Modeling and Simulation of Game Theory-based Defense Mechanisms against DoS and DDoS Attacks

V.Datla and C.Ellis
Department of Computer Science
University of Memphis
(vvdatla, ceellis@memphis.edu)
Advisor: Dr. Q. Wu, Dr. S. Shiva

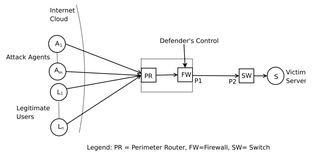
Introduction

The weakness of traditional network security solutions is that they lack a quantitative decision framework. As an example, a network administrator and an attacker can be viewed as two competing players participating in a game. For each of DoS (Denial of Service)and DDoS (Distributed Denial of Service) cases, we design a static game.

We propose game models to capture the interaction between the DoS/DDoS attacks and the potential mitigation techniques. We also present the theoretical analysis for the attacker's and defender's strategy which can lead to the Nash equilibrium. We implement a new module in NS-3, Nethook, that provides packet inspection capabilities similar to that of Linux NetFilter. This module can be reused in any future experiment that requires packet inspection.

We validate our analytical results via extensive simulation in ns-3.

Model



Symbol	Meaning
S	the victim server
PR	the perimeter router
FW	the firewall
SW	the switch
P_1	the outgoing point from FW
P_2	the incoming point to SW
B	the bandwidth of the pipe (P_1P_2) between the firewall FW and the switch SW
n	the number of legitimate users.
m	the number of attacking nodes
r_l	the expected bit rate of a legitimate flow
σ_l	the standard deviation of a legitimate flow rate
r_A	the bit rate of an attack flow.
γ	the minimum bit rate for a flow to be considered alive

Figure1. Network Topology- *n* legitimate users want to access the server *S*. The bandwidth of the pipe (P1, P2) between FW and SW is limited and is subject to a DoS/DDoS attack. The defender's control is present at the firewall. The attacker *A* has *m* agents.

Assumptions:

- We assume that one single attacker controls all of the attacking nodes.
- There is infinite bandwidth available on the channel between the PR and the FW.
- The FW's belief on the legitimacy of the flow decreases with the increase of the flow rate.
- A flow f is dropped in either of the two situations: (i) f does not pass the firewall rule. (ii) f is dropped with a probability at point P1 when total incoming flow rate T is more than the available bandwidth B.

We consider the interaction between the attacker and the network administrator (defender) as a two player static game, and study the existence of an equilibrium in these games. We also show the benefit of using the game-theoretic defense mechanism. against DoS and DDoS attacks

Analysis: Impact of Attack with No Defense

- Once a player decides his strategy he does not have a second chance to change it. We consider the attacker's reward is not necessarily the defender's cost, i.e. it could be a zero-sum or non-zero sum game.
- The only actions available to the attacker are to set the sending bit rate and to choose the number of attacking nodes, m. It is assumed that the bit rate is same for all of the attack flows, which is represented by r_A.
- In an attack situation the total flow rate is:

$$T = (X_1 + X_2 + \ldots + X_n) + m \cdot r_A.$$

- If T > B, then the denial of service occurs due to congestion in pipe (P1P2).
- Average bandwidth consumption (by the attacker) ratio is:

$$v_b^{nd} = \frac{m \cdot \alpha \cdot r_a}{B} = \frac{m \cdot r_A}{n \cdot r_l + m \cdot r_A}$$

- Ratio of lost users to the total number of users on average is :

$$v_n^{nd} = \frac{n - n_g}{n} = P[X_i < \frac{\gamma}{\alpha}] = P[X_i < \frac{\gamma(n \cdot r_l + m \cdot r_A)}{B}]$$

We model the attacker's net payoff as $V^a = w_b^a \cdot v_b^{nd} + w_n^a \cdot v_n^{nd} - w_c^a \cdot v_c$ where w_b^a, w_n^a, and w_c^a are the attacker's corresponding weight parameters.

We model the attacker's net payoff as $V^d = -w_b^d \cdot v_b^{nd} - w_n^d \cdot v_n^{nd} + w_c^d \cdot v_c$ where w_b^d, w_n^d and w_c^d are the defender's corresponding weight parameters.

Analysis: Impact of Attack in presence of Firewall

- Firewall is the defense agent of the network administrator. The firewall drops an incoming flow with a probability depending on the flow rate.
- The dropping rate is modeled by a sigmoid function as follows:

$$F(x) = \frac{1}{(1 + e^{-\beta\frac{(x-M)}{B}})}$$

where M represents the flow rate for which drop rate is 0.5 and β s a scaling parameter.

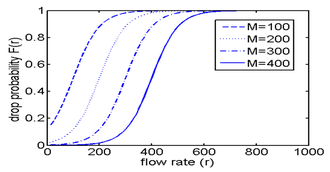


Figure 2. A few sample S curves. Dropping rate of a flow at the firewall is modeled by a S curve. The X axis is the flow rate and Y axis is the drop probability. m represents the flow rate for which drop rate is 0.5.

- FW drops a flow of rate r with a probability F(r). It is worth to note that some of the legitimate flows might get dropped at FW.
- We consider that the defender controls the value of M, which is the only defense action.
- Let the average number of legitimate flows passing through the firewall be n'. So, $n' = n.(1 - F(r_l))$ where $r_l$represents the expected rate of one legitimate flow.
- The average number of attack flows passing through the firewall is: $m' = m \cdot (1 - F(r_A))$.

