TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

BÁO CÁO MÔN HỌC

Môn: Nhập môn An toàn Thông tin

Tìm hiểu về phát hiện tấn công quét cổng

Giảng viên hướng dẫn: PGS. TS. Nguyễn Linh Giang

Nhóm sinh viên thực hiện:

Lê Đức Huy MSSV: 20141938

Bùi Văn Hạnh MSSV: 20173098

HÀ NỘI, 6/2020

MỤC LỤC

CHƯƠNG 1. ĐẶT VẤN ĐỀ 1

CHƯƠNG 2. CƠ SỞ LÝ THUYẾT 2

2.1 Cơ sở lý thuyết về mạng 2

2.1.1 Mô hình mạng TCP/IP 2

2.1.2 Cổng ứng dụng (port) 5

2.2 Tấn công quét mạng 5

2.2.1 Tấn công Thăm dò (Probe Attack) 6

2.2.2 Tấn công quét cổng 6

2.2.3 Các công cụ tấn công 9

2.3 Hệ phống phát hiện xâm nhập (IDS) 10

2.3.1 Khái niệm 10

2.3.2 Phân loại IDS 11

2.3.3 Signature-based IDS (SIDS) 12

2.3.4 Anomaly-based IDS 13

CHƯƠNG 3. PHÁT HIỆN TẤN CÔNG QUÉT CỔNG 14

3.1 Phát hiện áp dụng thuật toán 14

3.2 Phát hiện sử dụng ngưỡng (Threshold-based IDS) 14

3.3 Phát hiện sử dụng luật (Rule-based IDS) 16

3.4 Phát hiện sử dụng học máy (MachineLearning-based IDS) 17

CHƯƠNG 4. MỘT SỐ CHƯƠNG TRÌNH THỬ NGHIỆM 18

4.1 Snort 18

4.1.1 Mô hình cài đặt 18

4.1.2 Kết quả 18

4.2 Hệ thống phát hiện xâm nhập sử dụng học máy trên mạng SDN 19

4.2.1 Mạng SDN 19

4.2.2 IDS trên mạng SDN 20

4.2.3 Xây dựng SDN-IDS 21
4.2.4 Thu thập dữ liệu và học máy 22
4.2.5 Áp dụng vào SDN-IDS 23
4.2.6 Kết quả 23
CHƯƠNG 5. KẾT LUẬN 25

DANH MỤC HÌNH ẢNH

Hình 2.1 Mô hình mạng TCP/IP 2
Hình 2.2 Bắt tay ba bước trong giao thức TCP 3
Hình 2.3 Khuôn mẫu gói tin TCP 5
Hình 2.4 Idle Scan 8
Hình 2.5 Giao diện dòng lệnh của nmap 9
Hình 2.6 Quét cổng cùng netcat 10
Hình 2.7 Hoạt động của IDS 11
Hình 2.8 IDS trên mạng và IDS trên máy 11
Hình 3.1 Xây dựng IDS theo hướng sử dụng ngưỡng 15
Hình 3.2 Một mô hình phát hiện tấn công trên một Rule-based IDS với snort ... 16
Hình 4.1 Mạng truyền thống và mạng SDN 19
Hình 4.2 Kiến trúc của mạng SDN 20
Hình 4.3 Thiết lập kênh giao tiếp trong giao thức OpenFlow 21
Hình 4.4 Mô hình thu thập dữ liệu cho học máy 22
Hình 4.5 Mô hình hoạt động của SDN-IDS 23

DANH MỤC BẢNG

Bảng 2.1 So sánh TCP và UDP 4

Bảng 2.2 Các ứng dụng và cổng dịch vụ phổ biến 7

Bảng 4.1 Kết quả của một số mô hình học máy 23

CHƯƠNG 1. ĐẶT VẤN ĐỀ

Tấn công thăm dò là một trong những bước quan trọng đối với một kẻ tấn công trong việc tấn công vào một hệ thống cụ thể nào đó. Với mục tiêu dò tìm được những thông tin quan trọng, như mô hình mạng, thông tin dịch vụ, các lỗ hổng nền, các hacker có thể rút ngắn thời gian tấn công và đem lại hiệu quả cao hơn. Một trong những thông tin quan trọng của hệ thống mà kẻ tấn công muốn tìm ra là các dịch vụ đang hoạt động trong hệ thống. Để làm được điều đó, cách phổ biến nhất các hacker thường sử dụng là tấn công quét cổng dịch vụ.

Hệ thống phát hiện xâm nhập là một hệ thống theo dõi trạng thái, lưu lượng mạng, sau đó phân tích trên một chiến lược cụ thể để đưa ra kết quả xem có tấn công xuất hiện trong mạng hay không, tiếp đó có thể cảnh báo với người dùng. Với một số dịch vụ không thể có chiến thuật để ngăn chặn hoàn toàn, như các kết nối đến cổng dịch vụ, thì một hệ thống phát hiện xâm nhập hiệu quả sẽ đưa ra cảnh báo sớm về những tấn công quét cổng đang xảy ra trong hệ thống. Nhờ đó, người dùng sẽ có những kế hoạch ngăn chặn tấn công tiếp theo.

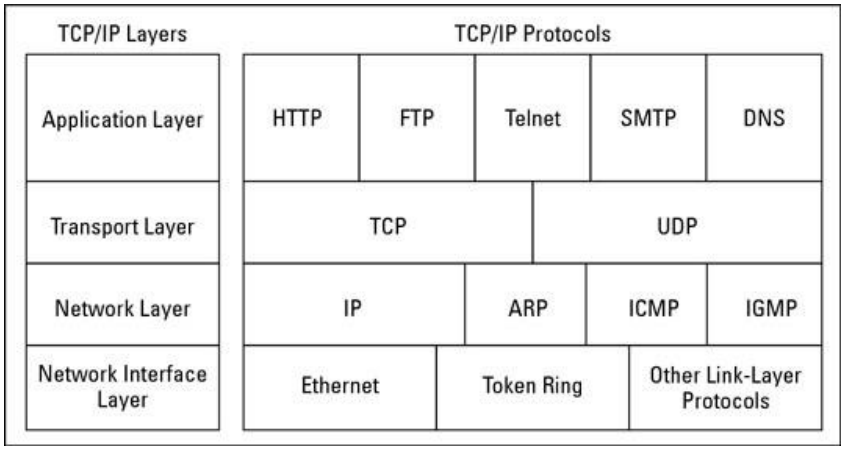
Trong khuôn khổ của đề tài này, nhóm chúng em đã tìm hiểu nội dung lý thuyết về tấn công quét cổng và các phương thức phát hiện ra kiểu tấn công này, đồng thời xây dựng một số mô hình phát hiện thực tế.

CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

2.1 Cơ sở lý thuyết về mạng

2.1.1 Mô hình mạng TCP/IP

Mô hình TCP/IP [1] [2], hay còn được gọi là mô hình Internet, là mô hình mạng máy tính phổ biến, được sử dụng rộng rãi trên hệ thống mạng máy tính trên toàn thế giới, được đơn giản hóa từ mô hình OSI 7 lớp truyền thống.



Hình 2.1 Mô hình mạng TCP/IP

Mô hình TCP/IP gồm 4 tầng (Hình 2.1 Mô hình mạng TCP/IP), có nền tảng cốt lõi là giao thức IP ở tầng mạng (Network Layer) và cặp giao thức TCP/UDP ở tầng giao vận (Transport Layer).

Pipeline của mô hình TCP/IP tương tự như mô hình OSI, bắt đầu từ tầng Ứng dụng, các dịch vụ đóng gói theo giao thức của mình, qua từng tầng phía dưới, ở mỗi tầng đều được đóng gói theo giao thức tương ứng. Khi đến địa chỉ, gói tin lại tiếp tục được bóc tách ở từng tầng, nếu không có lỗi sẽ đến được tầng Ứng dụng của phía nhận.

2.1.1.1. Tầng Ứng dụng (Applicaion Layer)

Là tầng trên cùng, có nhiệm vụ đặc tả giao diện, nội dung chương trình, cách thức biểu diễn, mã hóa thông tin, các khuôn dạng thông điệp trong chương trình. Một số giao thức tầng ứng dụng phổ biến có thể kể đến như:

- SNMP (Simple Network Management Protocol): Giao thức quản lý mạng
- HTTP(s) (Hypter-Text Transfer Protocol (với SSL/TLS)): Là giao thức được sử dụng trong các dịch vụ Web, trao đổi dữ liệu giữa web browser và server. Giao thức HTTP trao đổi thông tin không mã hóa, do đó hiện nay,

chủ yếu dịch vụ web sử dụng HTTPs, chính là HTTP + thêm giao thức SSL/TLS để trao đổi khóa và thống nhất khóa phiên cho dữ liệu.

