Assessing Denial of Service Vulnerabilities in DOCSIS

Scott Moser
Clemson University
100 McAdams Hall
Clemson, S.C. 29634-0974
smoser@cs.clemson.edu

Jim Martin
Clemson University
100 McAdams Hall
Clemson, S.C. 29634-0974
jim.martin@cs.clemson.edu

ABSTRACT

In previous work a DOCSIS model was added to '*ns*' to allow simulations to be run to analyze the performance of DOCSIS. These simulations showed that congestion caused by the asymmetric data paths and the MAC contention process caused several performance problems. It was shown that ACK compression could cause a drop in downstream throughput of TCP streams. A denial of service (DoS) threat was also identified, due to the DOCSIS contention process, allowing an attacker to overload the upstream channel, deteriorating the service quality perceived by all active subscribers.

Using an actual DOCSIS system, this study continues that simulation effort by running tests on a live system. The purpose is to show that the problems identified by simulation do exist in practice and to collect information from a live system that can be used to validate and improve the simulation model.

Categories and Subject Descriptors

C.2.5 [**Computer-Communication Networks**]: Local and Wide-Area Networks – *access schemes, high-speed, internet*.

General Terms

Measurement, performance

Keywords

DOCSIS, broadband access, Data over Cable, cable networks, TCP performance

1. INTRODUCTION

The use of broadband access via cable systems is now growing dramatically. The Data Over Cable Service Interface Specification (DOCSIS) [3] identifies the operation of the Media Access Control (MAC) layer for these systems. A DOCSIS system employs a hierarchical structure with the head-end connecting to a group of Cable Modem Termination System (CMTS) units each of which interfaces with many Cable Modem (CM) units. The current DOCSIS 1.1 system provides an asymmetric data path to the CM users with lower bandwidth in the upstream return path to the CMTS, while more of the cable bandwidth is allocated for downstream transmissions from the CMTS to the cable modems.

The CMTS controls the upstream flow of data between itself and the cable modems attached to it by sending MAP messages to the cable modems to indicate transmission timeslots for each CM waiting to send data.

The DOCSIS standard provides for Quality of Service (QoS) mechanisms so that different levels of service can be accommodated. The CMTS bases its timeslot allocation decisions on the QoS level of each data flow and real time requests from the cable modems for transmission bandwidth.

A contention scheme is used for the cable modems to request transmission bandwidth. The MAC protocol calls for each MAP allocation to contain timeslots for user data, maintenance data, and contention slots to be used for transmission requests [3][8]. A CM requests a transmission timeslot by using one of the available contention slots. Since the upstream and downstream channels are on different frequencies the CM can not determine contention as it transmits and must rely on notification from the CMTS during the next MAP time of the data slot allocations. The CM determines that contention has occurred if the next MAP message does not either assign a transmission slot, or acknowledge that the assignment is pending.

In prior work [9] it was shown that a DOCSIS network can become packet rate limited such that TCP acknowledgements are not transported upstream quickly enough causing poor downstream utilization. Since downstream TCP flows are impacted by the upstream transmission limits any intentional effort to cause additional upstream data traffic from a cable modem will increase congestion at the cable modem – CMTS interface. A denial of service (DoS) threat was identified, due to the upstream bottleneck, after an '*ns*'[12] model of DOCSIS was developed [10] and simulations were run. To accomplish the attack it is only necessary to know the IP addresses of a group of cable modems connected to a single CMTS. The attacker would then send pings, or TCP SYN packets, to that set of cable modems, causing increased congestion and a reduction in TCP throughput.

While the previous study identified a threat where multiple CMs under attack could impair performance, the research reported in this paper concentrates on the effects of an attack against a single CM. Simulations were first run using '*ns*' to provide a baseline of data. Equivalent tests were then run using both UDP and TCP data flows through an actual cable system connection. The results from the '*ns*' tests were used to determine initial parameters for

ACM SE'06, March, 10-12, 2006, Melbourne, Florida, USA

the live tests then the simulation test results were compared to the results of the tests on the physical cable system.

