Prediction of Ongoing Attacks using System Behavioural Traces

S.Aiswarya Lakshmi[1], V.Pavithra[2], Dr.K.Narasima Mallikarjunan[3]

[1]UG Student, Dept. of Computer Science and Engineering, Thiagarajar College of Engineering, Madurai, Tamil Nadu, India

[2]UG Student, Dept. of Computer Science and Engineering, Thiagarajar College of Engineering, Madurai, Tamil Nadu, India

[3]Assistant Professor, Dept. of Computer Science and Engineering, Thiagarajar College of Engineering, Madurai, Tamil Nadu, India

---***---

Abstract - *If a human thinks that the number of cyber attacks are increasing, then he/she is not wrong. In fact, it's probably worse than he/she realizes. Cyber attacks are a major threat faced by all organizations. They infect computer networks by extracting the information of the organizations. In order to secure information systems, organizations must be prepared for predicting attacks. The information security analyst, however, focuses on identifying the technological weakness but does not understand the motives of the adversaries and their variance in the process of attack. Predicting the attacker behaviour, victim machine behaviors and the consequences of attacks against physical systems has become a part of risk management. By doing so, an individual machine could be capable of detecting possible cyber-attacks in their early phases so that the defense action can be done before the system gets compromised. This paper proposes a model for generating the profile of an attacker and victim machine by self-analyzing based on the characterization of the system behaviors and responses so that an individual machine can recognize itself whether it under DDoS attack, identify the type of attack and predict whether it is made as a compromised attacker or an end victim.*

Key Words*: *DDoS attack variants; behavioral traces; attacker machine profile ; victim machine profile.

1. INTRODUCTION

Distributed Denial of Service attack (DDoS) is a form of attack where a lot of infected computers that are under the control of the attacker (zombie computers) are used to flood the victim machine directly or indirectly, with a huge amount of information and obstruct the victim services in order to prevent legitimate users from accessing them. There may be up to five components in Distributed Denial of Service (DDoS) attacks. The attacker/ master computer from where the attacks are initiated and the Victim/ Attacked server which comes under the attack are the two components that are always present. Presence of these two components makes it a Denial of Service attack (DOS). In between these two , there may be additional three components namely handlers/ controlling computers, zombies/ agents, reflectors/amplifying network which make it a Distributed Denial of Service attack. Handlers / controlling computers issue instructions to the zombies / agents. Zombies / agents are the computers from which the DDoS attacks are carried out. In some instances, they may be infected computers of Internet surfing users who download certain malicious software, unintentionally authorize attackers to control their systems as compromised attack sources. Reflectors/amplifying network amplifies the number of requests that arrive from zombies, and sends the amplified requests to the victim servers to disable it's service.

The real intruder is difficult to trace because many innocent user's machines are used as zombies to carry out the attacks on the target machine. There are no fixed IP addresses for the zombie computers and even if some of the attacking zombie computers are detected and blocked, the attacker can always bring together more computers. Many zombie computers do not communicate directly with the victim machine. Rather they spoof the victim machine's IP address and send requests to huge number of reflector computers that may be infected. This IP spoofing makes it possible for the reflectors to send massive reply packets to victim machine, as they need to respond back to all the requests from what they believe is the source.

In the majority of cases, the owners of the zombie computers may not know that they are being utilized by attackers. Instead of making the web servers down completely, there is only a periodic flooding of web servers with huge traffic in order to degrade the service. Sometimes, DDoS attack choke the internet bandwidth used by the victim machine and cripple the resources like CPU,RAM, Buffer memory of victim machine. In order to avoid all these difficulties, it is essential for an individual machine to safeguard themselves by monitoring their CPU utilization, network bandwidth , memory utilization periodically. This job must be automatically ensured so that an individual machine able to identify their position whether it is an intruder or a victim machine and the type of attack. Thus the prediction of ongoing attacks using system behavioral traces helps the machine from causing any major damage to the system environment.