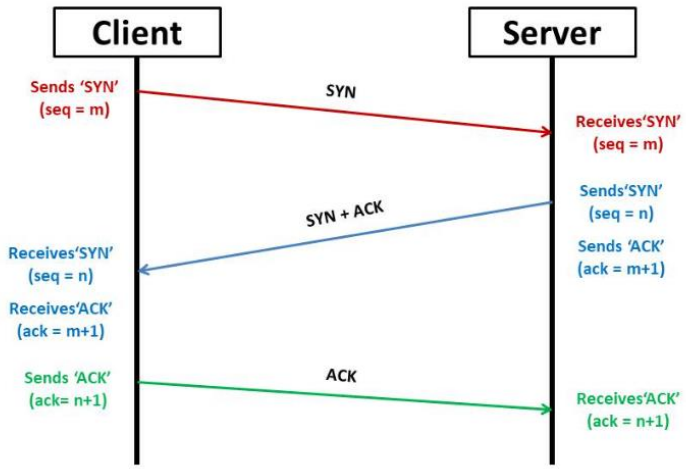
- FTP (File Transfer Protocol): Giao thức truyền file
- Telnet: Giao thức điều khiển từ xa, thường được sử dụng để kết nối đến một server qua môi trường mạng, sau đó thực hiện điều khiển trên giao diện command-line
- SSH: Tương tự như Telnet, tuy nhiên SSH có bổ sung thêm mã hóa thông tin truyền, đảm bảo an toàn hơn so vs Telnet.
- SMTP (Simple Mail Transfer Protocol): Giao thức được sử dụng để gửi mail giữa các mail server. Tuy nhiên không cung cấp giao diện cho người dùng có thể lấy mail từ mail server để đọc/lưu trữ.
- POP3: Là giao thức giải quyết vấn đề trên cho SMTP, được sử dụng để lấy mail có trong server.
- DNS: Giao thức phân giải tên miền

2.1.1.2. Tầng giao vận (Transport Layer)

Tầng nằm giữa tầng Mạng và tầng Ứng dụng, có nhiệm vụ cung cấp đường truyền giữa các hai tiến trình hoạt động trên 2 máy tính khác nhau. Địa chỉ của hai tiến trình này được xác định bởi một giá trị gọi là cổng dịch vụ (port).

Như đã nói ở trên, mô hình TCP/IP có nền tàng là cặp giao thức TCP/UDP ở tầng giao vận. Đây là hai giao thức chính, phần lớn các dịch vụ tầng ứng dụng đều sử dụng một trong hai (hoặc cả hai) giao thức này.

- **TCP (Tranmission Control Protocol)**: Đây là giao thức hướng kết nối (connection-oriented), tức là trước khi trao đổi thông tin giữa hai tiến trình, thì trước tiên cần thiết lập kết nối qua quá trình bắt tay ba bước



Hình 2.2 Bắt tay ba bước trong giao thức TCP

Mục đích của quy trình này là để xác thực lẫn nhau, sau đó thiết lập một kênh truyền an toàn, có khả năng giám sát kênh truyền, có nhiệm vụ phát hiện lỗi và đảm bảo tin cậy, tức có khả năng phát hiện và gửi lại gói tin hỏng, mất mát.

- **UDP (User Datagram Protocol)**: Ngược lại với TCP, UDP không yêu cầu thiết lập kênh truyền trước khi trao đổi dữ liệu. Với đặc tính này, 2 tiến trình sẽ không thể phát hiện ra gói tin bị mất mát trong quá trình truyền tin, tuy nhiên lại đảm bảo tốc độ nhanh, UDP Server có khả năng phục vụ cho nhiều client khác nhau, vì cơ chế chỉ dựa trên request-reply mà không cần giữ kết nối.

TCP	UDP
Hướng kết nối	Không kết nối
Độ tin cậy cao	Độ tin cậy thấp
Kiểm soát mất mát gói tin	Không kiểm soát mất mát gói tin
Số lượng client giới hạn	Số lượng client gần như không giới hạn
Có cơ chế kiểm soát tắc nghẽn	Không kiểm soát tắc nghẽn
Sắp xếp thứ tự gói tin	Không sắp xếp

Bảng 2.1 So sánh TCP và UDP

2.1.1.3. Tầng mạng (Network Layer)

Tầng mạng có nhiệm vụ đảm bảo truyền tin host-host, tức có nhiệm vụ tìm đường đi giữa hai host. Giao thức chính của tầng mạng là giao thức IP. Có nhiệm vụ định địa chỉ, tìm đường trên mạng (routing). Các giao thức tầng mạng không đảm bảo độ tin cậy, tức không có cơ chế phát hiện và gửi lại gói tin mất mát. Ngoài IP, còn có một số giao thức khác như ICMP, được sử dụng để thông báo lỗi và ARP, giao thức phân giải đia chỉ IP sang địa chỉ MAC (cho mạng local)

2.1.1.4. Tầng Data-link (Data-Link Layer)

Nhiệm vụ của tầng này là gửi dữ liệu thô từ điểm đến điểm. Một số giao thức có thể kể đến như:

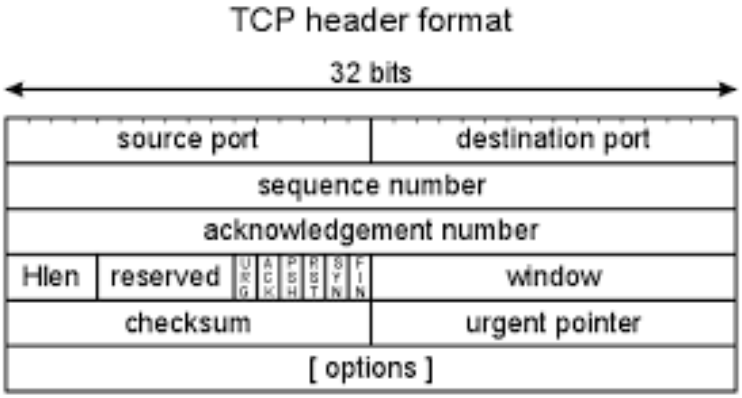
- Ethernet

- PPP
- Token Ring
- Wifi

2.1.2 Cổng ứng dụng (port)

Chúng ta đều biết, mỗi máy tính có một địa chỉ IP xác định. Như vậy nhờ các thuật toán routing ở tầng mạng, các gói tin có thể được chuyển đến đúng địa chỉ (thiết bị) cần đến. Tuy nhiên, một máy tính có thể chạy, sử dụng rất nhiều các dịch vụ khác nhau ở tầng ứng dụng. Do đó khái niệm cổng dụng vụ (port) ra đời.

Các gói tin IP sau khi được bóc tách sẽ được chuyển lên tầng giao vận (transport) để tiếp tục xử lý. Tại đây, dựa vào thông số port (nằm trong header của phần gói tin của giao thức TCP hoặc UDP), mà gói tin sẽ được chuyển đến đúng tiến trình đang hoạt động dịch vụ tương ứng.



Hình 2.3 Khuôn mẫu gói tin TCP

Như vậy có thể nói, port là điểm cuối trong kết nối mạng, là địa chỉ của một tiến trình mạng đang hoạt động tại một host. Như Hình 2.3 Khuôn mẫu gói tin TCP có thể thấy, thông số port có độ dài 16 bits, như vậy với mỗi thiết bị, có tối đa 65536 ports, được chia làm ba nhóm: (i) các cổng dịch vụ phổ biến (0-1023), (ii) các cổng dịch vụ được đăng ký (bởi các nhà phát triển phần mềm phổ biến) (1024-49151) và (iii) cổng cho những dịch vụ tạm thời hoặc cho các ứng dụng đặc thù (49152-65535). Mỗi dịch vụ trên tầng ứng dụng có thể sử dụng từ một đều nhiều port để kết nối, trao đổi thông tin.

2.2 Tấn công quét mạng

2.2.1 Tấn công Thăm dò (Probe Attack)

Tấn công thăm dò là loại hình tấn công nhằm vào việc do thám, dò tìm các thông tin của mạng hoặc các máy tính trong mạng. Với mỗi tầng trong mô hình TCP/ IP, cơ chế và đối tượng thăm dò lại khác nhau.

Đối với tầng ứng dụng, mục tiêu thăm dò thông thường là các thông tin chung của ứng dụng, máy chủ (phiên bản dịch vụ, hệ điều hành, cơ sở dữ liệu…), nhờ đó có thể dựa vào những lỗ hổng đã được phát hiện trên các thông tin cơ bản thu được và lập kịch bản tấn công. Ngoài ra, tấn công thăm dò vào các ứng dụng còn có thể là các chương trình, công cụ kỹ thuật cao (ví dụ như sqlmap [3]), để quét các lỗ hổng tồn tại trong hệ thống.

