A Partial Deployment Formulation for a Distributed Denial of Service Defense scheme

EL736 Final Project
Ho-Yu Lam
hlam01@utopia.poly.edu

I. Introduction

Distributed denial-of-service *(DDoS)* attacks are serious threats to servers in the Internet. In such an attack, large volume of packets are sent to exhaust critical resources of a victim host with the help of a large number of compromised Internet hosts *(zombies)*. Without the needed resources, services to legitimate users are denied.

Lam et. al. [1] proposed a Coordinated Detection and Response *(CDR)* scheme that defends against DDoS attacks. The CDR scheme consists of two types of detection and response agents, namely the Transit Agent *(TA)* and Stub Agent *(SA)*. The TAs and SAs are deployed in the Internet transit network and stub network respectively. As with all other distributed defense schemes, the effectiveness under partial deployment needs to be addressed. While almost all partial deployment scenarios effectively suppressed attack traffic, some configurations are better than others in protecting legitimate traffic from collateral damage.

This project aims to model the partial deployment problem using the link-path formulation. The main difficulty of this formulation is the different effects that the two agents can have on the two type of attack and legitimate traffic. The properties of the two agents, which shall be discussed in section II-A, makes existing placement schemes [2], [3] that mainly focus on homogeneous agent placement not applicable.

Section II provides a brief overview on the CDR scheme. Details of the formulation is provided in section III.

II. Coordinated Detection and Response Scheme

The Coordinated Detection and Response *(CDR)* scheme proposed by Lam et. al. [1] is a distributed defense scheme against DDoS attacks. They use a Transit-Stub network model [4] and is shown in figure 1.

In such a model, every domain can be classified as either a stub network or a transit network. A stub network connects end hosts to the Internet and is usually operated by a local ISP. A transit network interconnects stub networks. Backbone ISPs and regional ISPs are examples of transit networks. The Transit-Stub network model assumes that transit networks do not carry traffic for end users directly. Thus, traffic generating nodes (end hosts) are only connected to Stub networks.

As shown in the figure, very often the packets have to travel through several networks before getting to the destination. As for the scenario of a DDoS attack, each of the attackers, legitimate users and the victim server are connected to a stub

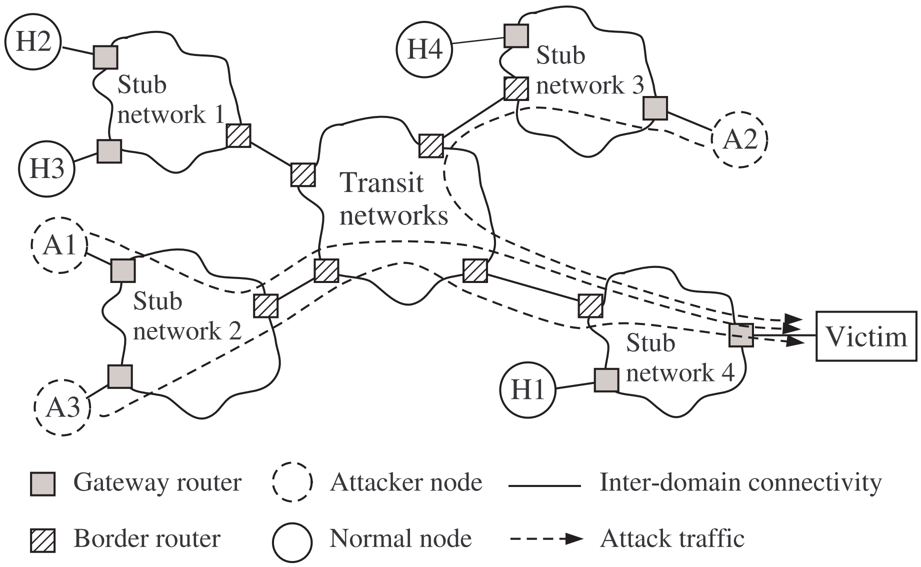

Fig. 1. Network model on which our scheme is based.

network. The traffic usually passes through two stub networks, one on the sender side and the other on the victim side, and one or more transit networks.

The CDR scheme proposes to deploy Stub agents in the stub network for both detection and filtering while Transit agents in the transit network for filtering only. It is obvious that traffic can be throttled only at networks with either SA or TA deployed. A common challenge for any scheme that required distributed deployment of agents is how to achieve a reasonable effectively without deploying agents in every possible deployment locations (full deployment). As mentioned by Lam et. al., the partial deployment of the CDR scheme can still be very effective in suppressing attack traffic. However, protection against collateral damage of legitimate traffic is dependent on deployment configurations. Some configurations are found to be more effective than others.

Since this project focuses only on the partial deployment problem, details of the detection and filtering schemes are not covered. Interested readers may refer to [1]. Properties of TAs and SAs that are relevant to our project is discussed in the next section.