2. RELATED WORK

There exists different modes of attack execution and detection of DDoS attacks. Though the attack modes and steps may vary, the system response or system behaviour of an individual machine are always unique. In order to predict the ongoing attacks of an individual machine by itself, the system behavioural traces could be considered.

Behaviour aided intruder testimony technique

have been proposed in [1] which model and predict attacker's intention. The type of DDoS attack and its intention could be identified in early stages. The underlying assumption in this strategy is that the collected user information could be converted into attacking profile .This attacking profile will more accurately predict the attacker's goals and tactics. This method needs to deal with an enormous amount of log data.

Detecting DDoS attacks using machine learning algorithms[2] tells that naïve bayes can have a better edge than other conventional methods and it can outperform the other classification methods such as J48 and random forest methods in terms of accurate decision making. They have focused on creating a real-time dataset taking into account the different features reported in the KDD benchmark dataset, validating the dataset and identifying the appropriate classification method.

Network risk management using attacker profiling[5] hypothesize that there is a relationship between network action sequence and attacker behavior and this relationship can be used to evaluate and manage network risk. They have described a five step detection and risk estimation model which uses graphs of attack and behaviors of attackers. This method proposes that the reduction of the risk to the network can be accomplished by patching priority locations that can be exposed by optimizing the before and after risk probabilities.

A review of anomaly based intrusion detection system[4] works out the foundations of the main anomaly based network intrusion detection technologies along with their operational architectures and also provides a classification based on the type of processing relevant to the target system's behavioral model. Anomaly based approach generates a reference profile of normal system functions, network activity and data. Any activity deviating from baseline will be viewed as an attack. It has high predictive ability to detect novel attacks.

Choosing parameters for detecting DDoS attack[3] comes up with an easy solution to DDoS source end detection. They have proposed towards making CUSUM algorithm more accurate by defining the most suitable parameter values. This helps in making the detection more precise.

3. PROPOSED METHODOLOGY

In this proposed approach, the data about resource usage of the individual computer system is collected .The collected data is analysed and submitted for comparing with the threshold values.Upon comparison using anomaly based approach, it is determined that the individual machine is in safe state or an attacker or victim. Naive bayes classification algorithm could be used for detecting the type of attack in intruder and victim machine. Figure 1 represents the proposed methodology.