Trong tầng giao vận, với hai giao thức chính được sử dụng ở tầng này là TCP và UDP, thông thường mục tiêu sẽ là kiểm tra các cổng dịch vụ đang được mở bởi host với hai giao thức này.

Với tầng mạng, kẻ tấn công thông thường chỉ muốn thăm dò các host đang hoạt động trong một hệ thống nào đó.

Như vậy, tấn công thăm dò mục đích chính là thu thập thông tin, kiểm tra hoạt động của các thực thể trong mạng, thường được sử dụng để phục vụ làm cơ sở cho các hình thức tấn công khác. Trong phạm vi của đề tài, có hai hình thức tấn công được nghiên cứu tới là Port scan – tấn công quét cổng.

2.2.2 Tấn công quét cổng

Port Scan [4], tấn công quét cổng dịch vụ, là hình thức tấn công ở tầng giao vận. Ở tầng giao vận, có hai giao thức chính là Tranmission Control Protocol (TCP) và User Datagram Protocol (UDP). Như đã đề cập đến ở trên, mỗi dịch vụ trên tầng ứng dụng sẽ có một (hoặc nhiều) cổng dịch vụ nhất định.

Mục tiêu của kẻ tấn công đối với hình thức tấn công này là nhằm phát hiện những cổng dịch vụ đang được mở với một địa chỉ IP xác định (tức có dịch vụ nào đang chạy không). Đây là hình thức tấn công phổ biến, có thể là bước tiếp theo sau bước tấn công IP Sweep (đã thu được các host hoạt động), và cũng có thể là nền tảng cho các hình thức tấn công tiếp theo trực tiếp vào một dịch vụ nào đó trên hệ thống. Thông thường, các cổng dịch được nhắm đến là những cổng mặc định của các dịch vụ phổ biến trên môi trường mạng (Bảng 2.2 Các ứng dụng và cổng dịch vụ phổ biến). Ngoài ra, kẻ tấn công cũng có thể quét một dải cổng bất kỳ để thu thập thêm thông tin, hoặc chỉ một số cổng cụ thể phụ thuộc vào mục đích thăm dò.

Cổng	Giao thức tầng giao vận	Dịch vụ
20, 21	TCP	FTP
25	TCP	SMTP
22	TCP	SSH
23	TCP	Telnet
53	UDP	DNS
80	TCP	HTTP
110	TCP	POP3
123	UDP	NTP
161,162	TCP/UDP	SNMP
443	TCP/UDP	HTTPS
520	UDP	RIP
1194	TCP/UDP	OpenVPN

Bảng 2.2 Các ứng dụng và cổng dịch vụ phổ biến

Dựa vào loại giao thức tầng giao vận (TCP hay UDP), mà cách thức và dấu hiệu nhận biết, thu thập thông tin là khác nhau:

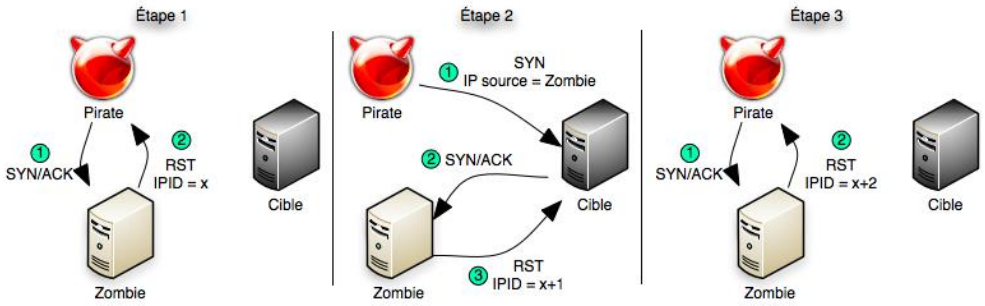
- **TCP**: Phổ biến nhất, kẻ tấn công thực hiện gửi một gói tin TCP (flag SYN), được sử dụng để bắt đầu một kết nối TCP, đến từng cổng trong dải cần quét. Nếu một cổng nào đó trả về gói tin TCP (flag SYN + ACK), tức là đang có dịch vụ sử dụng TCP hoạt động trên cổng tương ứng. Ngược lại, nếu không có gói tin trả về, hoặc trả về gói tin TCP (flag RST), tức là cổng đó đang không hoạt động, hoặc đã bị khóa.
- **UDP**: UDP là giao thức không kết nối (connectionless), do đó chỉ cần gửi một gói tin UDP bất kỳ đến mỗi cổng trong dải cần quét. Nếu không có gói tin trả về hoặc nhận được một gói tin ICMP (Type 3, code 3: Destionation host unreachable), tức đang không có dịch vụ UDP nào đang hoạt động, hoặc cổng tương ứng đã bị khóa.

Để tấn công tránh bị phát hiện bởi các hệ thống phòng thủ, đôi khi kẻ tấn công sử dụng nhiều máy tính khác nhau, mỗi máy tính chỉ quét một tập nhỏ cổng trong dải cần thăm dò, thậm chí sẽ giới hạn số kết nối trong một khoảng thời gian để tránh bị phát hiện bởi các hệ thống phát hiện xâm nhập theo dấu hiệu.

Ngoài ra, kẻ tấn công có thể sử dụng một số kỹ thuật tấn công nâng cao như:

- Decoy Scanning: Kẻ tấn công thực hiện giả mạo địa chỉ nguồn của gói tin scan, sử dụng nhiều địa chỉ nguồn khác nhau (trong đó chỉ có một địa chỉ nguồn là của kẻ tấn công). Do đó khi phát hiện tấn công, rất khó để truy vết ra kẻ tấn công thực sự để có biện pháp xử lý
- Idle Scan: Đây là kỹ thuật giúp kẻ tấn công có thể ẩn danh hoàn toàn. Bằng việc quét cổng thông qua một máy zombie khác. Các bước thực hiện như sau:
    - Kẻ tấn công gửi một gói tin TCP (flag SYN/ACK) đến một cổng của Zombie. Vì Zombie hoàn toàn đang không chờ gói tin SYN/ACK nào cả, do đó nó sẽ gửi trả một gói RST với IPID = x.
    - Kẻ tấn công gửi một gói SYN đến cổng mục tiêu cần quét, với địa chỉ IP là địa chỉ giả mạo của Zombie. Khi đó nếu cổng cần quét đang mở (có dịch vụ đang chạy), mục tiêu sẽ trả lời bằng gói SYN/ACK, lúc này Zombie sẽ trả lời bằng một gói RST với IPID = x+1
    - Tiếp theo kẻ tấn công tiếp tục gửi một gói TCP SYN/ACK đến Zombie để kiểm tra, nếu giá trị IPID của gói RST trả về là x+2 (tức là cổng mục tiêu đang quét là mở), nếu trả về là x+1 tức cổng đó đã đóng.

    Cách làm này tuy không mang lại tốc độ cao, tuy nhiên đảm bảo bảo mật được tối đa địa chỉ của kẻ tấn công.



Hình 2.4 Idle Scan

2.2.3 Các công cụ tấn công

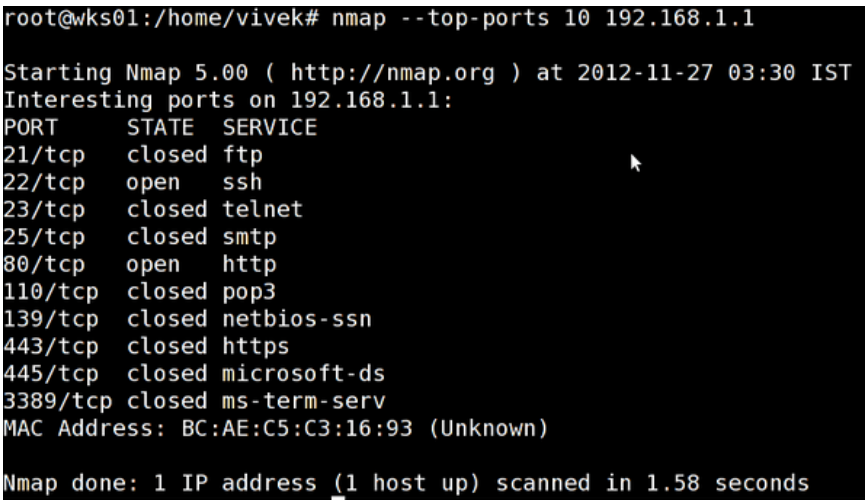
Với cơ chế tấn công rất đơn giản, nên cũng có rất nhiều công cụ tấn công có thể sử dụng, từ đơn giản đến nâng cao.

