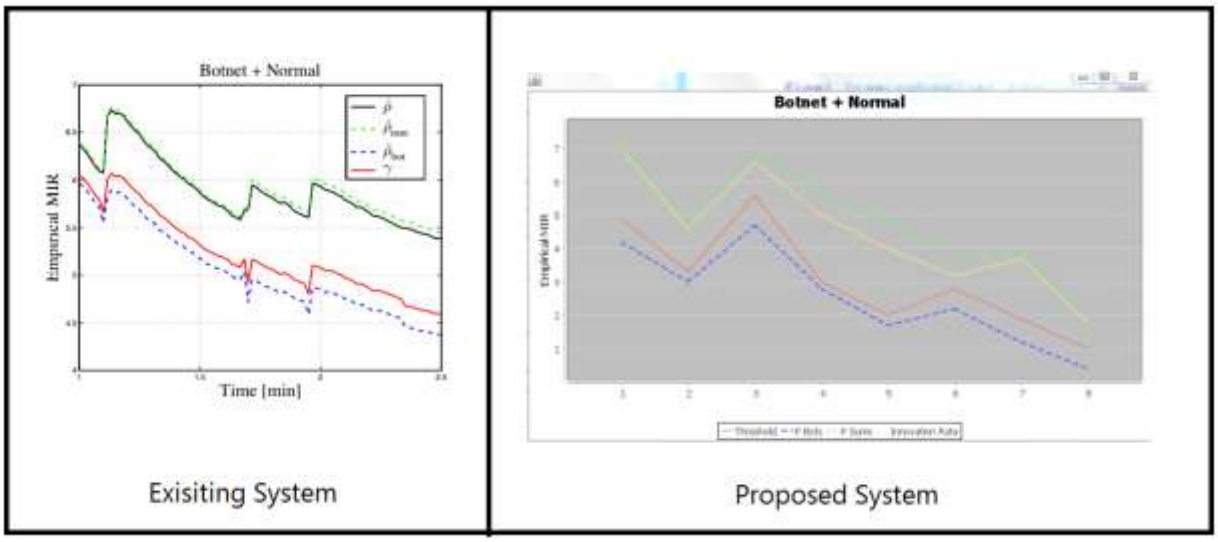
The proposed system and the existing system works same when there are no bot nets in the network




B. Botnet + Normal

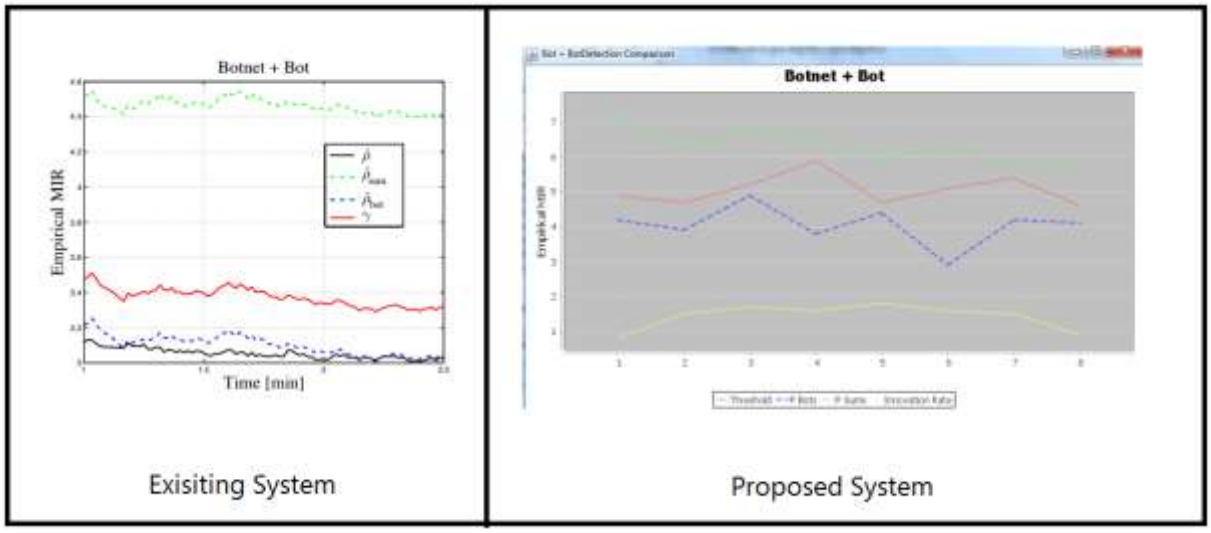
The Botnet + normal consists of two subnets under test, S1 (a botnet of size 10) and S2 (a normal user). Conclusions similar to those pertaining to a normal-normal pairing can be drawn, substantiating again the BIC. In proposed system, throughout the observation window, the activities of the two subnets are almost independent, i.e., the MIR essentially matches the upper bound.

In the existing system, as time elapses, a certain degree of correlation appears, but the MIR stays just above the threshold.




C. Botnet + Bot

Below is the case of a botnet/bot interaction. The existing system the empirical MIR approaches, as time elapses stands clear below the threshold. But in the proposed system, the activities of the two subnets are almost independent, i.e., the MIR essentially matches the upper bound.