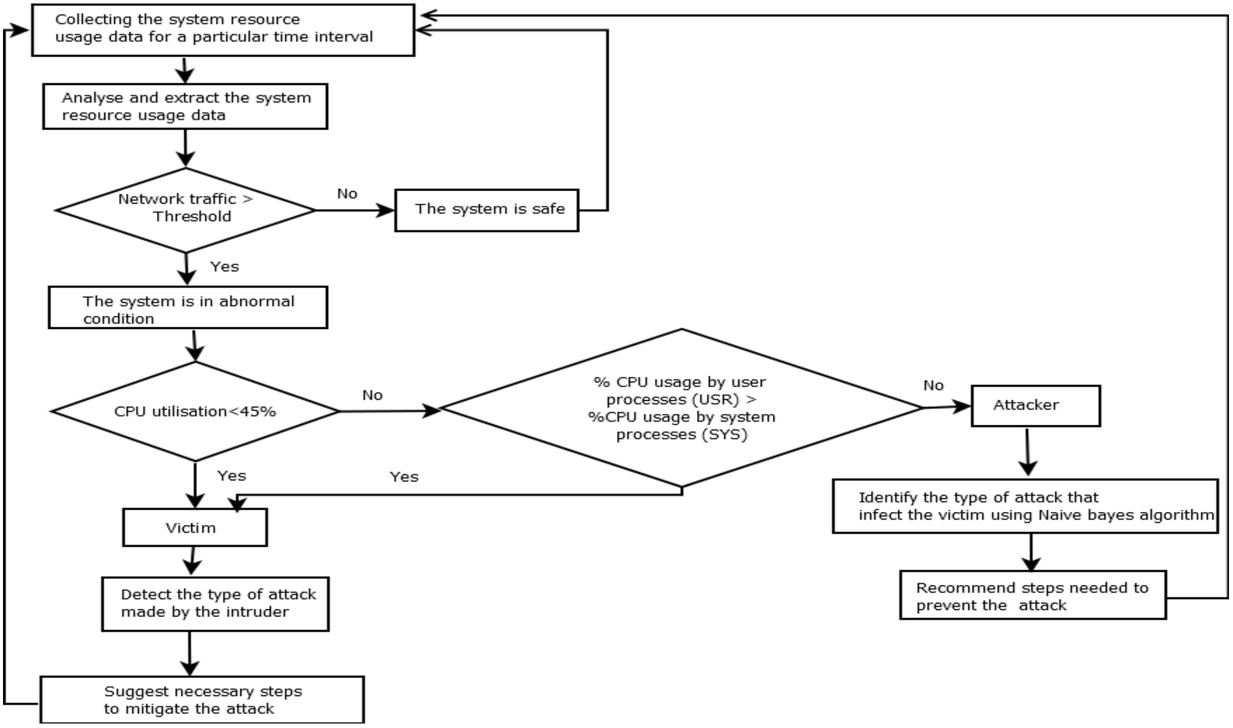
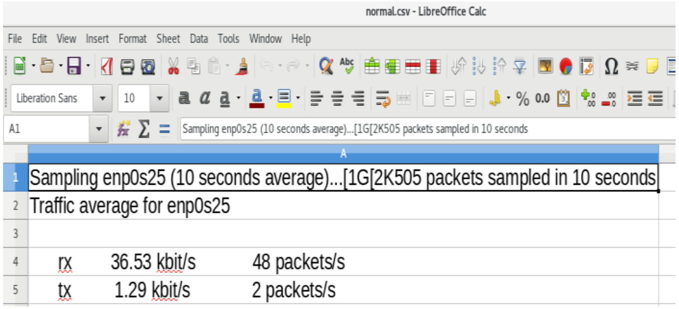

Collecting the system resource usage data for a particular time interval
Analyse and extract the system resource usage data
Network traffic > Threshold
No
The system is safe
Yes
The system is in abnormal condition
CPU utilisation<45%
No
% CPU usage by user processes (USR) > %CPU usage by system processes (SYS)
No
Attacker
Yes
Yes
Victim
Identify the type of attack that infect the victim using Naive bayes algorithm
Detect the type of attack made by the intruder
Recommend steps needed to prevent the attack
Suggest necessary steps to mitigate the attack

Fig – 1 : Prediction overview

3.1 Collection of data about system resource usage:

System resources include aspects of CPU utilisation, memory utilisation, swap utilisation, disk and network traffic. Data about system resource usage help us to determine the hardware resources that are underused and the applications that use many resources. When a DDoS attack hits a server machine, it can experience performance issues or completely crash the server by overwhelming the server machine resources including CPU, memory or even the entire network. The purpose of this type of attack is to overload the bandwidth of the network and cause utilization problems in the CPU or IOPS(Input/Output Operations Per Second).So collection of data about system resource usage is necessary for predicting the abnormality in the system. Mostly, the CPU utilization and network bandwidth become abnormal during an attack. The network bandwidth data is collected in comma separated value (csv) file using **vnstat** tool as shown in figure 2.


normal.csv - LibreOffice Calc
File Edit View Insert Format Sheet Data Tools Window Help
Liberation Sans
10
A1

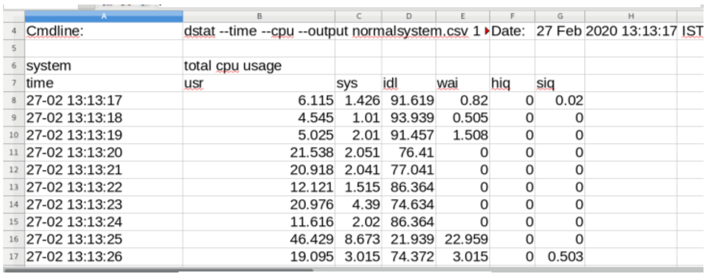
```
Sampling enp0s25 (10 seconds average)...[1G[2K505 packets sampled in 10 seconds
Traffic average for enp0s25

    rx      36.53 kbit/s        48 packets/s
    tx       1.29 kbit/s         2 packets/s
```

Fig – 2 : Network traffic data

The values in figure 2 are the average of received and transmitted traffic for a particular time interval.

