NGHIÊN CỨU KỸ THUẬT NHẰM HẠN CHẾ SỰ TIÊU TỐN BĂNG THÔNG MẠNG DO BỊ TẤN CÔNG DDOS

Trần Thị Yến[1], Lê Hoàng Hiệp[1],
Nguyễn Văn Trung[1], Nguyễn Văn Vũ[1],
Vũ Thị Nguyệt[1], Đinh Khánh Linh[1]

Title: *Study technique to limit bandwidth spending from DDoS attacks*

Từ khóa: *Tấn công băng thông, Giảm tải băng thông, Tấn công từ chối dịch vụ, DDos và băng thông, Từ chối dịch vụ phân tán*

Keywords: *Bandwidth Attack, Reducing bandwidth, Denial of service attack, DDos and bandwidth , Flood the Bandwidth, Distriuted Denial of Service.*

Lịch sử bài báo:
Ngày nhận bài: 12/6/2019;
Ngày nhận kết quả bình duyệt: 29/7/2019;
Ngày chấp nhận đăng bài: 12/8/2019.

Tác giả:
[1]Trường ĐH CNTT&TT Thái Nguyên

Email: *lhhiep@ictu.edu.vn*

TÓM TẮT

Với đặc điểm của kiểu tấn công DDos là làm ngập băng thông khiến cho người dùng không thể truy cập dịch vụ hoặc làm cho dịch vụ hoàn toàn tê liệt vì hết tài nguyên khiến cho người dùng không thể truy cập dịch vụ. Hơn nữa, các kỹ thuật tấn công DDoS khác nhau có thể làm quá tải tới băng thông hoặc bão hòa hệ thống bị tấn công theo những cách khác nhau. Có 3 loại tấn công thường gặp: tấn công băng thông (Volumetric attacks), tấn công giao thức (protocol attacks) và tấn công ứng dụng (application attacks). Trong bài báo này, tác giả tập trung nghiên cứu sự ảnh hưởng của DDoS tới băng thông mạng và đề xuất cải tiến giải thuật liên quan nhằm hạn chế sự ảnh hưởng của DDoS tới băng thông mạng.

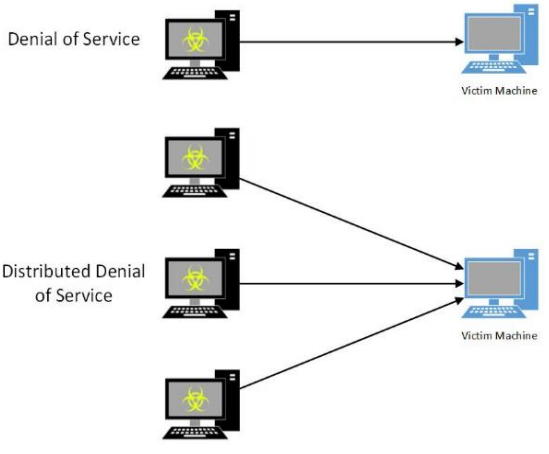
ABSTRACT

The feature of DDoS attack is flooding the bandwidth and preventing users from accessing the service or completely paralyzes the service as running out of resources so that users are not able to access the service. Furthermore, different DDoS attack techniques can overload the bandwidth or saturate the system being attacked in different ways. There are three common types of attacks: Volumetric attacks, Protocol attacks, and Application attacks. In this paper, we focus on studying the influence of DDos on network bandwidth and propose improving related algorithms to limit the influence of DDos on network bandwidth.

1. Giới thiệu

Hiện nay tấn công từ chối dịch vụ DDoS (Distributed Denial of Service) đã biến đổi rất nhiều, từ những kịch bản đơn giản, một đợt tấn công một chiều từ một điểm điều khiển đơn, đến các hình thức tấn công có độ phân tán cao, chia làm nhiều bước, với sự tham gia của rất nhiều nhóm sử dụng các hình thức trao đổi được mã hoá. Các lượt tấn công phối hợp được thực hiện đều đặn bởi rất nhiều nhóm hoạt động cùng nhau từ rất nhiều địa điểm được thực hiện một cách có chủ đích hoặc từ đặc tính của Hijacking

và mạng botnet. Tuy nhiên vẫn chưa có một kỹ thuật cụ thể nào có thể chống lại được hoàn toàn các kiểu tấn công DDoS. Trong đó ***tấn công băng thông mạng*** (Volumetric Attacks) là kỹ thuật thường gặp nhất và cũng là dễ thực hiện nhất. Thường thì những kẻ tấn công sẽ vận dụng các kỹ thuật khuếch đại để sinh ra các yêu cầu (request) mà không cần dùng đến một lượng lớn tài nguyên. Các vụ tấn công khuếch đại tận dụng lượng lớn phản hồi đến những request nhỏ, từ đó khuếch đại lượng request để làm quá tải hệ thống mục tiêu. Quá trình này thường được thực hiện bằng cách giả mạo nguồn của các gói, hay còn gọi là phản xạ hay tấn công phản xạ. Ví dụ, với việc giả mạo nguồn IP của một request DNS, kẻ tấn công có thể lừa máy chủ DNS gửi phản hồi đến mục tiêu thay vì nguồn truyền dữ liệu. Vì request gửi đến máy chủ DNS rất nhỏ nhưng phản hồi gửi đến hệ thống nạn nhân lại lớn nên kẻ tấn công sẽ sử dụng phản xạ để khuếch đại lượng request gửi đến hệ thống này.