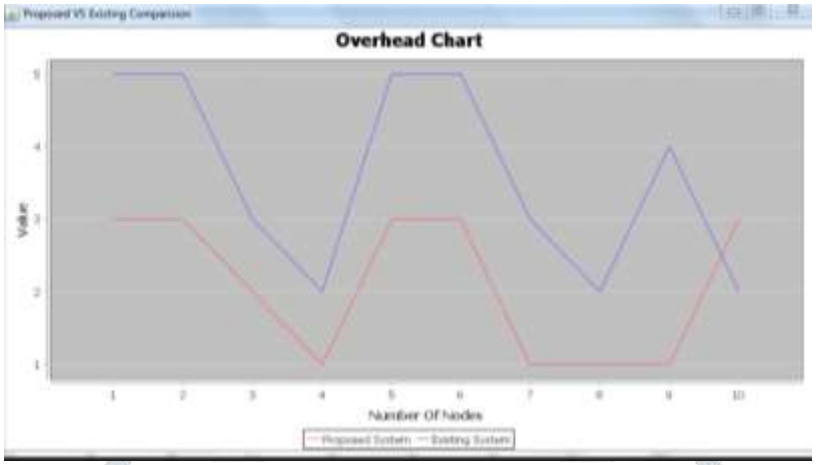



D. Delay Chart

The delay in sending the packets in the existing system is more than that of the proposed system. The below graph depicts that the delay in packet transfer is very less in the proposed system as compared to the existing system.

E. Overhead Chart

The proposed system has very less overhead as compared to the existing system. The packet transmission together with botnet detection with very less overhead is highly expected.

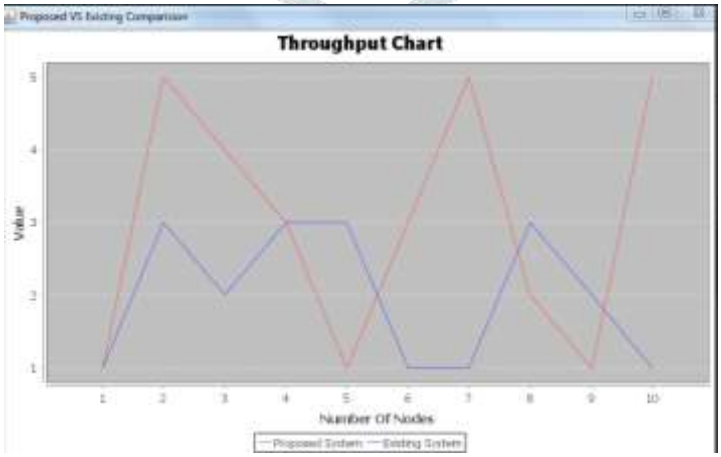



F. Packet Delivery Ratio Chart

The following graph depicts that the packet delivery success ratio of the proposed system is more than that of the existing system


G. Throughput Chart

In botnet detection, the throughput chart depicts the efficiency of detecting the botnet. The following graph depicts that the throughput graph of the proposed system is more than that of existing system.



## CONCLUSION

The Distributed Denial of Service attacks are attacks that can be caused by a botnet to compromise the security of the network. This attack can be targeted on the application layer and thus affect the interaction of the application layer with other layers which will crumble the system. This makes it very difficult to identify the real cause of the attack and makes it troublesome to detect the node that may have initiated the attack. This project proposes a unique model that uses a clustering algorithm to cluster all the nodes defined in the network. It then uses the one mode projection to detect the relation between each node and their communication links. Through coefficient clustering and other graph properties, interesting similarity matrices are found with respect to social behavior among different applications, and then a simple category-clustering algorithm is applied to group applications with similar social behavior into different clusters. The outcome is evaluated using multiple graphs that track the delay, throughput and precision levels of the proposed system. The clustering and the projection methods help to reveal the unknown indirect communication links in the network, which would have either wise, remained unexplored and would have been out of radar to analyze and detect any botnet activity.

## REFERENCES

[1]Nazrul Hoque, Dhruba K Bhattacharyya and Jugal K Kalita, Botnet in DDoS Attacks: Trends and Challenges,10.1109/COMST.2015.2457491, IEEE Communications Surveys & Tutorials

[2]Liying Li, Jianying Zhou, and Ning Xiao, DDoS Attack Detection Algorithms Based on Entropy Computing, ICICS 2007, LNCS 4861, pp. 452466, 2007.

[3]Jian Yuan and Kevin Mills, Monitoring the Macroscopic Effect of DDoS Flooding Attacks, IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, VOL. 2, NO. 4, OCTOBER-DECEMBER 2005.

[4]Yang Xiang, Ke Li, and Wanlei Zhou, Low-Rate DDoS Attacks Detec- tion and Traceback by Using New Information Metrics, IEEE TRANS- ACTIONS ON INFORMATION FORENSICS AND SECURITY, VOL. 6, NO. 2, JUNE 2011

[5]A. Chonka et al., Cloud security defence to protect cloud computing against HTTP-DoS and XML-DoS attacks, J. Netw. Comput. Ap- plicat. Jun. 23, 2010 [Online]. Available: http://dx.doi.org/10.1016/j. jnca.2010.06.004

[6]X. Jin et al., ZSBT: A novel algorithm for tracing DoS attackers in MANETs, EURASIP J. Wireless Commun. Netw., vol. 2006, no. 2, pp. 19, 2006. .

[7]G. Carl et al., Denial-of-service attack-detection techniques, IEEE In- ternet Comput., vol. 10, no. 1, pp. 8289, Jan./Feb. 2006.

[8]D. K. Bhattacharyya and J. K. Kalita, Network anomaly detection: A machine learning perspective. CRC Press, 2013.

[9]M. Feily, A. Shahrestani, and S. Ramadass, A survey of botnet and botnet detection, in Emerging Security Information, Systems and Tech- nologies, 2009. SECURWARE09. Third International Conference on. IEEE, 2009, pp. 268273.

[10]H. R. Zeidanloo, M. J. Z. Shooshtari, P. V. Amoli, M. Safari, and M. Za- mani, A taxonomy of botnet detection techniques, in Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on, vol. 2. IEEE, 2010, pp. 158162.