The CPU utilization data is collected in csv file using **dstat** tool as shown in figure 3.



	A	B	C	D	E	F	G	H
4	Cmdline:	dstat --time --cpu --output normalsystem.csv 1				Date:	27 Feb 2020 13:13:17 IST	
5								
6	system	total cpu usage						
7	time	usr	sys	idl	wai	hiq	siq	
8	27-02 13:13:17	6.115	1.426	91.619	0.82	0	0.02	
9	27-02 13:13:18	4.545	1.01	93.939	0.505	0	0	
10	27-02 13:13:19	5.025	2.01	91.457	1.508	0	0	
11	27-02 13:13:20	21.538	2.051	76.41	0	0	0	
12	27-02 13:13:21	20.918	2.041	77.041	0	0	0	
13	27-02 13:13:22	12.121	1.515	86.364	0	0	0	
14	27-02 13:13:23	20.976	4.39	74.634	0	0	0	
15	27-02 13:13:24	11.616	2.02	86.364	0	0	0	
16	27-02 13:13:25	46.429	8.673	21.939	22.959	0	0	
17	27-02 13:13:26	19.095	3.015	74.372	3.015	0	0.503	

Fig – 3: CPU utilization data

The values in figure 3 are all the percentages of the total CPU time.

- *usr*: Percentage of CPU usage by user processes.
- *sys*: Percentage of CPU usage by system processes.
- *idl*: Percentage of CPU usage by idle processes.
- *wai*: Percentage of CPU usage waiting for input or output.
- *hiq:* Percentage of CPU usage by hardware interrupt.
- *siq:* Percentage of CPU usage by software interrupt.

3.2 Fixing threshold values:

To detect a suspicious flow on the network, the threshold values to be used in the detection algorithm must be identified first. The values can be determined by running extensive simulations on sample flows, analyzing the system resource usage and monitoring the results in the presence and absence of attacks.

Algorithm 1 :

1.1: Initialize sampling period T and time window size t.
1.2: Construct sample flows of legitimate traffic after Every t on individual machine for different cases.
1.3: Record the received network traffic rx and transmitted network traffic tx for each flow.
1.4:Record for different time intervals and this range be R1 for received traffic,R2 for transmitted traffic.
1.5: Repeat steps 1.2,1.3 and 1.4 for attack traffic and let the range be R3 for received traffic,R4 for transmitted traffic.
1.6: Choose appropriate threshold value T1 that must be larger than the largest value of R1 and smaller than smallest value of R3.
1.7: Choose appropriate threshold value T2 that must be larger than the largest value of R2 and smaller than smallest value of R4.

Fig – 4 : Algorithm for the identification of threshold values T1 and T2

Algorithm 2 :

2.1: Initialize the sampling time T and time window t.
2.2: Create sample flows of legitimate traffic on individual machine after every t.
2.3: Calculate CPU utilization for each flow using Eq (1); let it be C1,C2,….Cn.Calculate average CPU utilization AvgC. AvgC=C1+C2+….Cn/n.
2.4:Record for different time intervals and this range be R5.
2.5: Repeat steps 2,3 and 4 for attack traffic and let the range be R6.
2.6: Choose appropriate threshold value T3 that must be larger than the largest value of R5 and smaller than smallest value of R6.

Fig – 5 : Algorithm for the identification of threshold value T3

To design the threshold, the following assumptions are made:

i. The threshold values that are calculated using algorithms 1 and 2 are used in the detection algorithm directly.

ii. The values of sampling period T and time window size t are taken as 10s and 1s respectively.

iii. The detection algorithm can be implemented on the individual computer as a part of the defence agents.