Hình 1. Kiểu tấn công DoS và DDoS

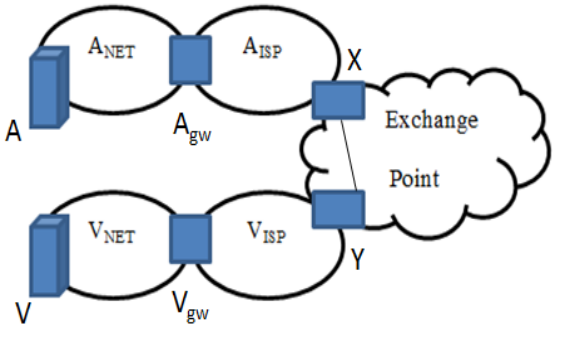
Lấy ví dụ, hãy tưởng tượng bạn đang ở một tòa nhà A đông đúc. Một người nào đó bấm chuông báo cháy và chạy vòng quanh rồi hét lên: Cháy cháy. Ngay lập tức, hàng trăm người sẽ gọi trung tâm cứu trợ 911 cùng lúc đó. Các đường dây đều rung đồng thời và người tiếp nhận sẽ phải chạy đua để trả lời những cuộc gọi đó. Đồng thời, có một vụ cháy thực sự diễn ra ở địa điểm B khác, tuy nhiên những người dân ở đó lại không thể liên lạc được với bộ phận 911 bởi vì họ đang quá bận rộn với những cuộc gọi đến từ tòa nhà A. Tình huống này cũng tương tự như kịch bản tấn công DDoS, khi mà những request hợp lệ bị khóa trong lúc hệ thống đang cố gắng giải quyết một lượng lớn những request có vẻ hợp lệ nhưng thực tế lại là giả mạo. Với ví dụ này, nếu số lượng người trong tòa nhà A đủ để bão hòa đường dây điện thoại khiến những người thực sự muốn gọi điện không thể gọi hoặc chỉ gọi được với chất lượng kém thì đây có thể gọi là một vụ tấn công băng thông. Với việc quan sát được các thiết bị định tuyến trên Internet hiện nay đủ tài nguyên lọc luồng thông tin cần thiết để ngăn chặn các tấn công DDoS, với điều kiện các luồng thông tin tấn công bị chặn ngay gần nguồn phát sinh. Từ đó giao thức Active Internet Traffic Filtering (AITF) đã được đề xuất (Katerina Argyraki & David R. Cheriton, 2005).

Giao thức AITF được phát triển bởi nhóm nghiên cứu hệ thống phân tán trường đại học Stanford nhằm ngăn chặn và phản ứng tức thời với những cuộc tấn công DDoS. Nhóm tác giả đã nghiên cứu và thử nghiệm giao thức AITF với kết quả khá khả quan: AITF có thể ngăn chặn tức thời hàng triệu luồng tấn công trong khi chỉ yêu cầu một sự tham gia của một lượng nhỏ các router. Với AITF giúp nạn nhân chặn các luồng tấn công không mong muốn chỉ trong vài mili giây. Việc ngăn chặn này tuy có hiệu quả cao nhưng lại làm băng thông của mạng sẽ bị cản trở, giảm xuống do dung lượng của các gói tin sẽ bị tăng lên khi đi qua các bộ định tuyến Router do việc viết lên các gói tin về đường dẫn "đi qua" các Router này. Để giải

quyết vấn đề này cần cải tiến giải thuật bằng cách dùng một xác suất ánh xạ giữa địa chỉ IP của Router với trường IP Identification với mục đích loại bỏ việc ghi thông tin về đường dẫn lên gói dữ liệu. Do tính chất của một cuộc tấn công DoS hay DDoS sẽ là tạo ra các gói tin giả mạo thật nhiều và gửi tới nạn nhân, nên việc đánh dấu gói tin với một xác suất và chỉ ghi một thông tin ánh xạ với địa chỉ IP của Router sẽ giúp nạn nhân giảm được băng thông gây nghẽn mạng và vẫn sẽ có thể tìm được đường đi của cuộc tấn công.

2. Mô tả về trường hợp của giao thức AITF

Với A_{NET} là một mạng của người tấn công là A và là nơi xuất phát luồng thông tin không mong muốn tới nạn nhân. Trong mạng này sẽ có một gateway là A_{gw} đây là router tấn công cũng chính là router gần với A nhất (Katerina Argyraki & David R. Cheriton, 2005).