2.2.3.1. Nmap

Nmap (Network mapper) [5] là một tiện ích mã nguồn mở và miễn phí dùng để khai thác thông tin mạng và kiểm tra bảo mật. Nhiều quản trị viên hệ thống và quản trị viên network đã chứng minh sự hữu dụng của nmap trong các tác vụ như kiểm tra mạng, quản lý dịch vụ và theo dõi thời gian hoạt động của máy chủ và dịch vụ.

Ưu điểm của nmap:

- **Đa tính năng**: Hỗ trợ nhiều kỹ thuật phân tích, quét thông tin mạng, cùng nhiều cơ chế để vượt tường lửa, IDS…
- **Hiệu năng tốt**: Nmap có thể được sử dụng để quét (scan) mạng lớn với hàng trăm nghìn máy.
- **Hỗ trợ trên nhiều nền tảng**: Hỗ trợ hầu hết các hệ điều hành bao gồm Linux, Microsoft Windows, FreeBSD, OpenBSD, Solaris, IRIX, Mac OS X...
- **Dễ dàng sử dụng:** Các thao tác trên môi trường dòng lệnh và giao diện đồ họa dễ hiểu, không phức tạp.
- **Miễn phí**: Mục tiêu chính của dự án **Nmap** là giúp internet trở nên an toàn hơn và cung cấp cho quản trị viên (hoặc hacker) một công cụ để khai thác mạng. Nmap hoàn toàn miễn phí đi kèm mã nguồn đầy đủ.
- **Tài liệu đầy đủ**: Tài liệu của nmap được viết khá chi tiết, dễ hiểu



Hình 2.5 Giao diện dòng lệnh của nmap

Với tấn công quét cổng, nmap chính là công cụ được sử dụng nhiều nhất, bởi cả hacker và quản trị mạng sử dụng để kiểm thử hệ thống (penetration test).

2.2.3.2. Netcat

Netcat là một công cụ đa chức năng, được cài đặt mặc định trên rất nhiều hệ điều hành Linux hoặc Unix, unix-like.

Netcat có thể hoạt động trên hai chế độ: Client và Server, tức có thể vừa lắng nghe kết nối TCP/UDP, chạy ứng dụng đơn giản để kiểm tra hoạt động của hệ thống mạng.

```
root@kali:~/Desktop# nc -nv -w 1 -z 192.168.56.102 10-81
(UNKNOWN) [192.168.56.102] 80 (http) open
(UNKNOWN) [192.168.56.102] 22 (ssh) open
root@kali:~/Desktop# nc -nvu -w 1 -z 192.168.56.102 110-120
(UNKNOWN) [192.168.56.102] 111 (sunrpc) open
root@kali:~/Desktop#
```

Hình 2.6 Quét cổng cùng netcat

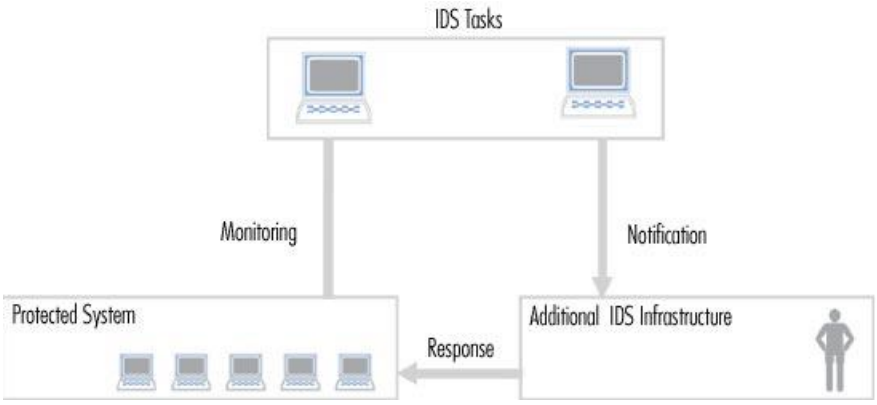
Ở chế độ Client, có một option của netcat là (-w) để phục vụ riêng cho mục đích quét cổng. Điểm mạnh của netcat là hoạt động nhanh, kết quả trực quan, tốc độ nhanh hơn nmap.

2.3 Hệ phống phát hiện xâm nhập (IDS)

2.3.1 Khái niệm

Phát hiện xâm nhập bất thường là quá trình theo dõi các thông số trên máy tính hoặc mạng, phân tích các thông số này để phát hiện dấu hiệu của xâm nhập trái phép. Các dấu hiệu xâm nhập này được gọi là dấu hiệu bất thường.

Một hệ thống phát hiện xâm nhập bất thường (Intrusion Detection System – IDS) được triển khai để giám sát, theo dõi các thông số của máy tính hoặc mạng để phát hiện các dấu hiệu bất thường.

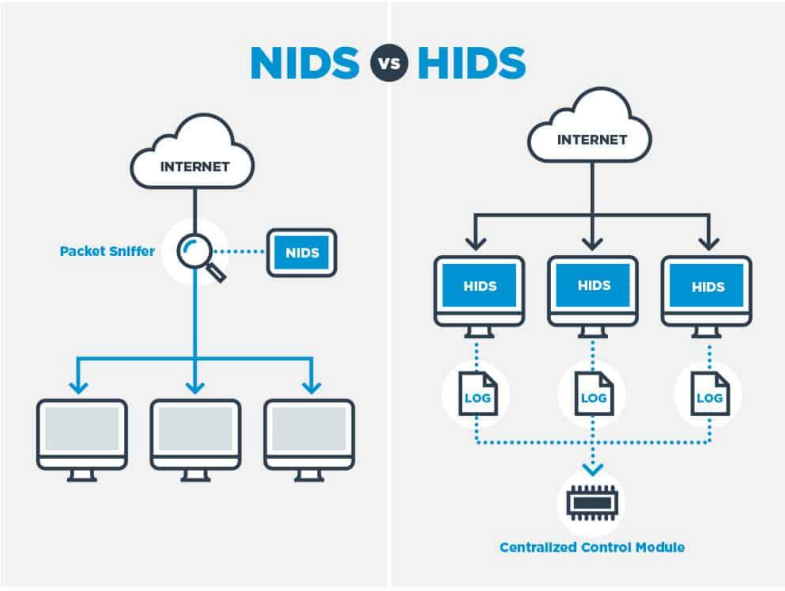


Hình 2.7 Hoạt động của IDS

2.3.2 Phân loại IDS

Các hệ thống phát hiện xâm nhập bất thường được chia làm hai loại, dựa theo vị trí thiết lập: **Network-based IDS** và **Host-based IDS**:

- **Host-based IDS (HIDS)**: Một máy tính, thiết bị trong mạng được cài đặt một số chương trình ngay trên máy. Chương trình trên máy tính chỉ theo dõi các thông số trên thiết bị được cài đặt (như files, lưu lượng mạng, log hệ thống). Host-based IDS, với việc được theo dõi rất nhiều chỉ số một cách chi tiết, sẽ dễ dàng và phát hiện được nhiều loại bất thường.
- **Network-based IDS (NIDS)**: NIDS được cài đặt tại một vị trí nào đó trong hệ thống mạng, theo dõi các thông số nhất định trong mạng, trong đó có lưu lượng mạng, hoặc một số thông số cá thể được gửi từ host trong mạng. Như vậy, NIDS sẽ có một cái nhìn tổng thể hơn so vs HIDS, và không làm ảnh hưởng đến hoạt động tại các thiết bị.



Hình 2.8 IDS trên mạng và IDS trên máy

Dựa vào kiểu theo dõi, IDS cũng có thể được chia làm hai loại: **Signature-based detection** hay **Anomaly-based detection**:

- **Signature-based IDS (IDS dựa trên dấu hiệu)**: là hệ thống IDS phát hiện xâm nhập dựa trên dấu hiệu cụ thể của xâm nhập bất thường. Khi giám sát, IDS tiến hành theo dõi các bộ dữ liệu, so sánh trực tiếp với cơ sở dữ liệu xâm nhập, khi có bộ dữ liệu khớp hoàn toàn với một dữ liệu xâm nhập trên cơ sở dữ liệu đó sẽ được xác định là tấn công và đưa ra cảnh báo. Hệ thống IDS này có ưu điểm là phát hiện chính xác, tỉ lệ phát hiện sai thấp, tuy nhiên sẽ không thể phát hiện ra những bất thường mới, hay biến thể của chúng.
- **Anomaly-based IDS (IDS dựa trên bất thường)**: là hệ thống IDS phát hiện không dựa trên một bộ dữ liệu cụ thể nào mà chỉ là những dấu hiệu bất thường dựa trên sự bất thường trong mạng, hay phát hiện theo sự thay đổi hành vi bất chợt của mạng. Cách làm này có ưu điểm có khả năng phát hiện sự bất thường rộng hơn, nhưng lại cho tỉ lệ sai sót cao hơn so với Signature-based IDS

Hai phần tiếp theo sẽ làm rõ hơn về hai kiểu hệ thống phát hiện này.