The threshold value T1 is used to determine the abnormal network flow in the system. It can be determined by creating a sample flow of legitimate traffic and estimating the network bandwidth. For the given time window, the value of the network band with is recorded for different cases like during browsing, video call, live streaming, uploading and downloading and during DDoS attacks like TCP SYN flood attack, ICMP flood attack and UDP flood attack.

The threshold value T3 is used to determine whether the individual machine is an attacker or a victim. It can be determined by measuring the system performance in terms of CPU utilization. For a given time window, the value of CPU utilization is recorded before attack for a normal system. The same process is repeated for attacker and victim machine during attack traffic . CPU utilization is the sum of work performed by Central Processing Unit. It shows the percentage of workload on a processor that indicates if any improvements are to be made to the device otherwise the capacity may get exhausted. The CPU utilization for a given time period is calculated as:

CPU utilization= 100% - (% of time spent in idle task) **Eq (1)**
(in %)

The threshold values play an essential role in the detection algorithm. The next step is to design the attack detection algorithm. The main task of the detection algorithm is to identify the suspicious flow and later confirm whether the individual machine is an attacker or victim and detect the type of DDoS attack.

3.3 Detection algorithm:

Initially, the various flows are constructed and their received network traffic, transmitted network traffic, system CPU utilization are calculated for every time window t. If the value of received network traffic and transmitted network traffic is less than T1 and T2 respectively then it indicates that the system is in safe state. Otherwise, the system is under attack. If the system is under attack, the values of CPU utilization is compared with T3.If the value of CPU utilization is less than threshold T3 then the system is a victim. Otherwise, compare the average CPU usage by user processes and system processes. Here these two CPU usage data are considered because in victim machine, the CPU usage by system processes are smaller than the CPU usage by user processes. The CPU usage by user processes are more due to the background running applications. If the average CPU usage by system processes is larger than the average CPU usage by user processes then the system is attacker. Otherwise, the system is victim.

Average CPU usage by = $SYS_1+SYS_2+....+SYS_n$ / T **Eq (2)**
system process (SYS)

Average CPU usage by = $USR_1+USR_2+....+USR_n$ / T **Eq (3)**
user process (USR)

The detection algorithm works by implementing various steps described in figure 6.

Algorithm 3 :

```
3.1: Initialize various parameters such as time
     window, sample time t, detection thresholds
     T1,T2and T3.
3.2 : Construct flows after every t on individual
     machine.
3.3 :Calculate received network traffic and
     transmitted network traffic and let it be rx and tx
     respectively.
3.4 : If (rx >T1 AND tx>T2) OR (rx>T1 AND tx<T2)
     OR (rx<T1 AND tx>T2),then the flow is
     suspicious;Otherwise the flow is not
     suspicious.
3.5 : To further detect whether the system is
     victim or an attacker,calculate the CPU
     utilization U of the system using Eq (1).
3.6 : Compare U with threshold T3.If U<T3,then
     the system is victim. Otherwise, calculate the
     values of average CPU usage by system
     processes(SYS) using Eq (2) and average CPU
     usage by user processes(USR) using Eq (3).
3.7 :If SYS > USR,then the system is victim.Otherwise it is an
     attacker.
```

Fig – 6: Algorithm for detecting the profile of the system

4. SYSTEM IMPLEMENTATION

This section outlines experiments performed to detect ongoing DDoS attacks.

4.1. DDoS attacks:

DDoS attacks like TCP SYN flood, ICMP flood, and UDP flood are created with the help of hping3 tool in Linux. The number of malicious packets are generated by hping3 tool from several attacker machines to target several victim machines. The attack environment consists of one attacker, two compromised attacker and four victims as shown in figure 7.

