Denial of Service Attacks Detection Based on Available Bandwidth Estimation

A. Niño, C. Guerrero

***Abstract* - Denial of Service Attacks have had an increasing sophistication that demands new solutions from the scientific community. Intrusion Detection Systems use regular detection techniques by inspecting IP packet headers among different rules for each type of attack. A simpler approach presented in this paper is based on the available bandwidth estimation as a metric to determine the unexpected variation in the bandwidth consumption as an indicator of a possible DoS attack. Using a modified version of the tool Traceband, we show that is possible to identify different types of flooding attacks implementing criteria according to the states of the Markov Hidden Model used to establish available bandwidth estimation values.**

***Keywords* - DoS, Available bandwidth estimation, detection, Denial of Service Attacks, network monitoring, Traceband.**

I. INTRODUTION

SUCCESSFUL Denial of Service (DoS) attacks not only affect the availability of a networking system, but also effect the reputation of the institution attacked; which conveys the general public perception of uncertainty concerning the application or web service violated. The economic impact can be even worse in the case of unavailability of a website where business transactions, such as an online banking or e-commerce, generate significant economic losses from transactions not carried out during the period of time that lasted the fault. The costs of recovering from DoS attacks can be thousands of dollars per day. According to a 2011 study by the Ponemon Institute, only in the United States the cost of cyber attacks could go from 1.5 million to 36.5 millions of dollars every year. It can also have political and social implications, such as lack of transparency in cases like Wikileaks or timely information from voting results. [1].

This paper aims to develop a warning system for Denial of Service Attacks based on the detection of variations in the available bandwidth. Testing is done using a controlled networking environment with only one type of Denial of Service Attack, as many of them differ only by IP protocol vulnerabilities to overload the system.

To develop an alert system for Denial of Service Attacks based on variations in the available bandwidth on the network, it was initially required to establish some criteria from the bandwidth consumption point of view to generate an alert when facing a possible DoS violation. That criteria represented in some rule is implemented in an available bandwidth estimation tool and tested over an Internet emulation testbed to evaluate its effectiveness.

The rest of the article is organized as follows. The following section includes a related work about DoS detection using available bandwidth estimation. Section III defines the available bandwidth approach and some of the tools presented in the literature to perform the estimation. Section IV describes the testbed and the methodology utilized in the experiments. Then, the criteria defined to generate the alerts and implementation details are shown in section V. Section VI shows results from the experimentation. Finally section VII concludes the paper.

II. RELATED WORK

Early detection of denial of service attacks is very important for many online applications, community services, e-commerce services and entertainment services, among many others. The study of these attacks include the categories of flooding, identified to prevent legitimate users from accessing systems as distributed attacks, including the infiltration of a large number of computers often taking advantage of their software vulnerabilities to launch attacks simultaneously from different sources [1]. To evaluate defense systems against denial of service attacks there are theory, simulation and emulation, also analyzed by the same authors.

When talking about emulation as measurement method, normally a emulation testbed is used, enabling to include elements such as network topology, traffic flow of the attack, and dissemination of data in a controlled test environment where you can evaluate using different networking scenarios [2]. Several works on distributed denial of service attacks (DDoS) use different testbed technologies such as DETER, Emulab, and Planetlab [3]. In [4] author's study has shown that in the case study of TCP denial of service attacks factors like CPUs, buses devices and devices drivers may create a bottleneck that simulators do not model.

This work based the DoS attack detection on variations in the available bandwidth. This requires knowing how the available bandwidth is estimated. Currently, there are several techniques and tools to obtain this estimation. Some of these tools are Pathload, Pathchirp, Spruce, IGI, and Abing [5]. Various tools have been evaluated based on variables such as the error rate, overhead, time, and reliability, and on the following factors: Capacity of the link propagation delay, packet loss rate, percent of traffic, and the packet's size. Another tool used in this paper to implement the DoS warning system is called Traceband [6]. According to the authors and when comparing the performance of the tool against Spruce and Pathload, Traceband is as accurate as them, but it is faster. The speed rate of the estimation is a

A. Niño, Universidad Autónoma de Bucaramanga, Bucaramanga, Colombia, anino337@unab.edu.co

C. Guerrero, Universidad Autónoma de Bucaramanga, Bucaramanga, Colombia, cguerrer@unab.edu.co

critical factor since it is intended to alert as soon as a possible about denial of service attack occurs.