Hình 2. Mô tả các thực thể của kẻ tấn công và nạn nhân

Trong mạng V_{NET} là mạng của nạn nhân là V và là nơi bị luồng thông tin không mong muốn xâm nhập tới thông qua gateway V_{gv}.

Trong hai mạng của nhà cung cấp dịch vụ là A_{ISP} và V_{ISP} còn có các gateway phía trên khi chuyển qua mạng internet là gateway X và gateway Y.

Giả sử khi có một luồng lưu lượng không muốn khi tới nạn nhân và khi đó nạn nhân muốn xác định một lưu lượng không mong muốn, nạn nhân sẽ gửi một yêu cầu lọc đến gateway của nó (V_{gw} trong Hình 2.). Gateway của nạn nhân sẽ tạm thời chặn lại các dòng lưu lượng không mong muốn này và xác định các router gần với nguồn tấn công nhất – gọi là gateway tấn công (A_{gw} trong Hình 2.). Sau đó gateway của nạn nhân sẽ thiết lập một kết nối cài đặt với gateway tấn công, nghĩa là sẽ có một thỏa thuận về truyền các gói tin. Ngay sau khi việc thiết lập kết nối được hoàn thành thì gateway của nạn nhân sẽ có thể bỏ bộ lọc tạm thời của nó đi. Còn nếu không hoàn thành được việc thiết lập kết nối thì gateway của nạn nhân sẽ yêu cầu một gateway khác gần nhất theo phương pháp "leo thang" để làm gateway tấn công (gateway X trong Hình 2.). Việc leo thang có thể đệ quy dọc theo con đường bị tấn công cho đến khi một router được hoàn thành việc kết nối. Nếu không có router phản ứng nào thì lưu lượng giao thông của cuộc tấn công sẽ bị chặn tại gateway của nạn nhân. Tuy nhiên trong giao thức AITF có cả hỗ trợ và thúc đẩy việc các router gần nguồn tấn công giúp chặn các luồng thông tin không mong muốn này.

Hoạt động của AITF trong mạng Internet:

❖ ***Thuật ngữ:***

Đường dẫn P của một luồng thông tin không mong muốn có hình thức như sau: {A A_{gw} X Y V_{gw} V}

Trong đó:

+ A là "nguồn tấn công" nghĩa là nút được cho là tạo ra các dòng lưu lượng không mong muốn tới nạn nhân. Nếu A = * có nghĩa là mọi lưu lượng qua A_{gw} là không mong muốn.

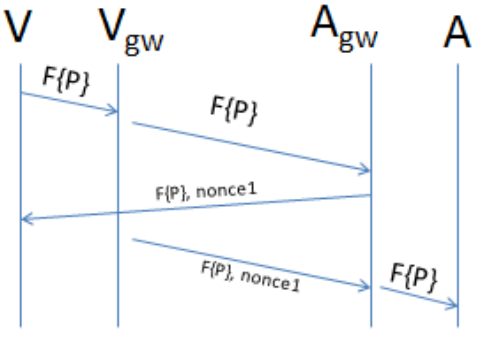
+ A_{gw} là "gateway tấn công" là chỉ router định tuyến gần nhất với nguồn tấn công tức là gấn nhất với A.

+ V_{gw} là "gateway của nạn nhân" tức là router định tuyến gần nhất với nạn nhân.

+ V là nạn nhân.

Ở đây chỉ giả định rằng chỉ nút V là chịu ảnh hưởng của vụ tấn công, tức là nếu đây là một cuộc tấn công làm ngập tràn thì chỉ có phần mạng từ V_{gw} tới V là tắc nghẽn.

❖ ***Chặn nguồn tấn công:***



Hình 3. Các thực thể trao đổi thông điệp

Như thể hiện ở Hình 3. thì AITF liên quan tới 4 thực thể:

- Nạn nhân V khi nhận được một luồng thông tin không mong muốn thì nạn nhân V sẽ gửi yêu cầu lọc tới gateway V_{gw} để chặn luồng thông tin là F.

- Gateway của nạn nhân V_{gw}:

+ Tiến hành cài đặt bộ lọc tạm thời để chặn luồng thông tin F trong khoảng thời gian T_{tmp} giây.

+ Thiết lập việc bắt tay 3 bước với gateway A_{gw}.

+ Bỏ bộ lọc tạm thời sau khi đã bắt tay thành công với gateway tấn công.

- Gateway tấn công A_{gw}:

+ Đáp ứng việc bắt tay 3 bước của gateway nạn nhân.

+ Cài đặt bộ lọc tạm thời chặn luồng thông tin F trong khoảng thời gian T_{tmp} giây sau khi đã hoàn thành việc bắt tay.