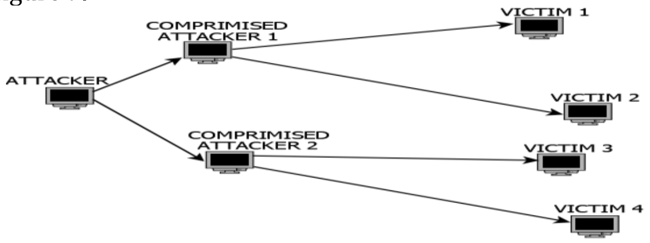


Fig – 7: Attack environment

4.2 Data collection:

The system resource usage data are collected using Dstat tool in every individual machine. Dstat is a powerful tool for generating system resource statistics. The network traffic data are collected using Vnstat tool. Vnstat is a console based network traffic monitor for linux that keeps a log of network traffic for the selected interface. These tools allow the data to be written directly to a csv file. Using python code, the collected data is analysed and extracted.

4.3 Fixing the threshold values:

The extracted data such as received and transmitted network traffic are compared with threshold values. Various analysis are done for fixing the network traffic threshold values. The table 1 given below represents the network bandwidth values of the system that is not undergoing or experiencing any attack for the time window 1s and sampling time 10s.Similarly, the network traffic data for different time intervals are collected. The threshold values T1 and T2 are identified as shown in figure 4.

Table -1: Network bandwidth values of the safe system

Cases	Received network traffic (rx)	Transmitted network traffic (tx)
In normal state	36.53Kbit/s	1.29Kbit/s
During browsing	799.84Kbit/s	61.70Kbit/s
During Upload/Download	3.43Mbit/s	900.78Kbit/s
Video call/live streaming	5.34Mbit/s	6.55Mbit/s
All cases(browsing, upload/download, video call/live streaming)	20.07Mbit/s	20.23Mbit/s

From table 1,the highest value of received network traffic (rx) and transmitted network traffic (tx) of the system that is not undergoing or experiencing any attack are 20.07Mbit/s and 20.23Mbit/s.Let it be R1 and R2 respectively.

The table 2 shows the network bandwidth details of attacker machine and victim machine for time window 1s and sampling time 10s.

Table -2 : Network bandwidth value of the unsafe system

Cases	Received network traffic (rx)	Transmitted network traffic (tx)
Attacker machine generating TCP SYN flood	1.75Kbit/s	60.14Mbit/s
Victim machine experiencing TCP SYN flood	55.17Mbit/s	59.23Mbit/s
Attacker machine generating ICMP flood	3.73Mbit/s	43.38Mbit/s
Victim machine experiencing ICMP flood	30.27Mbit/s	32.33Mbit/s
Attacker machine generating UDP flood	1.26Mbit/s	37.92Mbit/s
Victim machine experiencing UDP flood	30.49Mbit/s	33.27Mbit/s

Mostly, the victim transmits and receives too many packets and the attacker simply transmits. It may not receive the packets from the victim because of spoofed ip address. Consider the victim machine while fixing the range for received network traffic of the system under attack .This is because the victim machine receives too much of packets. From table 2,the smallest value of received network traffic (rx) of the victim machine under DDoS attacks is 30.27Mbit/s. Let it be R3.The smallest value of the transmitted network traffic (tx) of the system that is undergoing or experiencing DDoS attacks is 32.33Mbit/s and let it be R4.The threshold value of the received network traffic T1 must be between 20.07Mbit/s(R1) and 30.27Mbit/s(R3) and the threshold value of the transmitted network traffic T2 must be between 20.23Mbit/s(R2) and 32.23Mbit/s(R4).The threshold values T1 and T2 are fixed as 25 Mbit/s and 25Mbit/s respectively. The threshold value T3 is identified as shown in figure 5.

The threshold value for average CPU utilization must be identified. The sampling time T and time window t are initialized as 10s and 1s respectively. The sample flows of legitimate traffic on individual machine after every t is recorded. The CPU utilization data is collected in csv file as shown in figure 2.The average CPU utilization for the system under normal state is identified as 23.18%.Similarly,for different time intervals like 5s,15s,20s the average CPU utilization are identified as 20.45%,35.66%,27.29% and let this range be R5.These values are calculated using Eq (1) and (2).Likewise,the average CPU utilization for TCP SYN flood attack ,ICMP flood and UDP flood are calculated as 54.37%,52.58% and 48.8% respectively using Eq (1) and (2).Let this range be R6.The threshold value T3 must be larger than the largest value of R5 and smaller than the smallest value of R6.So T3 must be between 35.66% and 48.8%.Thus the threshold value T3 is fixed as 45%.