Relatively few works have been done directly on the detection of DoS attacks based on the estimation of available bandwidth services. These works are very interesting exhibits on detection of abnormalities based on bandwidth estimation using for example tools like PQLink [7] to allow end users to measure the estimated bandwidth of arbitrary links on a network. This tool uses a technique called trains of packet-quartets that only needs a single point for the measurement. The authors perform simulations to validate the efficiency of the tool and the anomaly detection approach.

There is other research made in Singapore to detect Distributed Denial of Services Attacks. This work is based on the detection of anomalous information and uses an algorithm to detect the attacks. They use simulations to verify the method and generate traffic with tracer [8].

In [9] another technique is presented, through TCP Congestion Window Analysis and CUSUM (Cumulative Sum). The authors develop a method of detecting TCP flooding based DoS. For testing the network simulator NS2 is used. The implementation is argued to have low cost because the congestion window, as parameter, is calculated by TCP and the CUSUM algorithm has low resource requirements. Although time detection is better than other methods, it takes several seconds to discover the attack.

Among the techniques used for detecting Flooding type of DoS attacks vary from profiles, change point detection, wavelet analysis, data mining, and fuzzy logic signals, none of these techniques working separately is fully effective to properly identify attacks due to false positives [10]. If the aim is higher effectiveness than that obtained by each of the techniques discussed, it may be possible that integration of several techniques could be required to accomplish it.

III. AVAILABLE BANDWIDTH ESTIMATION

The available bandwidth of an end-to-end path is a time-varying metric defined as the minimum of all non-utilized link capacities throughout the communication path [5]. To estimate the available bandwidth in a network, probing packets are sent and the delays due to network traffic are analyzed. There are several tools to measure the available bandwidth - ABETs (acronym for Available Bandwidth Estimation Tools) in an end-to-end path. Some of them such as Spruce, Pathload, IGI and Pathchirp, are evaluated in [5].

ABETs can mainly belong to the Probe Rate Model (PRM) or to the Probe Gap Model (PGM). The first of them introduce network traffic from the receiver to the transmitter, analyzing the One Way Delay generated by congestion in the path. Among them are tools such as TOPP, Pathload, Abget and pathchirp. Estimation tools based on the PGM, analyze the relationship between the input gaps and the output gaps on the link. Some tools of this type are Traceband, Spruce, Delphi, and IGI [8].

The tool used in this paper due to the fast estimations it performs is Traceband [6]. This end-to-end available bandwidth estimation tool utilizes a hidden Markov model (HMM) approach and the Probe Gap model (PGM). The process of estimation can be affected by many factors like the nature of the cross traffic, the load of the network, network interface card, router o host errors, among others. For that reason, a single pair of packets can't be used to accurate measure the available bandwidth. That is why all the mentioned ABETs uses a train of packets, generating as well some load on the link. In the case of Traceband, this load is less than 5% of the link capacity.

Every pair of probing packets in Traceband has an amount of time difference between them and another time difference with the next pair. The time difference in the pair on the sender is called input gap and on the receiver is called output gap. If there is no traffic on the link, the input gap and the output gap will be the same for each pair. If the output gap is bigger, it is due to traffic packets between them. According to the hidden Markov model these differences can be modeled by N states from Low to High, representing a level of availability. Each pair of packets generates an observation that corresponds to one of these states. From the mean of all the states can be statistically an estimation of the available bandwidth. Traceband authors find by testing that the best results were found with 10 states from Low to High.

Traceband runs several estimations, according to the initial parameters set on the client application. In the first estimation more pairs of packet are send to the server, and using the HMM the tool is able to learn the available bandwidth dynamics, allowing to use less pair of packets on the following estimations.

IV. METHODOLOGY

This section describes the network infrastructure utilized and the use of other tools to perform the experiments and evaluation of the effectiveness of the DoS detection.

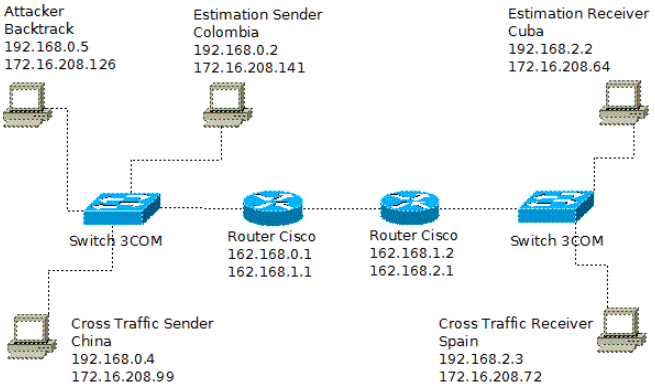

Fig. 1 Testbed diagram.