2. MOTIVATION

Most Internet traffic is TCP based and the primary concern is how the DOCSIS MAC protocol impacts these TCP flows. As described by Van Jacobson[6], data flow on a TCP connection follows a 'conservation of packets' principle where new packets are not placed into the network until a previous packet leaves the network. The acknowledgement packets returning to the sender are used to clock new packets into the network. This self-clocking feature provides for adjustment of the sending rate based on the round trip time through the network. When the system is in equilibrium the sender adapts its sending rate to the available bandwidth of the bottleneck link in the connection.

Under normal conditions there is a consistent, evenly spaced, flow of acknowledgement packets to the sender. If the acknowledgement packets become delayed this process is disrupted. In an asymmetric system, such as DOCSIS, with less upstream bandwidth, the consistent, evenly spaced, acknowledgement packet flow can be disrupted due to delays causing the acknowledgement packets to be bunched together. The smaller acknowledgement packets get queued behind larger data packets, causing them to bunch together and causing multiple acknowledgement packets to be transmitted from the queue at the same time. This is called ACK compression and is known to disrupt the self clocking feature of TCP [1,11,13].

DOCSIS implements a piggybacking scheme where further data requests can be piggybacked onto the current data transmission. This can improve the ACK compression problem somewhat by using the piggyback request to request bandwidth for the acknowledgement packets. The MAC header has the capability of defining an Extended Header field. These Extended Headers are used to request bandwidth for additional upstream transmissions during the current data transmission. With this approach, especially if the piggyback requests are only used for the acknowledgement packets, every request to send an acknowledgement packet does not have to go through contention. This has been shown to improve the effective throughput of TCP packets in an asymmetric network [4].

DOCSIS provides both Fragmentation MAC Headers, for splitting large packets into several smaller packets, and Concatenation MAC Headers, to allow multiple smaller packets to be combined and sent in a single MAC burst. Concatenation, like piggybacking, can also be used to reduce the occurrence of collisions by reducing the number of individual transmission opportunities needed. The negative impact is that ACK packets, when concatenated by the CM, become highly compressed, with multiple ACKs being delivered to the sender almost simultaneously. The concatenation MAC header precludes the use of the Extended Header field and therefore piggybacking of future requests can not be done in a concatenated frame.

As shown by simulation studies [9], a Denial of Service attack launched simply by sending a stream of pings, or a TCP SYN packet to a group of CMs can cause serious performance degradation. In [9], as the number of CMs under attack increased from 0 to 100 the collision rate increased from 48% to 68%, and the downstream utilization dropped from 45% to 10%. We extend this prior work by; further exploring the parameter space of such an attack, and testing the impact of a limited attack on a live network against a single CM.

3. RELATED WORK

While there has been work in the area of congestion-based denial of service most of it has been with 802.11 based wireless networks. In [2] the authors focus on the threats posed by DoS attacks against the 802.11 MAC protocol. They demonstrate MAC layer DoS vulnerabilities and analyze potential countermeasures. In [5] the authors show that weaknesses in the 802.11 MAC layer can be exploited to drop throughput and while adding fairness to the protocol will alleviate the problem somewhat it will not eliminate it. The authors in [16], after showing how an ad-hoc 802.11 network is vulnerable to malicious attacks and that intrusion prevention measures can reduce, but not eliminate, the problem, then describe methods of intrusion detection and appropriate response mechanisms. The authors in [7] evaluate the ability to detect misbehaving nodes and penalize them showing that handling misbehavior is important to the ability to guarantee service availability. In [14] the authors found that a malicious 802.11 node can easily prevent another station from sending and propose RTS validation as a defense against RTS/CTS induced congestion. Just as with 802.11 the denial of service vulnerability allows an attacker to exploit the inefficient upstream bandwidth request mechanism. However, malicious nodes are not our focus in this case. We focus here on the denial of service caused by unavailability of upstream bandwidth.

4. METHODOLOGY

4.1 Simulations

Prior to running tests on the physical system a simulation of the physical system was implemented using '*ns*'. Tests were run on the simulation to use for comparison with the actual physical system tests. Results of the physical system tests were then used as feedback to allow tuning of the model parameters. The simulation test model used is shown in Figure 1.