2.3.3 Signature-based IDS (SIDS)

SIDS là phương pháp phát hiện dựa trên dấu hiệu cụ thể nào đó của một loại hình tấn công. Trên thực tế, các hệ thống kiểu này được cài đặt với việc theo việc thiết lập luật. Tức với mỗi loại hình tấn công mà hệ thống cần giám sát phát hiện, sẽ có một (số) bộ luật tương ứng với dữ liệu theo dõi, đây đều là những luật được cài đặt cứng, Một bộ luật tốt sẽ cho phép phát hiện với tỷ lệ cao đi cùng với tỷ lệ phát hiện sai thấp. Đồng thời với việc có thể cài đặt luật cụ thể, SIDS có thể đưa vào rất nhiều loai hình tấn công. Đo đó, đây là giải pháp IDS được triển khai trên phần đại đa số các hệ thống phát hiện tấn công thương mại hiện nay. Một trong những IDS phổ biến nhất hiện nay là snort [6], một phần mềm mã nguồn mở, có thể cài đặt và thiết lập luật dễ dàng.

Tuy nhiên, SIDS cũng có một số khuyết điểm. Nó không thể phát hiện những tấn công lạ, hay có thông số thay đổi so với tấn công mạng đã xác định, khi đó để đạt được hiệu quả, cơ sở dữ liệu tấn công cần được cập nhật thường xuyên. Quá nhiều luật có thể làm giảm hiệu năng của hệ thống. Và với bộ luật được thiết lập cứng, nếu thay đổi quá lỏng, có thể khiến cho tỉ lệ phát hiện lỗi tăng cao. Và khi cần cập nhật luật tối ưu, có thể sẽ cần mất một khoảng thời gian cho nhà phát triển, giám sát nghiên cứu, kẻ tấn công có thể tiến hành thực hiện hành vi trái phép trong khoảng thời gian này.

2.3.4 Anomaly-based IDS

Anomaly-based IDS là một cách phát hiện tấn công mới, hứa hẹn sẽ thay thế cho Signature-based IDS trong tương lai. Cách hoạt động của hệ thống này lỏng hơn so với Signature-based IDS. Để có thể đánh giá, triển khai trên dữ liệu mạng được giám sát, hệ thống cần có được những dữ liệu đó, trong điều kiện mạng bình thường. Sau đó, khi theo dõi, hệ thống sẽ so sánh các dữ liệu mới với bộ dữ liệu đã có trước, tìm xem có sự sai khác đáng kể, và đưa ra kết quả, có phải là bất thường hay không.

Một phương pháp xây dựng hệ thống Anomaly-based IDS đang được phát triển rất nhiều hiện nay là áp dụng học máy (machine learning), việc ứng dụng học máy vào trong IDS có thể thay thế cho các hệ thống Signature-based IDS khi làm tăng hiệu năng xử lý dữ liệu (chỉ cần đưa vào mô hình học máy một lần, thay vì so sánh với toàn bộ cơ sở dữ liệu), phát hiện tốt với những mô hình tấn công lạ, dễ dàng cập nhật thay đổi, bổ sung dấu hiệu.

CHƯƠNG 3. PHÁT HIỆN TẤN CÔNG QUÉT CỔNG

3.1 Phát hiện áp dụng thuật toán

Các IDS này được xây dựng theo hướng này đều sử dụng một số mô hình về xác suất, hoặc trên một thuật toán có sẵn, sau đó xây dựng mô hình phát hiện tấn công dựa trên phân tích lưu lượng mạng.

Một số mô hình đã được công cố có thể kể đến như:

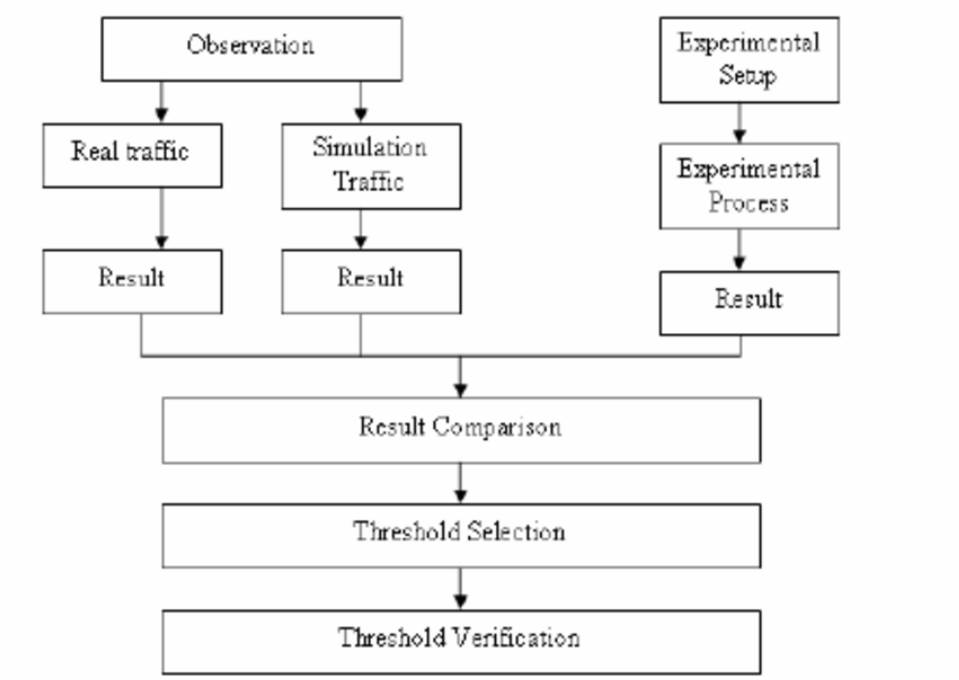
- GrIDS (Graph IDS) [7]: Đây là một phương pháp được sử dụng để phát hiện các tấn công dựa trên việc xây dựng, mô phỏng lại lưu lượng mạng trên bản đồ. Ý tưởng cơ bản của phương pháp này là vẽ lại hệ thống các máy tính (là các node) và và các cạnh thể hiện hoạt động mạng tương ứng giữa các node. Hệ thống này được đề xuất để phát hiện nhiều loại tấn công khác nhau. Tùy vào ý đồ muốn phát hiện loại tấn công nào. Trong việc phân tích phát hiện tấn công quét cổng, có thể cài đặt các cạnh là số kết nối TCP hoặc UDP trên cặp 2 port khác nhau của một host, hay số request TCP-SYN đến một cổng của môt host. Với tấn công quét cổng, thông số trên các cạnh này thường lớn. Ngoài ra nếu kẻ tấn công sử dụng phương pháp tấn công phân tán (tức sử dụng nhiều host để tấn công), thì host bị tấn công sẽ được nhiều host có cạnh là vector hướng đến host đó. Từ biểu đồ thu được, có thể áp dụng một số thuật toán để phát hiện tấn công quét cổng dựa trên hình ảnh của biểu đồ.
- PSD (Port Scan Detection) [8]: Ở hệ thống này, một hệ thống phát hiện tấn công quét cổng trên giao thức TCP (PSD) được đề xuất với việc phân tích phân bố cổng và flag của các gói tin TCP tới một host cần được bảo vệ, PSD sẽ tính toán chỉ số bất thường của mỗi packet và đưa ra dự đoán (xác suất cho packet đó là tấn công quét cổng)
- Robertson et al. [9]: Hệ thống được đề xuất với việc phân tích, xây dựng lại các phiên trao đổi giữa các địa chỉ IP, đánh dấu những địa chỉ IP thực hiện kết nối đến một host nhưng không có trả lời hoặc không có một kết nối sau đó. Tiếp theo một chỉ số bất thường được tính toán dựa trên số kết nối của địa chỉ IP này đến các host trong mạng (mà không có trả lời). Tiếp theo đó áp dụng thuật toán để đánh giá xem địa chỉ IP đó có phải đang tấn công quét cổng hay không.