A. *Testbed specifications*

The testbed mimics the behavior of an end-to-end transmission between two nodes. As it is shown in Figure 1, the testbed has the following elements: Five computer nodes connected through subnets 192.168.0.0/24 and 192.168.2.0/24. On the first subnet, a node called Colombia sends probing packets to estimate the available bandwidth. The node China sends synthetic traffic to emulate real Internet traffic. Finally, a Backtrack node is responsible for conducting the attack. In the second subnet a node called Spain receive the probing packets to estimate the available bandwidth. Another node called Cuba receives the synthetic

traffic and the denial of service. At each end of the subnet are a Switch and Router for communication between them.

All machines use Linux as operating system and have two network cards in order to make a one-off exercise and not affect the operational network with synthetic traffic and denial of service attacks that are executed during testing.

B. *Traffic generation*

To generate synthetic traffic, the Multi generator MGEN traffic tool, develop by the US Naval research Laboratory, is used. This is an open source software that allows to generate different types of traffic distributions such as Periodic, Poisson, and burst [11].

C. SYN Flooding Attack

Denial of Service Attacks are those that prevent a legitimate user to use a computer or network resource. Both internal and external users of a network can cause these attacks. Similarly, they can be malicious or not, as when users abuse the system resources occupied bandwidth in video file downloads, etc.

Within those malicious attacks where intruders gain illegitimate access to the system, it usually include the falsification of the source address of the attacks to avoid being traced. Such attacks are platform independent because they use the IP protocol that connects the machines regardless of the topology. The mechanism of attack can exploit vulnerabilities of critical systems or consume network resources [12].

SYN Flooding is a type of attack that intentionally does not complement the TCP handshaking protocol. With a large number of connections that are waiting to establish communication, the machine runs out of memory, achieving a denial of all services until the end of the timeout connections. This type of attack, as well as others, uses a significant amount of network resources to prevent legitimate users to use the channel. This is the reason why it's the main case of study during this work.

The tool Hping3 can be used to generate among others the SYN Flooding type of attack using its -S flag, however this attack is no longer useful since computer resources are bigger every day. To generate a visible attack is necessary to use as well the –faster flag.

Under different traffic conditions the Traceband ABET tool was used to analyze the behavior under a SYN flooding attacks:

With no traffic transmission, available bandwidth estimated using the Cuba Machine as the Server and the Colombia machine as the Client, is the same as the tight link, in this case 100Mb. During each estimation, the observations T remained stable at maximum state, in this case: 10. The first estimation took more pairs of packets in this case: 60, meanwhile the algorithm learns and the next estimations require less pair of packets, 40 to perform each estimation.

Using machine Spain to listen to the traffic transmission generated with MGEN from China machine the behavior or available bandwidth varies depending on the distribution used (Periodic, Poisson or Burst), the amount of packets send per second, and the size of the packets. For example using 3810 packets of 982 bytes, a 30Mb (30%) load is generated. During each estimation the state observations varies from 10 to 1 according to the load over the net.

Once the Traceband tool is working, the Backtrack machine is used to perform attacks over the Spain machine, observing each estimation was possible to determinate the criteria to detect that an attack had started.

V. DETERMINING THE CRITERIA

From Attacks can be determined the following criteria for the identification of denial of service attacks:

1. Loss in the number of observations. That is, lost in the pairs of valid packets arriving at the receiver to perform the estimation. This is explained by the network congestion, as the attack takes more time in execution will be longer the number of packets that do not reach its destination, it can also be observed with the results of the ping command.

2. No return to the state of available channel. If the state of higher bandwidth available is generated when the time difference (gap) between the departing packets is the same as when they reach the receiver, this means that there are packets in transit during this period. In normal traffic, there are states where there is no traffic between packages. In a denial of service attack there are always packages between the pair of each estimate, so the state do not return the one in the available channel again (state 10) until the TCP Congestion control reach the time out or 3 duplicate ACK to retransmit.

3. Finally, if the estimation is allowed to continue during the attack, it becomes successful denying network functions to the point that the estimation packets cannot be analyzed triggering the generation of a "Operation Timed Out" message from Traceband.