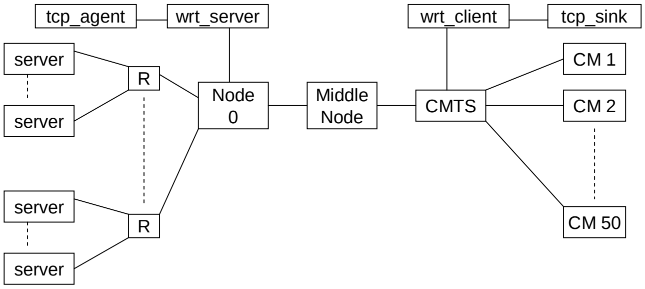

Figure 1. Simulation Model

Background traffic flows between the servers and the CMs in the system to simulate general traffic in the network. To simulate the DoS attack a ping was sent from a server to a CM.

4.2 Live System

The primary focus of this effort was to carry out tests on an actual DOCSIS network to show the effectiveness of a Denial of Service attack against a single CM and to collect information on the operational parameters of the live system to be used to improve the DOCSIS '*ns*' model.

The configuration involved is a single cable modem in a residential setting. Most operating parameters for the system were unknown, such as the number of CMs connected to the CMTS, the actual load at the time of the test, MAP times, piggybacking use and concatenation use, to name a few.

The configuration used for the tests was a PC, running Red Hat Linux 9, attached through a router to the CM. A machine on the Clemson University campus was used for the source end of the connection. A diagram of the system is shown in Figure 2.

For the first set of experiments the following was implemented:

- A continuous flow of UDP packets was sent from the campus machine to the home machine at various repetitive rates and packet sizes.

- The home machine echoed the packets back to the sender.

- The sender calculated the round trip time and loss rates of the packet flow.

During the experiments *tcpdump*[15] captured packets on the machines at both ends of the connection as shown in Figure 2. This data was then analyzed and plotted for evaluation.

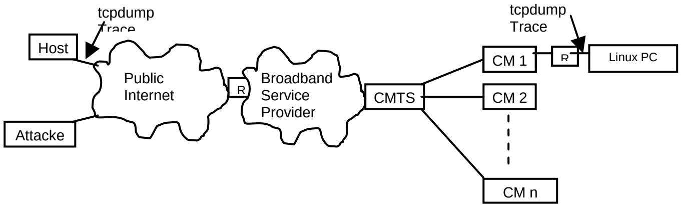

Figure 2. Live test network

A second set of experiments was then undertaken, identical to the first, with the exception that another campus machine was continuously sending repetitive TCP flows to the home machine to simulate a typical web traffic flow.

5. RESULTS

5.1 Simulation Results

An initial simulation was run to show the collision rate as the number of CMs under attack is increased. Figure 3 shows the results of the test run with 100 CMs as the number under attack increases from 0 to 100.

As the number of CMs under attack increases from 0 to 100 the collision rate goes from less than 10 to more than 65. As more CMs are stimulated by the attacker the rate of collisions gets very high. Additional analysis shows that the impact of this DoS is affected by system parameters including CMTS and CM buffer sizes and concatenation behaviors.

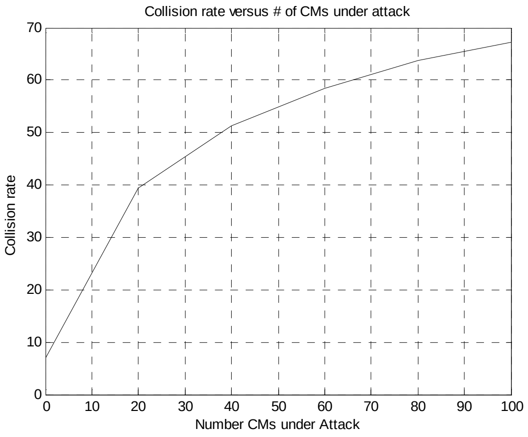

Figure 3. Collision Rate vs. Number of CMs under attack

We created a second simulation experiment, one that models the live system. The simulation uses a varying number of CMs (0-100), but only one CM under attack. Attack packets were sent every .5 seconds. Figure 4 shows the collision rate for these experiments. Figure 5 shows the downstream utilization.