+ Gửi một yêu cầu lọc tới nguồn tấn công A dừng luồng thông tin F trong khoảng thời gian Tlong lớn hơn rất nhiều so với T_{tmp} phút.

+ Bỏ bộ lọc tạm thời đi nếu A đáp ứng yêu cầu, còn nếu A không đáp ứng thì ngắt kết nối với A.

- Nguồn tấn công A sẽ phải dừng trong khoảng thời gian Tlong hoặc là bị ngắt kết nối.

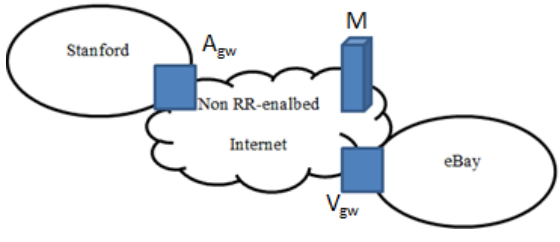
❖ ***Bảo mật giao tiếp:***

Khi V gửi một yêu cầu chặn luồng thông tin không mong muốn F tới gateway của nó là Vgw thì gateway V_{gw} sẽ gửi yêu cầu chặn F tới gateway A_{gw}, sau đó gateway A_{gw} sẽ gửi trả lại thông điệp cho gateway V_{gw} gồm F và một giá trị nonce1 (như Hình 3.) , và sau đó V_{gw} sẽ gửi lại cho A_{gw} để hoàn thành giao tiếp.

nonce1 = hash key F

Với khóa *key* ở đây là khóa cục bộ của hàm băm.

❖ ***Chống việc giả mạo:***



Hình 4. Điểm M giả mạo ở Stanford và gửi lưu lượng không mong muốn đến ebay.

Trong Hình 4. trên làm một ví dụ về điểm M là một điểm nguy hiểm trong mạng. Khi gateway của nạn nhân V_{gw} gửi yêu cầu chặn luồng lưu lượng không mong muốn tới gateway tấn công A_{gw}, mặc dù A_{gw} đã chấp nhận nhưng luồng thông tin tới gateway của nạn nhân V_{gw} vẫn ở mức cao và đến từ Stanford và nó sẽ kết luận sai rằng gateway tấn công là A_{gw} là không hợp tác, lúc này

gateway A_{gw} có thể bị ngắt kết nối tới V_{gw} hoặc là gateway của nạn nhân sẽ liên lạc với gateway tấn công mức trên để chặn luồng thông tin không mong muốn đến từ gateway A_{gw} ở Stanford.

Có một biện pháp có thể khắc phục được trường hợp này đó là khi gói tin đi qua router thì các router sẽ viết thêm vào các gói tin chỗ đường dẫn.

VD: Ghi dạng định tuyến : { * A_{GW} V_{GW} Ebay} nhưng khi thêm giá trị sẽ là { * A_{GW}: R1 V_{GW}: R2 Ebay}.

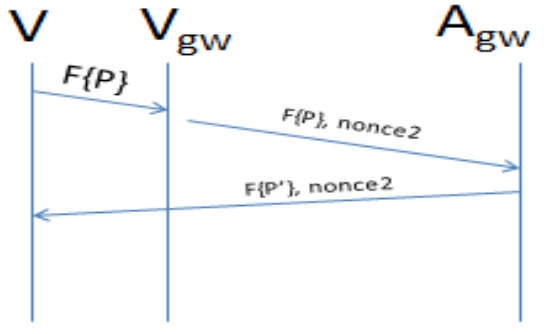
Giá trị R1 và R2 được tính toán như sau:

$R = hash_{key} D$

+ Khóa là khóa cục bộ

+ D gói tin đích.

Và quá trình điền thêm giá trị sau đường dẫn khi đi qua router nếu đúng thì sẽ thực hiện việc bắt tay 3 bước như Hình 3. ở trên. Còn nếu việc điền thêm giá trị đó là một giá trị sai thì sẽ thực hiện được việc bắt tay với 2 bước như hình Hình 5 dưới:



Hình 5. Việc xác thực với giá trị đường dẫn là sai

3. Nghiên cứu đề xuất cải tiến giải thuật giao thức AITF

3.1. Mô tả cải tiến chương trình

Với giao thức AITF thì việc lưu trữ đường đi của gói tin sẽ là dễ dàng nếu việc chuyển các gói tin đi trên các gateway là ngắn, nhưng nếu việc đó trở nên khó khăn khi gói tin đó sẽ phải qua nhiều router và việc ghi vào đường dẫn đó sẽ trở nên rất dài và dung lượng của gói tin sẽ tăng và càng tăng nguy cơ cạn kiệt băng thông của mạng. Nên việc cải tiến của giao thức AITF sẽ đi sâu vào việc làm giảm tải đường dẫn lưu thông tin của các nơi router mà gói tin đi qua và việc tăng kích thước của gói tin sẽ không còn là đáng kể nữa (Moti Geva, Amir Herzberg & Yehos Gev, 2014) (Saman Taghavi Zargar, James Joshi, Member & David Tippe, 2013) (D. G. Andersen. 2003) (T. Anderson, T. Roscoe, and D. Wetherall. 2003) (P. Ferguson and D. Senie. 2000) (A. Garg and A. L. N. Reddy. 2002).