A. Location where must coding criteria selected be implemented

To create an alert the criteria must be codded into the server or Traceband receiver where estimation algorithm takes place. The analysis of denial of service attack function should be called where we already have the results of all the estimation observations so that it can be assessed the T states and the number of pairs of packets received. However, it also should be noted that this analysis must be made between estimations and do not wait until the end of program execution to alert within seconds from the attack start.

B. *Modifying the source code of the tool Traceband.*

The source code modification program is shown in figure 2.

The source code for the modification of the program is the call to CreateAlarm function written in C Language, the result was as follows:

Given that the input parameters are:

- T - Number of observations on the estimation.
- * ABobs - Contains the observation data.

• cnt * - Number of packets on the sender.

• N - Number of possible states according to the estimation.

```
//=============================================================
//= T is the number of observations
//= ABobs has the obsetvations: {timestamp, av_bw}
//= cnt has the number of packets from the sender
//= N has the number of states
//=============================================================
void CreateAlarm(int *cnt, int *ABobs, int T, int Nu){
    int snd_pck=0;
    int cnt_dwn_lmt=0;
    char* respuesta="Denial of Service (DoS) Attack Found";
    snd_pck=*cnt/2; //Send packets
    if (T<snd_pck){
        for( i=0 ; i < T ; i++ ) //
          {
              if (ABobs[i]< Nu){
                  cnt_dwn_lmt++;
              }
              else
              {
                  cnt_dwn_lmt=0;
              }
              if(cnt_dwn_lmt>=T/4){ //1/4 de observations
                      printf("%s \n", respuesta);
                      return;
              }
          }
    }
}
```

Fig. 2 Traceband Alert Source Code

The program analyzes if the number of pair packets that came from the sender are the same that should have arrived. Otherwise the first criterion, the packet loss, is being achieved and starts to analyze the second one.

For each one of the observations, if the number of the observation is less than the number of possible states, a counter is incremented. If equal, the counter resets. On each iteration the number of consecutive observations that not reached the highest state is analyzed, if it is at least a quarter of the total number of observations received meets the second criteria and can print the corresponding alert.

The number of observations required to generate the alert was modified several times according to the evaluation tests performed to obtain fewer omissions in issuing the alert. Finally if the alarm is generated the execution of the function is aborted.

C. *Programming alerts.*

Since the outputs resulting from the available bandwidth estimation tool are text-mode console, so the initial programming of the alerts will be the same.

Once the denial of service attack is detected by the criteria described above the program output indicate the existence of abnormal network traffic through the following message "Denial of Service (DoS) attack found".

VI. EVALUATION

In order to perform evaluation of the alarm function, a group of experiments was design.

Traffic conditions: without traffic, Periodic, Poisson and Burst distribution with 30% and 60% total load. 60% is too high for normal traffic conditions but is evaluated in order to discover false positives.

Attack type: without attack, SYN Flooding, TCP Flooding or UDP Flooding. Although this project was looking to alert SYN flooding attacks, evaluation for other possible attacks is performed.

Number of experiments: For each combination 10 experiments are performed. In total 7 traffic conditions * 4 attack types * 10 experiments for 280 analysis.

In the analysis of TCP and SYN Flooding Attacks a sample of 7 estimations for a total of 300 observations is taken for reference.

To detect attacks with increased traffic (60%) is necessary that Traceband have at least four estimations before the attack starts, this is 2 seconds approximately. For UDP Flooding attack detection at least ten estimations are required.

A. *Results by attack type*

It was found that the attack-detection was usually performed between 0.75 and 1.5 seconds after attack starts, this means at least one or two complete estimations on the Traceband program.

With SYN flooding attack type alarm was not shown in 5 of the 70 experiments to an effectiveness of 92.8%.

With TCP flooding attack type alarm was not shown in 4 of the 70 experiments to an effectiveness of 94.2%.

Unlike previous, with UDP flooding attack multiple failures in detection where shown, especially in the case of Poisson traffic at 30%, generating overflow in the majority of experiments. For other types of traffic was more recurrent the program to stop before analyzing the traffic, with this type of attack alarm failed in 41 of 70 cases, to an effectiveness of only 41.4%.

B. *Results by generated traffic conditions*

Without traffic, the alarm was not displayed in 1 of the 20 experiments with TCP and SYN flooding attacks, to an effectiveness of 95%. For UDP flooding attack an effectiveness of 30% occurred.