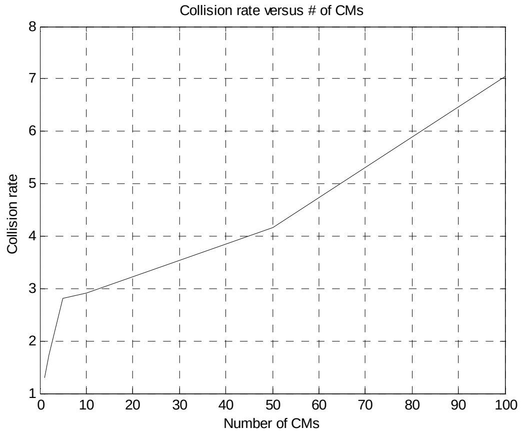

Figure 4. Collision Rate vs. Number of CMs

As can be seen the collision rate still increases, but far less dramatically. With only one CM under attack, the collision rate goes from just over 1 to about 7 as the number of CMs increased. The downstream utilization increases as the number of CMs is increased due to the fact that each CM consumes an amount of downstream bandwidth that is typical for a web browsing session. One CM under attack has little impact on overall system performance, although the performance perceived at that victim CM is greatly impacted.

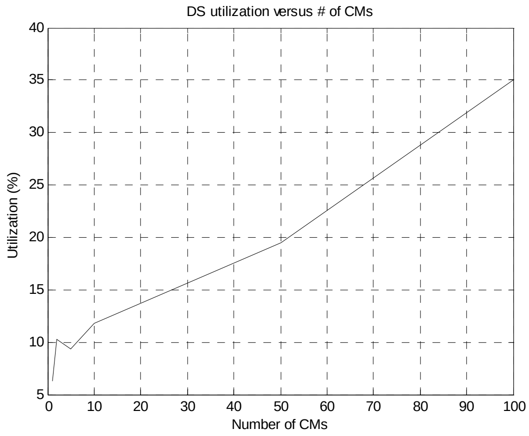

Figure 5. Downstream Utilization vs. Number of CMs

Following this a system was setup with 50 CMs and again only 1 CM under attack. These tests were run with varying attack rates from 1 ms. to 30 seconds. Then tests were run with packet size varied from 64 bytes to 3000 bytes. These tests all confirm the impact of a single CM under attack to the overall system performance was minimal. This seemingly obvious result is actually significant as the upstream and downstream service rates were set to the channel capacities.

5.2 Live System Results

Attack rates of .5 ms. to 50 ms. were chosen for the live system experiments. In addition, packet sizes of 12, 64, 256 and 512 bytes were used. Figure 6 shows the packet loss rates for these runs. It can be seen that as the attack rate frequency increases the packet loss rate increases. With the larger packet sizes the increase begins with slightly less frequent attack rates.

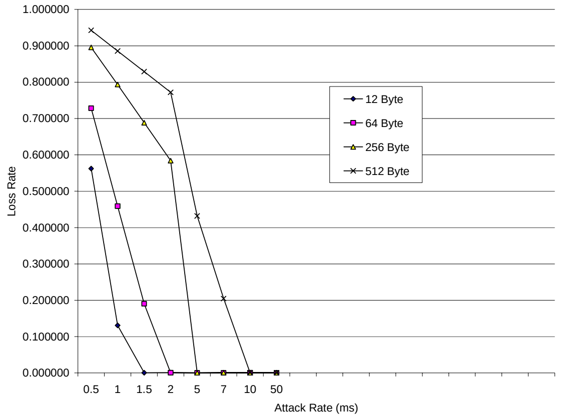

Figure 6. Loss Rate vs. Attack Arrival Rate

Evaluation of the numbers indicates that the drop-off point actually begins to occur when the combination of the packet size and the attack rate begin to reach the point where the available bandwidth is being consumed. For example, the 512 byte packets, plus 28 bytes for IP and UDP headers, give us a total packet size of 540 bytes. Sending one packet every 7 ms would give us (540*8)/.007 = 617143 bps. Sending one packet every 10 ms would require 432000 bps. Repeat tests suggest that the available downstream bandwidth was 5 Mbps and that the available upstream bandwidth was 512 Kbps. In the 7 ms case we see a 20% packet loss rate, while the loss rate for the 10ms rate was near zero since the 512Kbps was not exceeded. The DoS, in this case, was caused by flooding.