A. Properties of Transit agent and Stub agent

An SA deployed in the stub network analyze all traffic flowing through the network and differentiate legitimate flows from attack flows. Since our focus is on the partial deployment problem, we assume that the SA can always perfectly identify legitimate flows from attack flows. Once identified, attack traffic would be rated limited with a gradually increasing rate until all attack traffic is blocked. There are some rate-limit adjustment dynamics, which happens during a transient period of minutes. Given that a typical DDoS attack lasts for many hours if not days and weeks, the dynamics during the transient period of minutes are unimportant as far as the partial deployment problem is concerned. Thus, we assume that the SA completely blocks attack traffics once deployed.

It is noteworthy that there exists too many stub networks in the Internet for a full deployment to be feasible. Therefore, the CDR scheme includes a second level of defense– the Transit agent. The TA is responsible for rate-limiting attack packets passing through the transit network as instructed by the victim stub agent. Also, a TA will mark packets passing through it in such a way that the victim server is able to identify which interface of which TA that a certain packet has passed through. Without going into details of how this feedback-control system works, this project assumes that the victim will always instruct a TA to completely block traffic from an interface of which non-zero attack traffic passed through.

To avoid TA being overloaded by the large volume of traffic flowing through the transit network, TA are not differentiating between attack traffic and legitimate traffic but doing computationally light weight unconditioned rate-limit task on each ingress interface. This action, however, will also affect legitimate traffic. To avoid collateral damage as much as possible, SA shall mark legitimate packets such that those packets are not rate-limited by TAs.

III. Formulation

indexes:

$g = 1, 2, ..., G$	legitimate demands
$a = G+1, G+2, ..., G+A$	attack traffic (demands)
$d = \{a, g\}$	demands
$p = 1, 2, ..., P_d$	candidate paths for flows realizing demand d
$t = 1, 2, ..., T$	candidate links for transit agents
$s = T+1, T+2, ..., T+S$	candidate links for stub agents
$e = \{t, s\}$	links

constants:

δ_{edp}	$= 1$ if link e belongs to path p realizing demand d; 0 otherwise
c_t, c_s	link capacity of Transit agent t and Stub agent s respectively
h_g, h_a	volume of legitimate demand g and attack traffic a respectively
ξ_t, ξ_s	unit cost of a transit agent t and a stub agent s respectively
B	Budget for agent deployment
$f(g,p)$	$= \begin{cases} 0 & \text{if } \sum_t \delta_{tgp} \sum_a \sum_p \delta_{tap} \gamma_t \hat{x}_{ap} > 0 \\ 1 & \text{otherwise} \end{cases}$
$b(a,p)$	$= \begin{cases} 0 & \text{if } \sum_t \delta_{tap} \gamma_t > 0 \\ 1 & \text{otherwise} \end{cases}$

variables:

x_{pd}	flow allocated to path p of demand d
$\hat{x}_{pa}$	offered attack flow a allocated to path p after SA filtering, if any
γ_t	$= 1$ if TA deployed on link t; 0 otherwise
γ_s	$= 1$ if SA deployed on link s; 0 otherwise

objective:

$$\text{maximize} \qquad \frac{\sum_p \sum_g x_{gp}}{\sum_g h_g} \tag{1}$$

constraints:

$$\sum_p x_{gp} \le h_g, \ \forall g \tag{2}$$

$$\sum_g \sum_p \delta_{egp} x_{gp} \le c_e - \sum_a \sum_p \delta_{eap} x_{ap}, \ \forall e \tag{3}$$

$$\sum_p \hat{x}_{ap} = h_a - h_a \sum_s \gamma_s \delta_{sa1}, \ \forall a \tag{4}$$

$$x_{gp} \le h_g f(g,p) + h_g \sum_s \gamma_s \delta_{sg1}, \ \forall g \tag{5}$$

$$x_{ap} = \hat{x}_{ap} b(a,p) \ \forall a \tag{6}$$

$$\sum_t \xi_t \gamma_t + \sum_s \xi_s \gamma_s < B \tag{7}$$

This project assumes an AS level topology with transit-stub network model. Each node represents an autonomous system of either a transit network or a stub network. We further assume that no demand originates from transit nodes.

Demands from G legitimate clients are represented as legitimate demands g in the network. To represent the attack traffic from A distributed zombies, demand with indexes $a = G+1, G+2, ..., G+A$ are used. Although both transit agents and stub agents are deployed in Internet routing nodes, the flow filtering activities are carried out on individual links. Therefore, one may view the agents as if they were deployed in individual candidate ingress links as far as the effect on flow allocation is concerned.

As discussed in section II-A, a single stub agent deployment should affect all demands going through that stub network. A virtual link and virtual rode is appended to each stub network node in the topology, where demands are generated. Demands are now originating from virtual nodes instead such that each virtual link is now a single SA candidate link that affects all demands from the corresponding Stab network. The candidate links are simply those that connects a stab node to a transit node or one between transit nodes. A tree view of the mentioned topology model is shown in figure 2. Indexes for candidate links of TAs and SAs are denoted as t and s.

The partial deployment problem can be viewed as a deployment with budget constraints. ξ_t and ξ_s are used to specify the unit cost of deploying an agent to each of the candidate links t and s respectively. The budget for deployment is given by B. Equation 7 specifies the budget constraint.