3.2 Phát hiện sử dụng ngưỡng (Threshold-based IDS)

Đây là một phương pháp phát hiện tấn công cơ bản, có độ hiệu quả tùy thuộc vào thiết lập cài đặt của hệ thống.

Cách hoạt động chính của phương pháp này là phân tích chỉ số mạng, đưa ra một công thức tính toán cho các chỉ số này, sau đó đặt ra một ngưỡng chặn, trong quá trình hoạt động của hệ thống, các chỉ số tương ứng của các gói tin hay kết nối của các host liên tục được tính toán, nếu chỉ số đó vượt ngưỡng sẽ kích hoạt báo động rằng có tấn công mạng.

Đối với tấn công quét cổng, các chỉ số đặt ngưỡng có thể rất đơn giản, ví dụ như số gói tin đến các cổng khác nhau của một host mà không được trả lời, hay số kết nối không hoàn thành của một địa chỉ IP đến host dược bảo vệ, số host thực hiện các kết nối không hoàn thành đến một host cần bảo vệ.



Hình 3.1 Xây dựng IDS theo hướng sử dụng ngưỡng

Phương pháp này được sử dụng ở nhiều hệ thống IDS hiện tại. Trong những hệ thống đơn giản, với loại tấn công quét cổng, chỉ có một số chỉ số như trên được theo dõi và đặt threshold. Với các hệ thống phức tạp hơn, tùy vào thuật toán, nghiên cứu đến hoạt động thực của máy trong mạng, hệ thống sẽ tính chỉ số dựa trên nhiều thông số khác nhau để ra một điểm số. Để xác định được threshold hợp lý, sẽ cần phải thiết lập mộ mô hình mô phỏng, bao gồm cả lưu lượng thông thường, và lưu lượng tấn công để phân tích, tính toán (như mô tả trong Hình 3.1 Xây dựng IDS theo hướng sử dụng ngưỡng

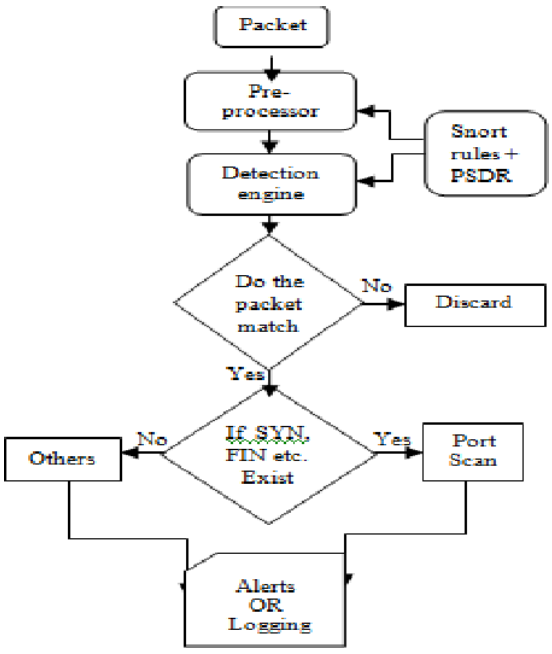
Ví dụ: Trong hệ thống Bro [10] được đề xuất, chỉ có chỉ số duy nhất được đề xuất là số kết nối của một địa chỉ nguồn đến các port khác nhau của một/nhiều host. Sau đó thiết lập ngưỡng cho chỉ số đó.

Ưu điểm của phương pháp này dễ dàng cài đặt, không tốn tài nguyên, hiệu năng tốt, do chỉ cần xử lý một vào chỉ số, không cần thuật toán phức tạp. Tuy nhiên vấn đề chọn ngưỡng hợp lý không phải dễ dàng cho cách thức này. Việc chọn chỉ số quá cao có thể dẫn đến bỏ qua tấn công, ngưỡng quá thấp thì có thể gây ra tỉ lệ báo lỗi cao. Hơn nữa chỉ số ngưỡng này phụ thuộc vào hoạt động thực tế của mạng, ngưỡng tối ưu có thể thay đổi theo hoạt động mạng, có thể do yếu tố thời gian hoặc thay đổi số lượng người sử dụng.

3.3 Phát hiện sử dụng luật (Rule-based IDS)

Phương pháp này thực hiện phân tích các lưu lượng mạng đi qua IDS, so sánh với các luật trong cơ sở dữ liệu luật và đưa ra kết quả. Phương thức này có thể bị hiểu nhầm với phương pháp sử dụng ngưỡng vì cách hoạt động tương đối giống nhau. Rule-based IDS cũng chọn ra một số chỉ số nhất định trong lưu lượng mạng, bao gồm: giao thức, địa chỉ nguồn, đích, số hiệu cổng nguồn, số hiệu cổng đích. Nếu được thiết lập trên host (Host-based IDS), có thể theo dõi cả các thông số liên quan đến content của dữ liệu tầng Ứng dụng.

Với loại tấn công quét cổng, chỉ số nhận biết nằm ở số kết nối đến một địa chỉ IP, ta có thể đặt thêm một số thông số như count (đếm số kết nối) và khi đạt ngưỡng sẽ phát ra cảnh báo (giống như Threshold-based IDS).



Hình 3.2 Một mô hình phát hiện tấn công trên một Rule-based IDS với snort

Trong một hệ thống IDS, thông thường sẽ hướng tới nhiều loại hình tấn công khác nhau, do đó hệ thống cơ sở dữ liệu cho IDS có thể sẽ rất lớn, để có thể bao quát được nhiều loại tấn công. Với việc mỗi packet đi qua đều cần được phân tích, việc so sánh luật này có thể làm giảm hiệu năng của IDS, khiến cho yếu tố phát hiện thời gian thực không đảm bảo, đặc biệt là trong trường hợp lưu lượng mạng lớn.

Đồng thời, Rule-based IDS cũng cần phải thiết lập phù hợp với hoạt động mạng thực tế của hệ thống.

3.4 Phát hiện sử dụng học máy (MachineLearning-based IDS)

Học máy là một công cụ đang ngày càng sử dụng nhiều trên nhiều lĩnh vực, bài toán của công nghệ thông tin. Học máy cũng đã được áp dụng vào mô hình hệ thống phát hiện xâm nhập bất thường.

Nguyên lý của hệ thống này là nghiên cứu một mô hình đặc trưng của một dữ liệu thu được từ hệ thống mạng. Sau đó dựa trên dữ liệu dán nhãn có sẵn, hoặc thực hiện các mô hình mô phỏng thu được, sẽ tiến hành đưa vào một số mô hình học máy để học. Mô hình đưa ra các chỉ số tốt nhất sẽ được chọn.

Sau đó mô hình này được đưa vào trong hệ thống phát hiện tấn công, các đặc trưng được tính toán trên thời gian thực, xử lý và đưa qua mô hình đã học để dự đoán, đưa ra kết quả.

CHƯƠNG 4. MỘT SỐ CHƯƠNG TRÌNH THỬ NGHIỆM

4.1 Snort

Trong phần này,chúng em sử dụng công cụ Snort để phát hiện scanning trên máy, với mô hình của một Host-based IDS và Rule-based IDS.

4.1.1 Mô hình cài đặt

Mô hình mạng gồm 2 máy tính:

- Một máy ảo (có địa chỉ IP: 192.168.56.1)
- Máy thật (có địa chỉ IP: 192.168.137.57)

4.1.2 Kết quả

Khi thực hiện tấn công quét cổng máy thật từ máy ảo:

```
[root@localhost ~]# nmap 192.168.137.57

Starting Nmap 5.51 ( http://nmap.org ) at 2020-06-08 21:24 ICT
Nmap scan report for LAPTOP-UP01BTFJ.mshome.net (192.168.137.57)
Host is up (1.0s latency).
Not shown: 994 closed ports
PORT     STATE    SERVICE
25/tcp   filtered smtp
110/tcp  filtered pop3
135/tcp  open     msrpc
139/tcp  open     netbios-ssn
445/tcp  open     microsoft-ds
5357/tcp open     wsdapi

Nmap done: 1 IP address (1 host up) scanned in 18.38 seconds
```