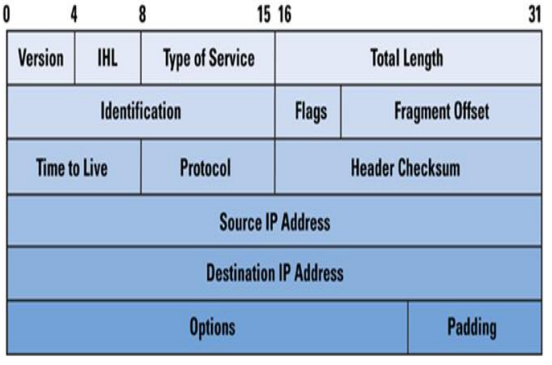
Bằng việc cài đặt một xác suất trên router mà gói tin sẽ đi qua có được đánh dấu hay không, với việc đặt một xác suất này thì gói tin sẽ chỉ phải lưu đúng một nơi bất kỳ trên đường mà gói tin đi qua, vì bản chất của việc tấn công DDoS là sẽ phải gửi nhiều các gói tin giả mạo thì sẽ thành công. Để lần ngược lại nơi bắt đầu tấn công, có thể lấy thông tin từ các router trên đường luồng dữ liệu tấn công "đi qua", khi đó sẽ cho phép tại đích đến (chính là nạn nhân) có thêm thông tin để dựng lại đường đi của luồng tấn công, qua đó có thể thực sự tìm ra nguồn gốc tấn công (D. Dean, M. Franklin, and A. Stubblefield. 2001).

3.2. Phương pháp và công việc cải tiến

- Phương pháp đánh dấu gói tin theo xác suất:

Một xác suất p được định nghĩa tại tất cả các router, mỗi gói tin sẽ được đánh dấu với thông tin thêm bằng cách sử dụng giá trị của p. Đường đi của tấn công sẽ được xây dựng lại bằng cách theo dõi ngược các gói tin IP đã được đánh dấu này. Để tăng thêm hiệu quả bằng cách đánh không cố định mà được hiệu chỉnh theo xắc suất. Như vậy vấn đề đặt ra ở

đây là việc đánh dấu sẽ diễn ra như thế nào, và phần nào trong khuôn dạng của một gói tin IP sẽ được "đánh dấu"? Header của một gói tin IPv4 có khuôn dạng như sau:



Hình 6. IP Header

Trường IP Identification là trường để xác định và chủ yếu là để xác định các phần của các phân đoạn của một IP datagram gốc. Trường IP Identification có độ dài 16 bits. Đối với mục đích dò ngược, chúng ta cần sử dụng vừa vặn 16 bits này cho giá trị "đánh dấu" và giá trị "khoảng cách". Thực tế cho thấy hầu hết đường đi trên Internet của các gói tin đều không quá 30 bước truyền. Do đó việc sử dụng 5 bit (tương ứng 32 bước truyền) để lưu thông tin khoảng cách của gói tin đến nơi xuất phát của gói tin. Còn 11 bits còn lại (có thể cung cấp 211 = 2048 giá trị có thể) sẽ được sử dụng cho việc đánh dấu gói tin qua router. Trong nghiên cứu này thì không ghi vào trường IP Identification mà ghi vào trường Options (Jelena Mirkovic, Janice Martin & Peter Reiher, 2004).

Một hàm băm sẽ được sử dụng là hàm băm h(.), với hàm băm này chúng ta sẽ ánh xạ một địa chỉ IP của router sẽ được đánh dấu trên gói tin với 11 bits giá trị đánh dấu. Hàm thống kê này là một hàm thống kê ngẫu nhiên đáng tin cậy, có nghĩa là đối với một địa chỉ IP bất kỳ nào thì tất cả 2^{11} = 2048 giá trị có thể đánh dấu làm đầu ra. Giả sử độ dài đường đi của một cuộc tấn công là k. Điều đó có thể cho phép nói rằng có k router tham gia vào lược đồ giữa điểm xuất phát và đích đến.

Giá trị xác suất đánh dấu sẽ là:

$$P_d = \frac{1}{d-1+c} \qquad (1)$$

trong đó *d – 1* là giá trị trường khoảng cách của gói tin được nhận từ một router cách *d* bước truyền so với nguồn của tấn công, chúng ta cần đảm bảo các giá trị xác xuất luôn nhỏ hơn hoặc bằng 1. Do đó *c* là giá trị trường khoảng cách của gói tin được nhận từ một router cách d bước truyền so với nguồn của tấn công, chúng ta cần đảm bảo các giá trị xác xuất luôn nhỏ hơn hoặc bằng 1. Do đó $c \geq 1$, $c \in R$.