The second set of experiments was run with the repetitive TCP flows sent during the time of the attack. Figure 7 shows the downstream throughput of those TCP flows. It can be seen that the throughput drops as the UDP data flows consume more of the bandwidth. The larger the packet size, the sooner that occurs.

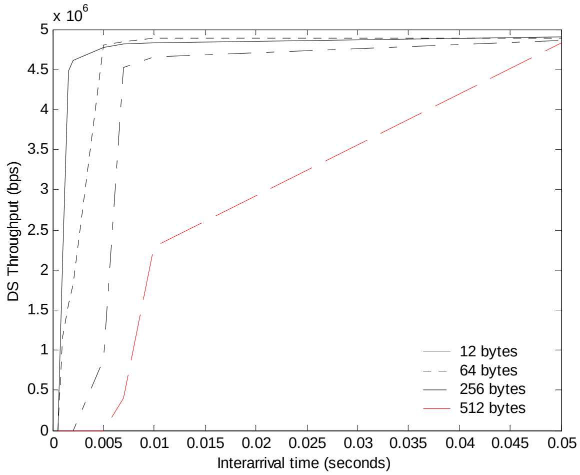

Figure 7. Downstream TCP Throughput vs. Attack Rate

The *tcpdump* files, collected at both the sending and receiving sides of the connection, were analyzed to produce probability distribution functions for the inter-arrival times of the packets. Figure 8 shows the distribution for the packets leaving the sender for a 5 ms attack rate.

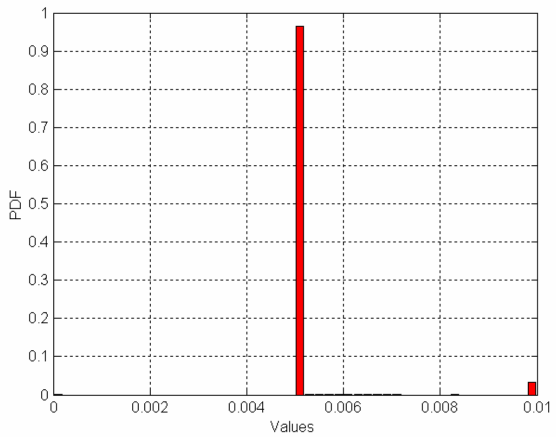

Figure 8. PDF of packets from sender (5 ms)

Figure 9 shows the arrival distribution of the packets arriving at the target CM for the same run. This plot confirms the variability in the transmission channel from sender to receiver.

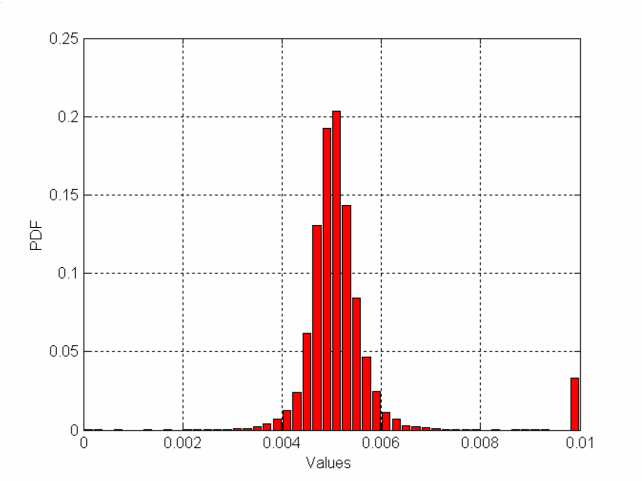

Figure 9. PDF of arrival packets at victim (5 ms)

Figure 10 shows the distribution for the packets arriving back at the sender after being returned from the victim. While the packets arrived every 5 ms. the return rate is determined by the map time of the CMTS. We see a major mode at 4 ms. and a minor mode at 6 ms. Since the CM, once a packet is ready to send, must request during the first map time, and then receive a mini-slot during the second, we can conclude that the map time of this CMTS is 2 ms. The mode at 6 ms. indicates the packets either arriving just after the request, or being delayed by the contention process.