Kết quả thu được trên snort đang được chạy trên máy thật:

```
WARNING: No preprocessors configured for policy 0.
06/08-21:32:26.167092 192.168.56.1:5353 -> 224.0.0.251:5353
UDP TTL:255 TOS:0x0 ID:4905 IpLen:20 DgmLen:125
Len: 97
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

WARNING: No preprocessors configured for policy 0.
06/08-21:32:26.167732 fe80:0000:0000:0000:0000:95a5:3ba2:ac99:31c1:5353 -> ff02:0000:0000:0000:0000:0000:0000:00fb:5353
UDP TTL:1 TOS:0x0 ID:0 IpLen:40 DgmLen:87
Len: 39
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

WARNING: No preprocessors configured for policy 0.
06/08-21:32:26.168548 192.168.56.1:5353 -> 224.0.0.251:5353
UDP TTL:1 TOS:0x0 ID:4906 IpLen:20 DgmLen:105
Len: 77
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

WARNING: No preprocessors configured for policy 0.
06/08-21:32:26.168682 fe80:0000:0000:0000:0000:95a5:3ba2:ac99:31c1:5353 -> ff02:0000:0000:0000:0000:0000:0000:00fb:5353
UDP TTL:1 TOS:0x0 ID:0 IpLen:40 DgmLen:125
Len: 77
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

WARNING: No preprocessors configured for policy 0.
06/08-21:32:26.169200 192.168.56.1:5353 -> 224.0.0.251:5353
UDP TTL:1 TOS:0x0 ID:4907 IpLen:20 DgmLen:67
Len: 39
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

WARNING: No preprocessors configured for policy 0.
06/08-21:32:26.169670 fe80:0000:0000:0000:0000:95a5:3ba2:ac99:31c1:5353 -> ff02:0000:0000:0000:0000:0000:0000:00fb:5353
UDP TTL:1 TOS:0x0 ID:0 IpLen:40 DgmLen:87
Len: 39
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

WARNING: No preprocessors configured for policy 0.
06/08-21:32:26.170269 fe80:0000:0000:0000:95a5:3ba2:ac99:31c1:5353 -> ff02:0000:0000:0000:0000:0000:0000:00fb:5353
UDP TTL:1 TOS:0x0 ID:0 IpLen:40 DgmLen:125
Len: 77
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

WARNING: No preprocessors configured for policy 0.
06/08-21:32:26.170925 192.168.56.1:5353 -> 224.0.0.251:5353
UDP TTL:1 TOS:0x0 ID:4908 IpLen:20 DgmLen:105
Len: 77
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

4.2 Hệ thống phát hiện xâm nhập sử dụng học máy trên mạng SDN

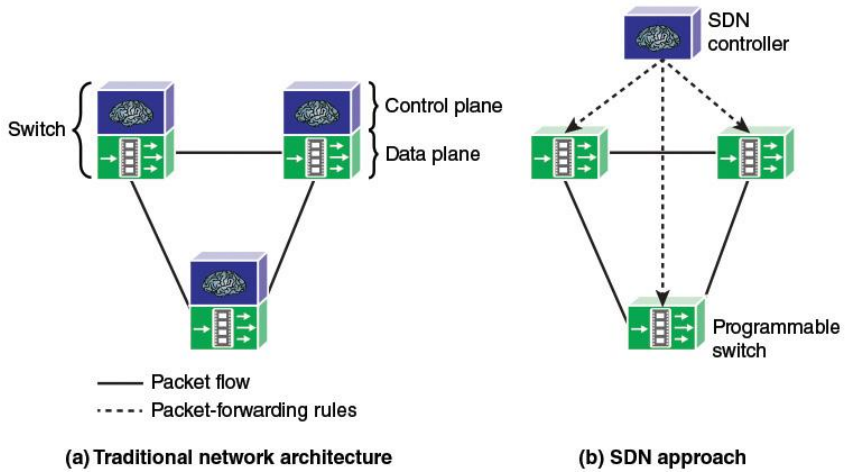
Trong phần này, em xin được tóm tắt quy trình thực hiện một mô hình học máy trên mạng SDN, là một hệ thống Network IDS và Anomaly-based IDS

Các công cụ sử dụng:

- **Nmap**: Công cụ tấn công portscan
- **Mininet**: Một chương trình mô phỏng mạng SDN
- **POX**: Chương trình controller cho mạng SDN

4.2.1 Mạng SDN

Trong mô hình mạng truyền thống, các thiết bị mạng (router, switch, firewall, IDS…) đều gồm hai thành phần: phần cứng (dataplane) và phần mềm (control plane). Có rất nhiều nhà sản xuất cho các thiết bị này, mỗi nhà sản xuất sử dụng các firmware, phần mềm riêng tối ưu cho sản phẩm của riêng họ. Với việc các thành phần điều khiển (control plane) được phân tán tại từng thiết bị. Điều này sẽ gây nên một sự khó khăn trong việc phát hiện lỗi, mở rộng, thay đổi chính sách khi sẽ phải thiết lập thủ công trên từng thiết bị, đặc biệt là trong điều kiện hệ thống mạng ngày một phát triển, phức tạp hơn.



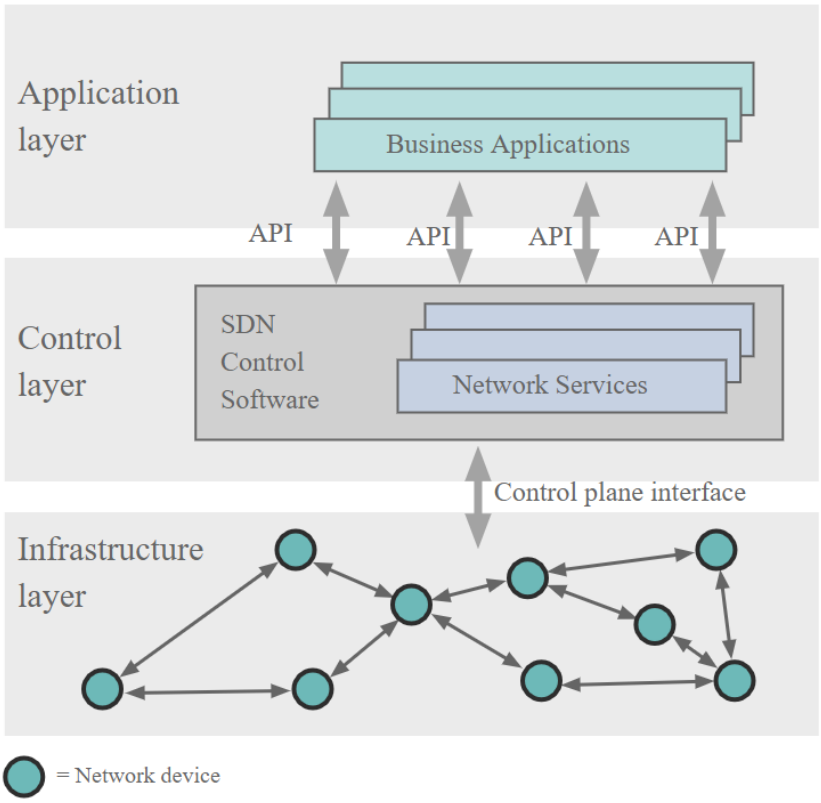
Hình 4.1 Mạng truyền thống và mạng SDN

Để giải quyết vấn đề đó, mô hình mạng SDN đã ra đời, với việc tập trung tất cả các bộ phận điều khiển vào một phân vùng duy nhất (Hình 4.1 Mạng truyền thống và mạng SDN

Nguyên tắc chính của mạng SDN là tách biệt hoàn toàn lớp chuyển tiếp dữ liệu (data plane) và lớp điều khiển, đồng thời tập trung lớp điều khiển lại một chỗ (thuận tiện cho việc thực hiện chính sách, giám sát mạng, phát hiện và sửa lỗi…)

Mô hình của một mạng SDN gồm có ba thành phần chính:

- **Lớp hạ tầng (Infrastructure Layer)**: Gồm các thiết bị mạng (chỉ còn phần cứng). Thực hiện nhiệm vụ chính là chuyển tiếp dữ liệu, dựa theo các luật được cài đặt từ controller.
- **Lớp điều khiển (Control Layer):** Bộ não chính của của kiến trúc mạng SDN được tập trung tại bộ điều khiển (Controller) của SDN. Lớp điều khiển là phần cốt lõi trong mạng, nằm ở giữa và làm nhiệm vụ kết nối lớp hạ tầng và lớp ứng dụng. Lớp điều khiển trực tiếp làm nhiệm vụ đưa ra cài đặt luật chuyển tiếp (flow rules) cho các switch.