5. RESULTS AND DISCUSSION

The profile of the system as attacker or victim is predicted as given in figure 6 continuously. The system resource usage data and network traffic data are collected for an individual machine using python and various tools like dstat, vnstat to self analyse for detection. The threshold values for received network traffic T1,transmitted network traffic T2 and average CPU utilization T3 are identified .

Now to test the detection success rate, a attack window of 6 hours was conducted with four attackers with instruction to initiate any of the three DoS attack variants and record their attack duration for validation process. Based on the proposed behavioral model, the threshold values T1,T2 and T3 were identified as 25Mbit/s, 25Mbit/s and 45%.Initially,the collected data for an individual machine such as received network traffic data (rx) is compared with T1 and transmitted network traffic data (tx) is compared with T2 for the different observed instance as shown in table 3.

Case1 : If rx>T1 and tx>T2,then the system is unsafe.
Case2 : If rx>T1 and tx<T2,then the system is unsafe.
Case3 : If rx<T1 and tx>T2,then the system is unsafe.
Case4 : If rx<T1 and tx<T2,then the system is safe.

Table -3 : Network bandwidth values of the observed system

S No	Time window	rx	tx	Status
1	15 – 03 13:13:17 to 13:15:23	41.56 Mbit/s	39.67 Mbit/s	Unsafe
2	15 – 03 13:19:02 to 13:20:02	21.29 Mbit/s	17.83 Mbit/s	Safe
3	15 – 03 13:23:07 to 13:23:57	37.66 Mbit/s	16.59 Mbit/s	Unsafe
4	15 – 03 15:10:12 to 15:10:42	15.78 Mbit/s	14.65 Mbit/s	Safe
5	15 – 03 15:12:18 to 15:13:28	23.72 Mbit/s	40.12 Mbit/s	Unsafe
6	15 – 03 16:25:35 to 16:25:55	35.8 Mbit/s	10.52 Mbit/s	Unsafe

If the system is identified as unsafe,then the average CPU utilization data for an individual machine is calculated. It is compared with threshold value T3 (45%) for different observed unsafe instances as shown in table 4.If average CPU utilization is less than 45%, then it is considered as a victim. Otherwise, calculate the average CPU usage by system processes (SYS) and average CPU usage by user processes (USR) of the individual machine and then they are compared. If the average CPU usage by system processes (SYS) is greater than the average CPU usage by user processes (USR), then the profile of the individual system is recorded as attacker. Otherwise, it is a victim.

Table -4 : CPU utilization values of the observed system

Unsafe instant	Avg CPU	SYS	USR	Profile
1	35.65%	-	-	Victim
3	76.79%	52.8%	20.6%	Attacker
5	65.31%	11.04%	51.3%	Victim
6	40.32%	-	-	Victim (Burst traffic)

In unsafe instant 6,it is observed that the victim machine experienced burst traffic. During continuous monitoring it is identified that the system is unsafe for the first 10s and then for the next 10s the system changed to safe state. Hence, it is considered as a burst traffic which exists for a short period of time.

After predicting the profile of the system, the future task is to detect the type of DDoS attack using Naive bayes classification algorithm and to suggest necessary steps to mitigate the attack. The algorithm 3 is performed again and again for continuous monitoring of the system. The live network packet data of an individual machine is collected using TShark in pcap file. The features are extracted from pcap file. The training dataset formation is done and Naive bayes classification algorithm is applied. The classification predictor will predict the type of DDoS attack.The administrator or the system user can take necessary actions accordingly.