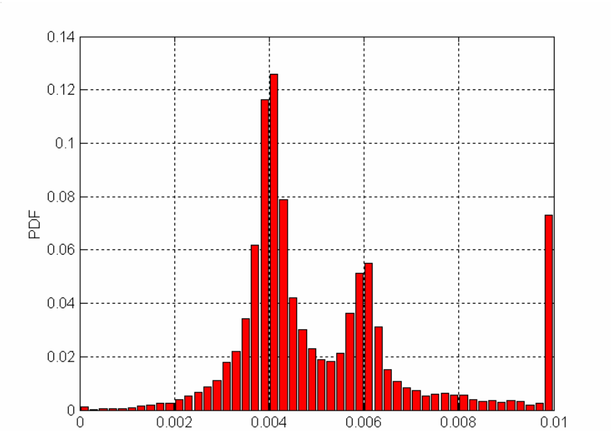

Figure 10. PDF of packets returned from victim (5 ms)

As a comparison, Figure 11 is a plot of a 5 ms. run by the simulator. Comparing Figure 11 with Figure 10 indicates that the model is providing an accurate simulation of the system.

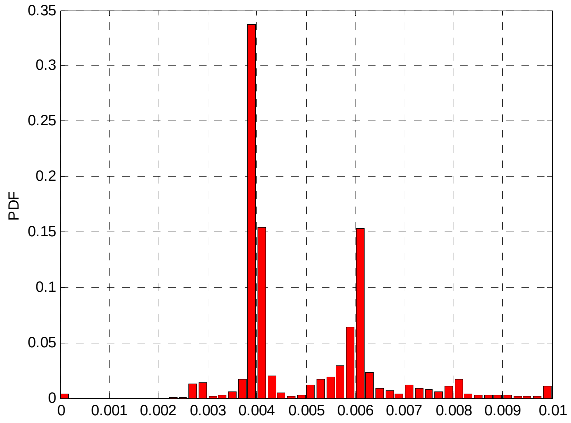

Figure 11. PDF from simulation (5 ms)

A distribution function for a run with a 2 ms. attack rate is shown in Figure 12. The appearance of a nearly 45% mode at zero indicates that those packets are being delivered consecutively with no delay between them. Figure 13 shows the distribution function for a run with a 1 ms. attack rate. In this case almost 80% of the packets are concatenated. The system is obviously concatenating packets for delivery from the CM to the CMTS. The faster the packets arrive, the more of them are concatenated. With a 1 ms. attack rate only a small number of packets are going through the contention process. In Figure 10 with a 5 ms. attack rate this does not occur because the packets arrive at a rate slower than the map time and can not be concatenated for transmission during a single map time.

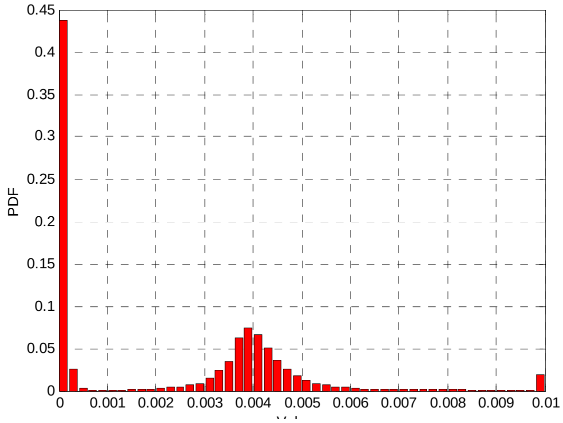

Figure 12. PDF with 2 ms. attack rate

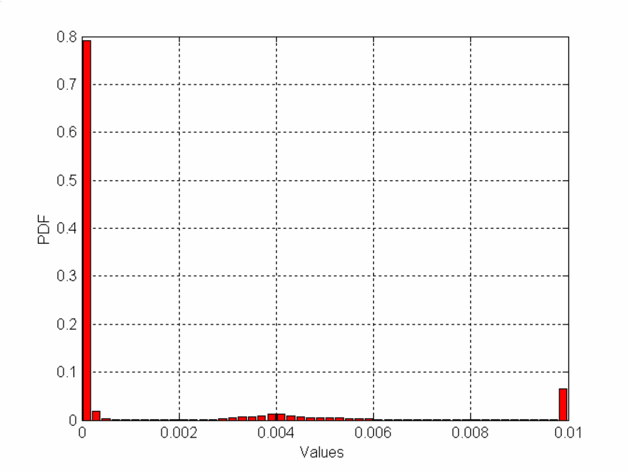

Figure 13. PDF with 1 ms. attack rate

6. CONCLUSIONS AND FUTURE WORK

We summarize our results as follows:

- The impact of a DoS that includes multiple CMs is highly dependent on system configuration, in particular on the maximum number of IP packets that are allowed in a concatenated frame.
- A DoS involving a single CM victim will not significantly impact other CMs even when service rates are very high.
- Due to the limited upstream bandwidth it is easy to accomplish a DoS attack by flooding alone.
- It is possible to identify DOCSIS MAC configuration items by observing performance.

The data collected from the live system has validated the model developed for '*ns*' and provided some additional information by verifying the 2 ms. map time and that the system does use concatenation when possible. This additional configuration information can be used in the future to improve the simulation.

The ability to launch a DoS attack against a DOCSIS customer was demonstrated. However, when attacking a single CM the DoS is due largely to a flooding attack. A true congestion based attack would require an attack against multiple CMs communicating to the same CMTS.

We are developing a DOCSIS test-bed which will allow us to test a true congestion based attack against a group of CM units simultaneously.

7. REFERENCES

[1] Balakrishnan, H., et. al., *TCP Behavior of a Busy Internet Server: Analysis and Improvements*, IEEE Infocomm98, 1998

[2] Bellardo J. and Savage S. *802.11 Denial-of-Service Attacks: Real Vulnerabilities and Practical Solutions*, Proceedings of the USENIX Security Symposium, August 2003

[3] Cable Television Labs Inc., CableLabs, *Data Over Cable Service Interface Specification – Radio Frequency Interface Specification*, SP-RFIv2.0, available at http://www.cablemodem.com/specifications/specifications20.html

[4] Elloumi O., et. Al., *A Simulation-based Study of TCP Dynamics over HFC Networks*, Computer Networks, Vol. 32, No. 3, pp301-317, 2000

[5] Gupta V., Krishnamurthy S. and Faloutsos M. *Denial of Service Attacks at the MAC Layer in Wireless Ad Hoc Networks*, Proceedings of 2002 MILCOM Conference, Anaheim, CA, October 2002

[6] Jacobson, V. *Congestion Avoidance and Control*, Proceedings of SIGCOMM '88 (Stanford, CA, August 1988), ACM

[7] Kyasanur P. and Vaidya N. *Detection and Handling of MAC Layer Misbehavior in Wireless Networks*, Proceedings of the International Conference on Dependable Systems and Networks, San Francisco, CA, June 2003

[8] Lin, Y., Yin, W., and Huang, C., *An Investigation into HFC MAC Protocols: Mechanisms, Implementation, and Research Issues,* IEEE Communications Surveys and Tutorials, available at http://www.comsoc.org/livepubs/survey/public/3q00issue/yin.html

[9] Martin J. *The Interaction Between the DOCSIS 1.1/2.0 MAC Protocol and TCP Application Performance,* HetNets2004

[10] Martin, J. and Shrivastav N. *Modeling the DOCSIS 1.1/2.0 MAC Protocol,* Proceedings of the 2003 International Conference on Computer Communications and Networks, Dallas TX, October 2003.

[11] Mogul, J., *Observing TCP Dynamics in Real Networks*, Technical Report, Digital Western Lab, April 1992

[12] *ns 2,* available at: www.isi.edu/nsnam/ns

[13] Paxson, V., *Measurements and Analysis of end-to-end Internet Dynamics*, Ph.D. dissertation, University of California, Berkeley, CA, 1997

[14] Ray S., Carruthers B. and Starobinski D. *RTS/CTS-Induced Congestion in Ad Hoc Wireless LANs,* Proceedings of IEEE WCNC, March 2003

[15] *tcpdump*, available at: www.tcpdump.org

[16] Zhang Y. and Lee W. *Intrusion Detection in Wireless Ad-Hoc Networks*, Proceedings of the Sixth Annual International Conference on Mobile Computing and Networking, Boston, MA, August 2000