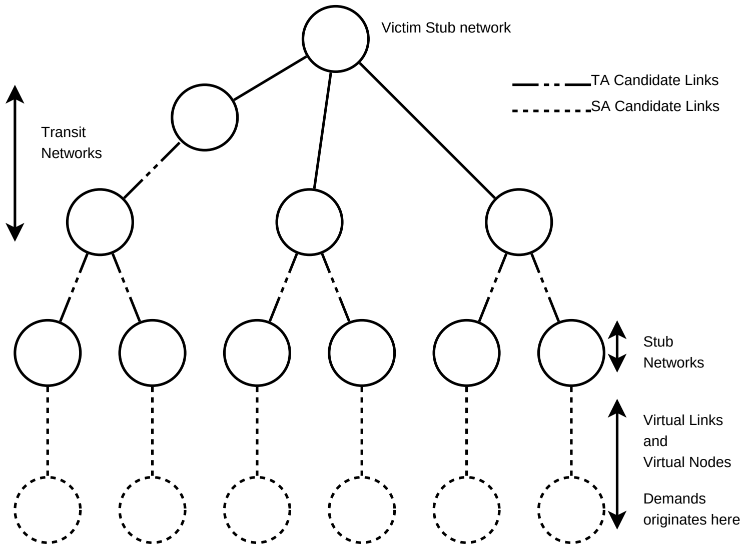

Fig. 2. A tree view of the topology model.

As with standard link-path formulations, we define the constants as shown above. δ_{edp} defines the link-path incidences. Capacities of individual candidate links for TAs and SAs are given by c_t and c_s respectively. Demands of legitimate clients and attack traffic are represented by h_g and h_a respectively.

SAs, where deployed, would identify and filter attack traffic. Thus, after passing through a candidate SA deployment link, the effective demand h_a of an attack traffic that concerns TAs can be zero instead of h_a. The variable $\hat{x}_{pa}$ together with equation 4 is used to reflect this property. The $\sum_s \gamma_s \delta_{sa1}$ in equation 4 gives one when a SA is deployed on link s that attack demand a passes through. Only the first path, as specified as δ_{sa1}, is considered in the sum because there is exactly one SA candidate link in which all paths of a demand must pass through. As a result, the offered attack volume $\hat{x}_{ap}$ is forced to zero if its corresponding SA is deployed or remains h_a otherwise.

After calculating $\hat{x}_{ap}$, the offered attack volume after SA filtering, the attack traffic may also be filtered by TA. Equation 6 determines if a TA is in place to filter the attack traffic. Function $b(a,p)$ returns zero if at least one of the TA candidate links along path p of attack flow a has TA deployed. That forces the attack flow to be $\hat{x}_{ap}$ when no TA is deployed on the path or zero otherwise.

Since attack traffic is unresponsive to packet drops, the effective bandwidth available to legitimate demands on each link can be approximated by subtracting the total offered attack demand going through the link from the link capacity, as shown on the right part of equation 3. The constraint in equation 3 effectively limits the legitimate flows to the remaining capacity, thus simulating the effect on legitimate flows sharing the same link with attack flows.

For simplicity, we view the filtering action of TAs as a total blockage of traffic if there is any non-zero attack demands on the link. The function $f(g,p)$ is defined to reflect this assumed filtering criteria, for the sake of readability in the constraints section. The input parameters are the legitimate flow index g and the candidate path p concerned. $f(g,p)$ gives zero if at least one of the candidate TA links along the path p that the legitimate demand g goes through satisfies both: (1) has TA deployed; and (2) has non-zero offered attack traffic passing through it. This function $f(g,p)$ is used in the first term on the right of equation 5 to reflect the collateral damage by TA.

The second term of equation 5 reflects the nice effect of SA which avoids collateral damage. As mentioned in section II-A, SA would mark legitimate packets such that the packets can pass through TA without being filtered. The summation $\sum_s \gamma_s \delta_{sg1}$ returns one if the SA link serving the legitimate demand concerned is deployed, thus adding h_g to the upper bound of x_{gp}. Although the resulting upper bound may be as high as $2h_g$, equation 2 prevent the sum of x_{gp} from exceeding the actual demand.

With all these constraints, the objective is to find the partial deployment configurations γ_t and γ_s such that the fraction of legitimate demands that are satisfied is maximized.

References

[1] H. Lam, C. Li, S. Chanson, and D. Yeung, “A coordinated detection and response scheme for distributed denial-of-service attacks,” in *Proceedings of the IEEE International Conference on Communications (ICC)*, Jun. 2006.

[2] B. Li, M. Golin, G. Italiano, X. Deng, and K. Sohraby, “On the optimal placement of web proxies in the Internet,” *INFOCOM’99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 3, 1999.

[3] L. Qiu, V. Padmanabhan, and G. Voelker, “On the placement of Web server replicas,” *INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 3, 2001.

[4] K. Calvert, M. Doar, and E. W. Zegura, “Modeling internet topology,” *IEEE Communications Magazine*, vol. 35, pp. 160–163, June 1997.