Figure 3 shows the behavior of the estimates for each type of attack when there is no presence of cross traffic on the network. States of 10 are observed at baseline, which means there is no network traffic, the first highlighted area indicates the beginning of each type of attack, then the states vary, but it stands, the presence of long periods in which states do not reach the maximum (10) and the short duration of UDP Flooding attacks compared with its peers TCP and SYN as shown in highlight areas in the bottom, where the end of the estimation is shown by the loss packets.

With 30% traffic (all distributions) failure occurred in 2 of the 60 experiments of TCP and SYN flooding attacks to an effectiveness of 96.7%. For UDP flooding attack an effectiveness of 50% occurred.

In Figure 4 can be seen the comparison of each of the attack types analyzed in the presence of 30% Poisson traffic.

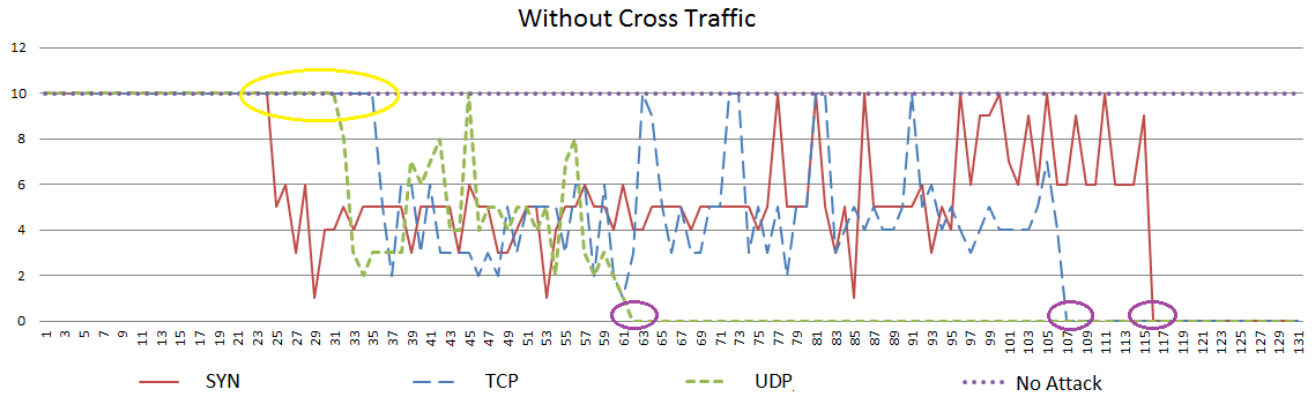

Fig. 3. Observations during attacks without cross traffic.

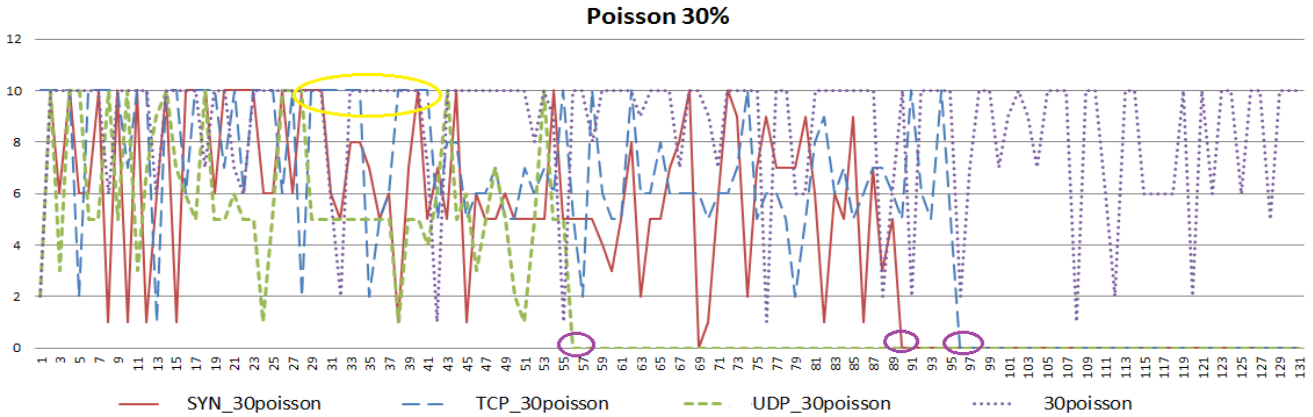

Fig. 4 Observations during attacks with Poisson traffic at 30%.