Gọi α_d là xác suất mà nạn nhân nhận được một gói tin đã đánh dấu bởi một router cách *d* bước truyền từ kẻ tấn công. Khi đó:

$$\alpha_d = P_d . \Pi_{i=d+1}^{k}(1 - p_i) \qquad (2)$$

Kết hợp công thức trên ta sẽ có:

$$\alpha_d = \frac{1}{k-1+c} \qquad (3)$$

Do đó ta thấy rằng xác suất của việc nhận một gói tin đã được đánh dấu bơi bất kỳ router nào dọc đường đi của tấn công sẽ phụ thuộc vào độ dài của đường đi chứ không phụ thuộc vị trí của router.

- Thuật toán đánh dấu gói tin:

Một router dọc theo đường đi của một gói tin sẽ đọc giá trị khoảng cách trong trường IP Identification. Sau đó router sẽ tìm đến bảng chứa các giá trị khoảng cách và xác suất đánh dấu tương ứng. Quyết định sẽ được thực hiện như sau: Router sẽ sinh ra một số ngẫu nhiên, nếu số ngẫu nhiên này nhỏ hơn hoặc bằng xác suất thì gói tin sẽ được đánh dấu, và sẽ ghi giá trị của hàm băm *h* (địa chỉ IP) vào trường IP

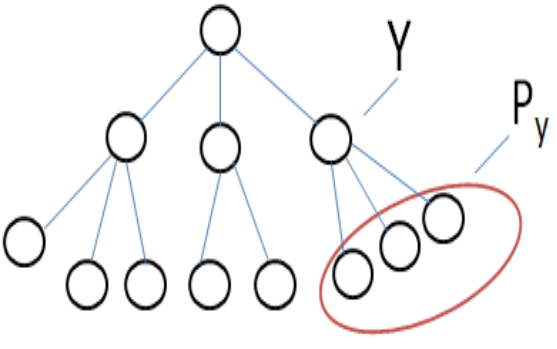
Identification. Giá trị khoảng cách trong trường IP Identification khi đó sẽ tăng thêm và gói tin sẽ được định tuyến. Kể cả trường hợp quyết định không đánh dấu gói tin nhưng nó vẫn luôn tăng giá trị khoảng cách trong trường IP Identification, và gói tin vẫn được định tuyến.

Thuật toán đánh dấu gói tin như sau:

m = h (địa chỉ IP)
for each gói tin
read d= giá trị của trường khoảng cách
sinh ra một số ngẫu nhiên x ∈ [0, 1]
p = xác suất đánh dấu tương ứng với d
if x ≤ p (nếu xảy ra, gói tin được đánh dấu)
write m vào trường đánh dấu
giá trị trường khoảng cách = d + 1

- Xây dựng lại đường đi của tấn công:

Để xây dựng lại đường đi của một gói tin và xác định nguồn gốc của tấn công, nạn nhân cần một bản đồ các router. Nạn nhân sẽ so khớp với các dấu của gói tin với các router trên bản đồ, đi qua đó có thể xây dựng lại được đường đi của gói tin của kẻ tấn công. Bản đồ này được xem như một đồ thị có hướng G. Gốc của G là nạn nhân, tất cả các đỉnh trong G sẽ là các router, mỗi router y trong G sẽ bao gồm tập hợp y các con của nó.



Hình 7. y và p_y trong đồ thị G

Trong suốt quá trình diễn ra tấn công DDoS, nạn nhân sẽ nhận một lượng lớn các dấu từ các router. Trước khi xây dựng lại đường đi dựa trên các dấu này, chúng ta cần phân nhóm các dấu dựa trên độ dài đường đi của tấn công. Giả sử có n kẻ tấn công (tấn công từ chối dịch vụ phân tán) ở những khoảng cách khác nhau so với nạn nhân. Trong trường hợp này, nạn nhân sẽ có các tập hợp khác nhau các dấu, mỗi tập hợp sẽ chứa các dấu từ các kẻ tấn công có cùng khoảng cách đến nạn nhân. Đặt các giá trị giờ đây là |μ| tập hợp khác nhau của các dấu, mỗi tập tương ứng cho các giá trị của trường khoảng cách sẽ là $0 \le k \le 31$. Gọi tập các dấu nhận bởi nạn nhân với giá trị khoảng cách $k \in \mu$ là λ_k. Ký hiệu số kẻ tấn công tại khoảng cách k bước truyền là n_k. Khi đó ta sẽ có:

$$\lambda_k = n_k . k \qquad (4)$$

Bây giờ ta sẽ xem xét thuật toán xây dựng lại đường đi của tấn công. Đồ thị G được duyệt qua bởi mỗi tập các gói tin có cùng giá trị trường khoảng cách cho mỗi tập λ_k, $\forall_k \in \mu$). Bắt đầu tại điểm gốc của đường tấn công là nạn nhân. Các dấu của "láng giềng" con với nạn nhân được kiểm tra với mỗi dấu trong tập. Các dấu của các router mà đã so khớp sẽ được thêm vào đồ thị tấn công. Tiếp tục lặp lại như vậy , "con" của router sẽ được kiểm tra với kiểu cách tương tự. Quá trình này xảy ra cự ly lặp lại cho đến khi chiều sâu của đồ thị bằng với đường đi. Đường đi của tấn công sẽ được đưa trong S_d, với $0 \le d \le k$.