Ratio of average bandwidth consumption by the attacker is:

$$v_b = \frac{m' \cdot r_A}{n' \cdot r_l + m' \cdot r_A}.$$

Ratio of lost users to the total number of users on average is:

$$v_n = \frac{n - n'}{n} + \frac{n' \cdot P[X_i < \frac{\gamma(n' \cdot r_l + m' \cdot r_A)}{B}]}{n}.$$

- We use the notion of Nash equilibrium to determine the best strategy profile of these two players. Each player has the goal to maximize his payoff.
- The attacker needs to choose an optimum m and r_A.
- The defender needs to choose an optimum M for the sigmoid function to be used by the firewall.
- Recall that M represents the flow rate for which the drop probability at FW is 0.5.

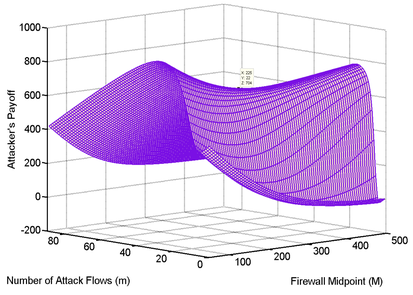


Figure 3. The attacker's payoff V_a for different number of attack flows, m and different values of the firewall midpoint, M. We observe a saddle point at m = 22; M = 225, which represents the Nash equilibrium.

Simulations

- NS-3 is an advanced network simulator tool written in C++.
- Flow Monitor was not applicable in this situation as it depends entirely upon the traced output of packet data, rather than inspecting these packets as they traverse NS-3's protocol stack.
- In our module we had the necessity of developing a packet filtering module based on the game theory model and collecting statistics on the game agent was an interest for us.

For the implementation of the packet filtering module we had to implement a unique network hook, which we have written to observe packet flow information as they actually pass through the stack rather than at the end of the simulation.

Contribution to NS3

- Our NetHookFilter module provides a means to manipulate the standard packet handling routines in NS-3.
- NetFilter is an useful component of modern networked systems useful for addressing various issues regarding packet inspection and manipulation.
- Traditionally NetFilter implements hooks during a packet's traversal through the protocol stack at the following locations: pre-routing, local deliver, forward, and post-routing.
- Each hook corresponds to locations in which one might be interested in viewing/manipulating a packet as it traverses the stack.

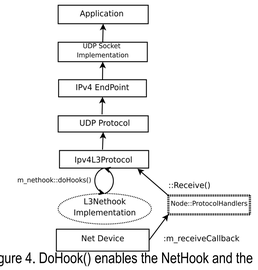


Figure 4. DoHook() enables the NetHook and the NetHook returns a boolean value that determines whether the packet needs to be dropped or not.

Experiment Setup

- We use the traditional dumbbell topology for this experiment.
- The experiment is run in 10 cycles, where there are 50 legitimate nodes whose packet size is 512 bytes and sends at a rate of 15Kbps.
- The first cycle has 5 attack nodes that send at a total of 5Mbps that is divided evenly between each attack node, and the number of attack nodes increases by 5 for each cycle.
- Within each cycle we change the filter midpoint setting three times at 250Kbps, 500Kbps, and 700Kbps respectively.

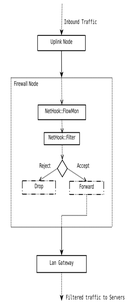


Figure 5. NetHook::Filter Integration into Experiment Network Topology

Results

Our simulation focuses only on the percentage of bandwidth consumed by the legitimate and attacking nodes.

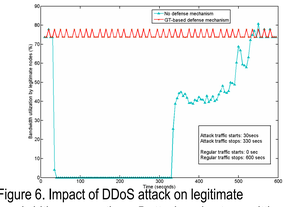


Figure 6. Impact of DDoS attack on legitimate bandwidth consumption—5 attack nodes transmitting at 1Mbps each (5Mbps total), 50 legitimate nodes transmitting at 15Kbps each (750Kbps total), and the S-curve midpoint is set at 500Kbps

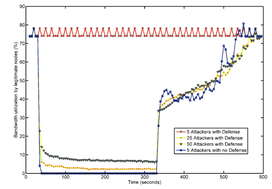


Figure 7. Bandwidth consumed by legitimate nodes when varying the number of attack nodes—The total attack bitrate remains at 5Mbps.

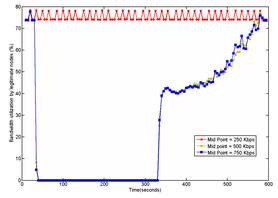


Figure 8. Bandwidth consumed by legitimate nodes when varying the S-curve midpoint—There are 15 attack nodes whose rate equals 5Mbps in total.

Conclusion and Futurework

- We presented a game theoretic model as a defense mechanism against a classic bandwidth consuming DoS/DDoS attack.
- Validation of our analytical results was performed utilizing the NS-3 network simulation tool.
- We plan on extending our simulation in the future to incorporate the a normal distribution to select the bitrate for legitimate flows.
- In addition, we plan on studying the applicability of our game theoretic based defense mechanism for scenarios where the attacker is interested in exploiting specific protocol mechanisms in order to create the condition.
- The TCP congestion window is one example of such possibilities. Further, we plan on simulating a dynamic game where both the attacker and defender can alter their strategy during the attack event.
- Furthermore, we plan on contributing our NetHook module to the NS-3 codebase in order to make it available to other researchers interested in packet manipulation within the simulator.