6. CONCLUSION

This paper introduces a model for generating the profile of an attacker and victim machine. The advantage is that there is no need for centralized network monitoring server for predicting the attacker and victim machine. Each and every individual machine self analyze using system behavioural traces and identify whether it is an attacker or a victim. As a result, an individual machine could be capable of detecting cyber-attacks in their early phases so that defence action can be done before the system gets compromised. Our future work is to detect the type of DDoS attack and suggest necessary steps to mitigate.

7. REFERENCES

[1] Mallikarjunan, K. N., Shalinie, S. M., & Bhuvaneshwaran, A. (2019). BAIT: behaviour aided intruder testimony technique for attacker intention prediction in business data handling. International Journal of Business Intelligence and Data Mining, 14(1-2), 177-198.
[2] Mallikarjunan, K. N.,Bhuvaneshwaran, A., Sundarakantham, K., & Shalinie, S. M. (2019). DDAM: Detecting DDoS Attacks Using Machine Learning Approach. In Computational Intelligence: Theories, Applications and

Future Directions-Volume I (pp. 261-273). Springer, Singapore.

[3] Pu, S. (2012, December). Choosing parameters for detecting DDoS attack. In 2012 International Conference on Wavelet Active Media Technology and Information Processing (ICWAMTIP) (pp. 239-242). IEEE.

[4] Jyothsna, V. V. R. P. V., Prasad, V. R., & Prasad, K. M. (2011). A review of anomaly based intrusion detection systems. International Journal of Computer Applications, 28(7), 26-35.

[5] Dantu, R., Kolan, P., & Cangussu, J. (2009). Network risk management using attacker profiling. Security and Communication Networks, 2(1), 83-96.

[6] Mallikarjunan, K. N., Muthupriya, K., & Shalinie, S. M. (2016, January). A survey of distributed denial of service attack. In 2016 10th International Conference on Intelligent Systems and Control (ISCO) (pp. 1-6). IEEE.

[7] Mallikarjunan, K. N., Shalinie, S. M., Sundarakantham, K., & Aarthi, M. (2019). Evaluation of security metrics for system security analysis. In Computational Intelligence: Theories, Applications and Future Directions-Volume I (pp. 187-197). Springer, Singapore.

[8] Mallikarjunan, K. N., Prabavathy, S., Sundarakantham, K., & Shalinie, S. M. (2015, December). Model for cyber attacker behavioral analysis. In 2015 IEEE workshop on computational intelligence: theories, applications and future directions (WCI) (pp. 1-4). IEEE.

[9] Mallikarjunan, K. N., Shalinie, S. M., & Preetha, G. (2018). Real Time Attacker Behavior Pattern Discovery and Profiling Using Fuzzy Rules. Journal of Internet Technology, 19(5), 1567-1575.

[10] Shalinie, S. M., Kumar, M. M., Karthikeyan, M., Sajani, J. D., Nachammai, V. A., Sundarakantham, K., & Mallikarjunan, K. N. (2011, June). CoDe—An collaborative detection algorithm for DDoS attacks. In 2011 International Conference on Recent Trends in Information Technology (ICRTIT) (pp. 113-118). IEEE.

[11] Prasad, K. M., Reddy, A. R. M., & Rao, K. V. (2014). DoS and DDoS attacks: defense, detection and traceback mechanisms-a survey. Global Journal of Computer Science and Technology.

[12] Kshirsagar, D., Sawant, S., Rathod, A., & Wathore, S. (2016). CPU load analysis & minimization for TCP SYN flood detection. Procedia Computer Science, 85, 626-633.

[13] Steel, C. M. (2014). Idiographic Digital Profiling: Behavioral Analysis Based On Digital Forensics. Journal of Digital Forensics, Security and Law, 9(1), 1.

[14] Murali, A., & Rao, M. (2005, August). A survey on intrusion detection approaches. In 2005 International Conference on Information and Communication Technologies (pp. 233-240). IEEE.

[15] K., & Mallikarjunan, K. N. (2011, June). CoDe—An collaborative detection algorithm for DDoS attacks. In 2011 International Conference on Recent Trends in Information Technology (ICRTIT) (pp. 113-118). IEEE.