Thuật toán xây dựng lại đường đi được thực hiện như sau:

$\forall k \in \mu$
S_0 = nạn nhân
for d = 0 to (k -1)
$\forall y$ *in* S_d
$\forall R \in \rho_y$
if $R \in \lambda_k$ *then*
insert $R \rightarrow S_{d+1}$
output S_d
output S_k

3.3 Giải thuật của thuật toán cải tiến

Với giải thuật xây dựng lại đường đi, khi một gói tin chưa được đánh dấu tại router nào khi nó đi qua một router thì nó sẽ có một xác suất đánh dấu gói tin mà xác suất này thỏa mãn việc đánh dấu gói tin sẽ được đánh dấu và sẽ lưu lại khoảng cách của router đến với từ nguồn xuất phát gói tin (người tấn công) vào 5 bits đầu của trường Identification trong header gói tin IP, 11 bits còn lại của trường này sẽ được làm đánh dấu với việc ánh xạ từ địa chỉ IP của route. Với một cuộc tấn công DoS hay DDoS mà xảy ra thì lượng gói tin sẽ đến nạn nhân sẽ rất nhiều nên việc tất cả router trên đường đi của gói tin sẽ có khả năng được đánh dấu hết. Nên việc dựng lại đồ thị và dò ra được đường đi của gói tin xuất phát từ nơi tấn công là có thể. Khi chúng ta đã biết được đường đi của gói tin rồi thì phần việc còn lại sẽ là phần việc chính của AITF đó là gateway tấn công sẽ phải được yêu cầu chặn luồng thông tin không mong muốn này từ nơi xuất phát và có thể ngắt kết nối với nơi tấn công.

4. Kết quả thử nghiệm

4.1 Cài đặt

Chương trình giải thuật cải tiến phần đánh dấu lại gói tin được viết với một file MarkPacket.c

Việc biên dịch được thực hiện trên máy Red Hat 4.1.1-52 Linux version 2.6.18-8.el5 Với phiên bản của gcc 4.1.1.20070105

Biên dịch với lệnh

gcc MarkPacket.c –o MarkPacket

4.2 Chạy chương trình

Chương trình được chạy trên một máy ảo Red Hat 4.1.1-52 Linux version 2.6.18-8.el5, trong khi có một terminal khác để chạy một chương trình Client/Server để nhằm mục đích tạo ra một gói tin (packet) được chuyển qua card mạng, còn lại một terminal chính ta sẽ chạy chương trình đánh dấu gói tin (Đào Đình Thái, 2010).

Trong trường hợp cụ thể sau đây thì chương trình Client/Server đã kết nối với nhau và truyền một thông tin với nội dung là "*test*" tới một địa chỉ đích là "192.168.1.2".

Bây giờ thử chạy chương trình đánh dấu gói tin:

./Markpacket eth1 eth0 192.168.1.2

+ Với eth1 là tham số muốn bắt các gói tin trên card mạng eth1

+ Với eth0 là tham số chúng ta muốn chuyển các gói tin trên card mạng eth0

+ Với tham số thứ 3 là 192.168.1.2 là đích cần chuyển gói tin đến.

Với các thông tin của gói tin nhận được là:

```
[root@thai2805 KLTN_DaoDinhThai]# ./MarkPacket eth1 eth0 192.168.1.2
Destination MAC: 00 0C 29 91 F7 EC
Source MAC: 00 0C 29 79 BA 5D
Protocol: 08 00
----------------------
type of service         : 0
total length            : 56325
identification          :0010010011001111
fragment offset         : 32
time to live            : 64
protocol                : 17
checksum                : 51501
source address          : 172.16.0.2
dest address            : 172.16.0.1
Options                 : 17767
Distance                : 8
Options (bin)           : 0100010101100111
----------------------
Hash (IP): 769
```