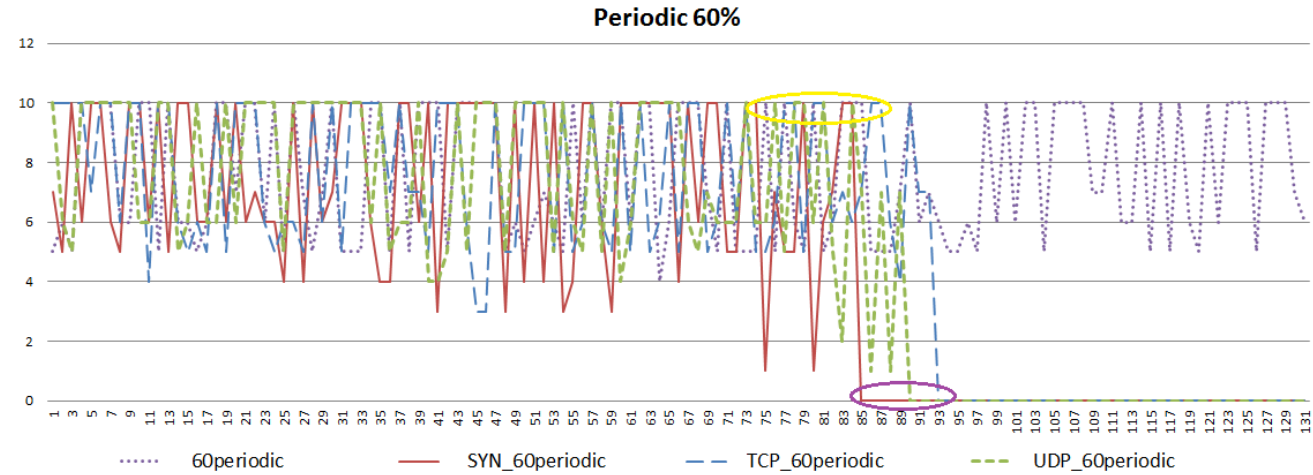

Fig. 5 Observations during attacks with Periodic traffic at 60%.

With 60% traffic (all distributions) failure occurred in 6 out of the 60 experiments of TCP and SYN flooding attacks to an effectiveness of 90%. For UDP flooding attack an effectiveness of 36.7% occurred. This result is consistent, since such a large traffic occasionally generated overflow on the estimation. Figure 5 shows an example of each attack in the presence of 60% periodic cross traffic, highlighting the states of low bandwidth available at the end of the detection of each one of the attacks.

C. *Special Considerations*

Overall failures that may occur for the alert message not been displayed on the Traceband receiver are:

a. The attack is so strong that the pair packets do not reach the receiver and do not met the first two criteria screening for lack of information (58% of failures). The fact that the message "Operation timed out" appears is already a warning, but isn't always an attack, it can also be displayed by a shutdown of the host machine of the estimation.

b. The number of consecutive observations less than 10 does not reach a quarter of the total observations, this is, does not meet the second criteria in any of the estimations. It may be possible that if the pairs continue coming for analysis, would be detected, however in this case the "Operation timed out" occurs earlier. (28% of failures).

c. Overflow. When an observation calculated bandwidth is 0, the algorithm in the next iteration starts showing errors in the generation of the matrices A and pi, showing negative values or nan (Not A Number), the result of this estimation is not reliable. (14% of failures).

In all experiments with attack, detection was performed, resulting in an "Operation timed out" in the Traceband sender, however, not every time the alert message is displayed on the receiver according to the codding done.

Could also be determined that there were not false positives during the experiments, as traffic above 60% is a value much higher than normal performance on a network, but at no point the estimator indicated a denial of service attack if it was not run by the Hping3 command.

Evaluation performed without traffic are the reference values of exclusive attacks and behavior, when not attacks running the value of available bandwidth is equal to the channel (100Mbps) and observations remain constant in the maximum number of states (10).

VII. DICUSSION

As outlined on the related work section, there are several alternatives for flooding DoS attacks detection using available bandwidth estimation, with this modified version of Traceband we present a novel technique taking advantage of the speed generated by the use of the Hidden Markov Model implemented on this tool. As result, this method is faster than others presented and shows no false positive alerts. However it has several limitations if the attack is able of disabling the network functions before the program algorithm is executed.