Hình 8. Thông tin gói tin đến

Với thông tin ở trên ta có địa chỉ đích đến là 172.16.0.1 và nguồn là 172.16.0.1 với khoảng cách của gói tin cách nơi xuất phát của gói tin được ghi trên gói tin là 8. Và mã của hàm băm IP của địa chỉ đích là 769 theo cơ số 10. Sau khi chúng ta phân tích và sửa lại gói tin trước khi chuyển đến một địa chỉ khác thì gói tin được đánh dấu với thông tin như sau:

```
Destination MAC: 00 0C 29 91 F7 EC
Source MAC: 00 0C 29 79 BA 5D
Protocol: 08 00
----------------------
type of service         : 0
total length            : 56341
identification          :0010010011001111
fragment offset         : 32
time to live            : 64
protocol                : 17
checksum                : 51501
source address          : 172.16.0.2
dest address            : 192.168.1.2
Options                 : 19201
Distance                : 9
Options (bin)           : 0100101100000001
----------------------
Packet sent successfully
```

Hình 9. Thông tin gói tin đi

Với thông tin trên là gói tin đã được chỉnh sửa thì gói tin đã được thay đổi trường IP Identification với khoảng cách là 9, tăng một bước truyền so với gói tin đến và địa chỉ đến đã được đổi thành 192.168.1.2 và nó được chuyển đi thành công. Vì vậy, với việc thực hiện trên việc cải tiến giải thuật giao thức AITF để tránh hiện tượng tăng tải của mạng lên là một việc có thể thực hiện được.

5. Kết luận

Với giải pháp lọc luồng thông tin trên mạng sử dụng giao thức AITF đã góp phần ngăn chặn/chống tấn công từ chối dịch vụ DDoS. Giao thức AITF làm cho nạn nhân có thể tự nhận ra cuộc tấn công và ngăn chặn một luồng thông tin không mong muốn xâm nhập tới chính nó. Trong khi AITF vẫn còn nhược điểm là chiếm dụng băng thông do việc lưu dữ liệu đường dẫn vào gói tin nên làm tăng kích thước gói tin, thì đã có một cải tiến với việc lưu đường dẫn trên gói tin. Sự thay đổi cách lưu này làm kích thước gói tin không tăng lên là bao nhiêu so với gói tin ban đầu, nên việc tăng tải băng thông của mạng không còn là vấn đề. Với việc cải tiến giải thuật giao thức mạng AITF này đã làm mở rộng một bước phát triển trong việc ngăn chặn các cuộc tấn công từ chối dịch vụ và tấn công từ chối dịch vụ phân tán. Trong nghiên cứu này tuy đã có những thành công bước đầu về giải pháp chống tấn công từ chối dịch vụ, nhưng nó vẫn còn gặp một số yếu điểm là phải cần một số lượng gói tin lớn để cho việc tìm ra nguồn tấn công. Trong tương lai nghiên cứu sẽ cố gắng phát triển để việc tìm ra đường đi và nguồn gốc của cuộc tấn công ở mức độ sớm và hướng tới mục đích ngăn chặn triệt để các cuộc tấn công từ chối dịch vụ để DDoS không còn đáng lo ngại nữa trong cộng đồng internet như hiện nay.

TÀI LIỆU THAM KHẢO

Katerina Argyraki and David R. Cheriton. (2005). Active Internet Traffic Filtering

Real-Time Response to Denial-of-Service Attacks. *05 Proceedings of the annual conference on USENIX Annual Technical Conference*. Stanford University.

Moti Geva, Amir Herzberg and Yehos Gev. (2014). Bandwidth Distributed Denial of Service: Attacks and Defenses. *IEEE Security & Privacy, 12 (1)*, 54-61.

Jelena Mirkovic, Janice Martin and Peter Reiher. (2004). A Taxonomy of DDoS Attacks and DDoS Defense Mechanisms. *ACM SIGCOMM Computer Communication Review, 34(2)*.

Đào Đình Thái. (2010). Cải tiến giao thức AITF để giảm tải mạng. *Khóa luận tốt nghiệp ĐH, Công nghệ thông tin*. Trường ĐH Công nghệ, ĐH Quốc gia Hà Nội.

Saman Taghavi Zargar, James Joshi, Member and David Tippe. (2013). A Survey of Defense Mechanisms Against Distributed Denial of Service (DDoS) Flooding Attacks. *IEEE Communications Surveys & Tutorials*, *15(4)*, 2046 – 2069.

D. G. Andersen. (2003). Mayday: Distributed filtering for internet services. *In In Proceedings of 4th Usenix Symposium on Internet Technologies and Systems*.

T. Anderson, T. Roscoe, and D. Wetherall. (2003). Preventing internet denial-of-service with capabilities. *In In Proceedings of HotNets II.*

D. Dean, M. Franklin, and A. Stubblefield. (2001). An algebraic approach to IP Traceback. *In Proceedings of the 2001 Network and Distributed System Security Symposium*.

P. Ferguson and D. Senie. (2000). *Network Ingress Filtering: Defeating Denial of Service Attacks which Employ IP Source Address Spoofing*. RFC 2827.

A. Garg and A. L. N. Reddy. (2002). Mitigation of DoS attacks through QoS Regulation. *In Proceedings of IWQOS workshop.*