VIII. CONCLUSIONS

This work shows how it can be effectively detected the start of a denial of service SYN Flooding attack if it has real effects on the network, through the study of the behavior of the available bandwidth. From additional assessment it can be concluded that the detection of TCP IP Flooding attacks can be performed in the same manner, however if UDP packets are used the program is not as effective showing the alarm.

After the alert, a network administrator can know that the network is crossing abnormal traffic, but the real source of the attack is unknown especially since the attacks allows to hide IP source, making the traffic is redirected to a not existing IP or not responding to communication attempts of the victim.

Error in the generation of the alarm is due mainly to the initial force of the attacks that does not allow continuation of normal program execution, breaking down the necessary client communication for the analysis of the time difference between packets in a pair in the server.

ACKNOWLEGDEMENTS

This work was supported by the Gobernación del Departamento de Santander – Universidad Autónoma de Bucaramanga scholarship.

REFERENCE

[1] D. Kaur and M. Sachdeva, “Study of Recent DDoS Attacks and Defense Evaluation Approaches,” *ijetae.com*, vol. 3, no. 1, pp. 332–336, 2013.

[2] S. Ning and Q. Han, “Design and implementation of DDoS attack and defense testbed,” pp. 220–223, 2012.

[3] D. Kaur, M. Sachdeva, and K. Kumar, “Study of DDoS attacks using DETER Testbed,” *researchmanuscripts.com*, vol. 3, no. 2, p. 13, 2012.

[4] R. Chertov, S. Fahmy, and N. B. Shroff, “Emulation versus Simulation: A Case Study TCP-Targeted Denial of Service Attacks,” in *2nd International Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities, 2006. TRIDENTCOM 2006.*, pp. 316–325.

[5] C. D. Guerrero and M. A. Labrador, “On the applicability of available bandwidth estimation techniques and tools,” *Computer Communications*, vol. 33, no. 1, pp. 11–22, Jan. 2010.

[6] C. D. Guerrero and M. A. Labrador, “Traceband: A fast, low overhead and accurate tool for available bandwidth estimation and monitoring,” *Computer Networks*, vol. 54, no. 6, pp. 977–990, Apr. 2010.

[7] L. He, S. Yu, and M. Li, “Anomaly Detection Based on Available Bandwidth Estimation,” in *2008 IFIP International Conference on Network and Parallel Computing*, 2008, pp. 176–183.

[8] L. He, B. Tang, and S. Yu, “Available bandwidth estimation and its application in detection of DDoS attacks,” in *2008 11th IEEE Singapore International Conference on Communication Systems*, 2008, pp. 1187–1191.

[9] M. Alenezi and M. J. Reed, “Denial of service detection through TCP congestion window analysis,” in *World Congress on Internet Security (WorldCIS-2013)*, 2013, pp. 145–150.

[10] G. Carl, G. Kesidis, R. R. Brooks, and S. Rai, “Denial-of-service attack-detection techniques,” *IEEE Internet Computing*, vol. 10, no. 1, pp. 82–89, Jan. 2006.

[11] “Multi-Generator (MGEN) | Networks and Communication Systems Branch.” [Online]. Available: http://www.nrl.navy.mil/itd/ncs/products/mgen. [Accessed: 10-Jun-2014].

[12] A. Hussain, S. Schwab, R. Thomas, and S. Fahmy, “DDoS experiment methodology,” Proceedings of the DETER Community Workshop on Cyber Security Experimentation (Vol. 8), 2006.

Angélica María Niño Díaz, is a Systems Engineer from Universidad Industrial de Santander, with graduate studies on information security from Universidad Pontificia Bolivariana de Bucaramanga and a Master degree on Free Software from Univesidad Autónoma de Bucaramanga and Universitat Oberta de Calalunya. Her topics of interest include network and application security, management systems and financials markets.

Cesar D. Guerrero, is a professor at Universidad Autónoma de Bucaramanga (UNAB). He received his M.S. degree in Computer Science in 2002 and his M.S. degree in Computer Engineering in 2007. In 2009 he obtained his Ph.D. in Computer Science and Engineering from University of South Florida. Dr. Guerrero is Fulbright scholar and IANAS Fellow from the American National Academy of Sciences. He has been visiting professor at Stanford University (USA) in 2012 and Carlos III University (Spain) in 2013. His research interest includes Network Measurement and Education for Innovation. He has served as Technical Program Committee member of several journals including Computer Networks and Computer Communications, both Elsevier Science journals.