- **Lớp ứng dụng (Application Layer):** Bao gồm nhiều ứng dụng và các dịch vụ, được chạy phía trên lớp điều khiển, giao tiếp với lớp điều khiển qua Northbound API để theo dõi và quản lý hệ thống mạng. Người dùng có thể dễ dàng phát triển các ứng dụng trên nền tảng API được cung cấp bởi controller



Hình 4.2 Kiến trúc của mạng SDN

Hai lớp liền nhau giao tiếp với nhau bằng các API. Như vậy, với việc tập trung điều khiển tại một vị trí, SDN cho phép người sử dụng có thể lập trình các ứng dụng để điều khiển controller, qua đó trực tiếp giám sát và điều khiển hoạt động của mạng.

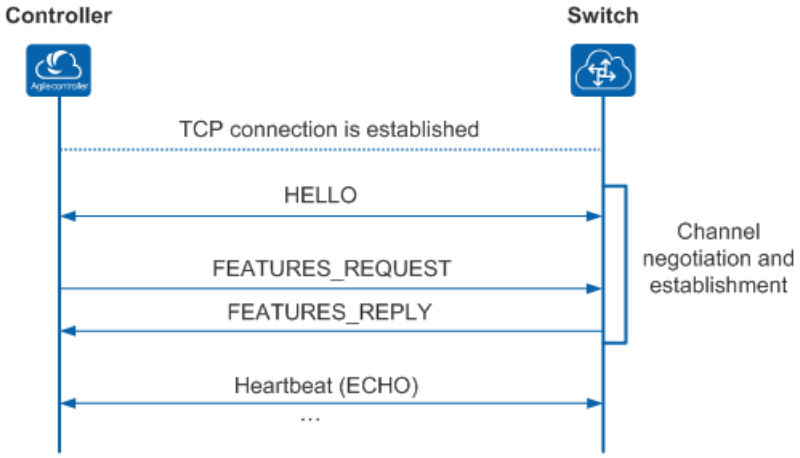
4.2.2 IDS trên mạng SDN

Như đã nói đến ở trên, ta có thể lập trình các ứng dụng khác nhau trên lớp ứng dụng. Do đó, phục vụ cho thử nghiệm nội dung của đề tài này, chúng em xin đề

xuất một mô hình phát hiện xâm nhập là một chương trình có tên SDN-IDS nằm trên lớp ứng dụng của mô hình IDS. Nếu trong mô hình mạng chỉ có 1 switch kết nối nhiều host, ta có thể coi như đây là mạng truyền thống với một switch và một thiết bị IDS thu thập thông tin từ switch đó.

4.2.3 Xây dựng SDN-IDS

Trong mạng SDN, các switch thường giao tiếp với Controller ở tầng điều khiển bằng giao thức OpenFlow. Giao thức OpenFlow hoạt động trên nền của giao thức TCP/TLS, với cổng mặc định của controller là 6633 hoặc 6653. Trước khi giao tiếp, Switch và Controller trước tiên phải thực hiện bắt tay 3 bước, trao đổi khóa. Như vậy là kênh truyền giữa Controller và Switch được đảm bảo về vấn đề bảo mật. Sau đó, cả hai bên đều gửi các gói tin OFPT_HELLO với mục đích để thiết lập phiên trao đổi OpenFlow, với mục đích trao đổi thông số, thông nhất phiên bản OpenFlow được sử dụng và gán ID. Từ lúc này, giữa Controller và Switch đã được thiết lập một kênh truyền an toàn, sử dụng cấu trúc thông điệp của giao thức OpenFLow để giao tiếp.



Hình 4.3 Thiết lập kênh giao tiếp trong giao thức OpenFlow

OpenFlow cung cấp những giao diện thông điệp, cho phép Controller và Switch trao đổi với nhau bao gồm thay đổi flow, thông số của flow …

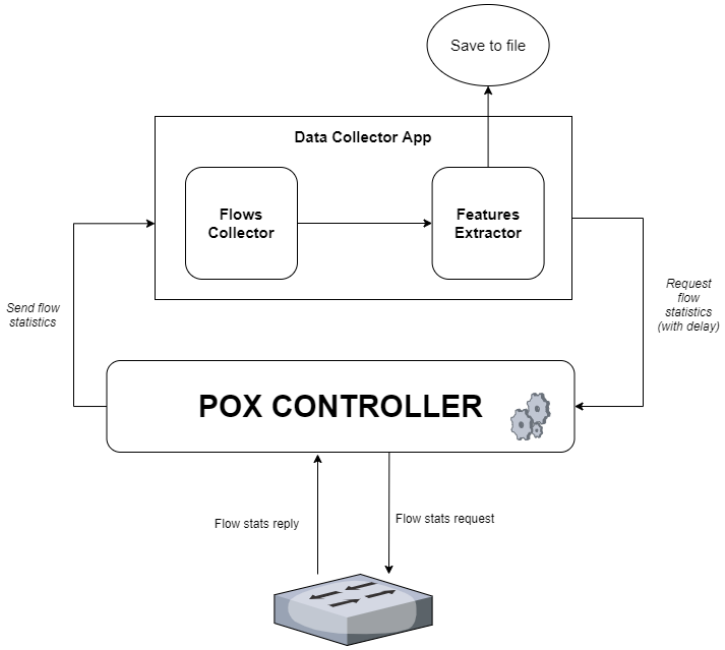
Như đã đề cập đến ở trên, hệ thống phát hiện xâm nhập ở đây là một ứng dụng của mô hình mạng SDN. SDN-IDS là một chương trình theo dõi mạng, có cấu trúc định kỳ lấy thông số flow từ switch (thông qua controller), đưa các đặc trưng phân tích từ flow vào mô hình học máy đã được thực hiện học và đưa ra đánh giá.

Như vậy, việc xây dựng SDN-IDS cần có hai thành phần chính:

- Xây dựng mô hình thu thập dữ liệu và đưa vào mô hình học
- Thu thập mô hình và đưa ra vào triển khai dự đoán trong SDN-IDS

4.2.4 Thu thập dữ liệu và học máy

Mô hình thu thập dữ liệu được mô tả như Hình 4.4 Mô hình thu thập dữ liệu cho học máy



Hình 4.4 Mô hình thu thập dữ liệu cho học máy

Mô hình mạng sẽ được mô phỏng gồm các máy host thông thường. Dữ liệu flow mạng được dán nhãn ‘normal’ sẽ được thu riêng biệt bằng việc các host trong mạng sử dụng các dịch vụ mạng thông thường. Dữ liệu mạng được dán nhãn tấn công ‘portscan’ sẽ được thu thập bằng việc tấn công thực tế giữa hai máy trong mạng với công cụ nmap. Cuối cùng dữ liệu thu thập được sẽ được đưa vào một số mô hình học máy và so sánh, chọn ra một mô hình có kết quả tốt nhất.

Các mô hình học máy được sử dụng và kết quả:

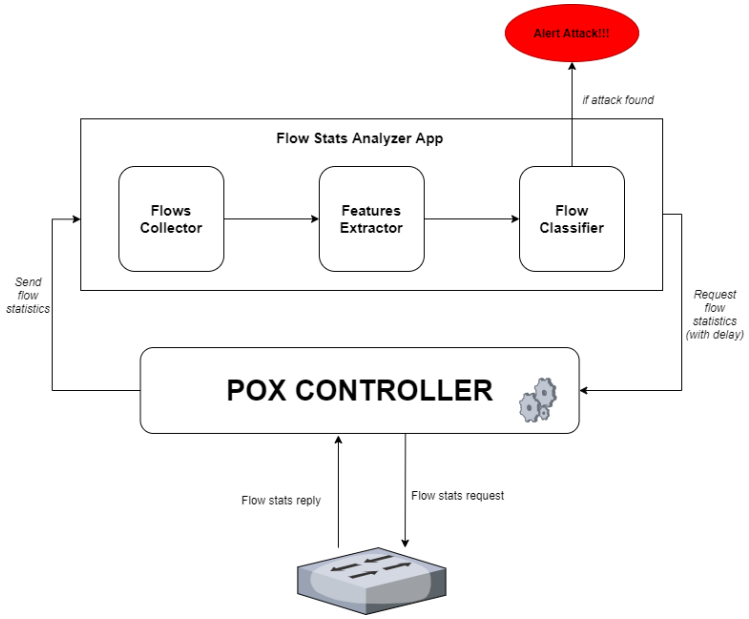
Model		**Validating loss**	**Validating accuracy**
CNN Conv1D	1 loop	0.0107	99.76%
	2 loops	0.0088	99.90%
CNN Conv2D		0.0281	99.52%
Naïve Bayes	GaussianNB		91.16%
	CategoricalNB		98.05%

KNN	k = 3		99.43%

Bảng 4.1 Kết quả của một số mô hình học máy

4.2.5 Áp dụng vào SDN-IDS

Sau khi đã thực hiện học và chọn được mô hình có kết quả tốt nhất. Em sẽ đưa mô hình này vào việc dự đoán và phát hiện tấn công với mô hình hoạt động như Hình 4.5 Mô hình hoạt động của SDN-IDS:



Hình 4.5 Mô hình hoạt động của SDN-IDS

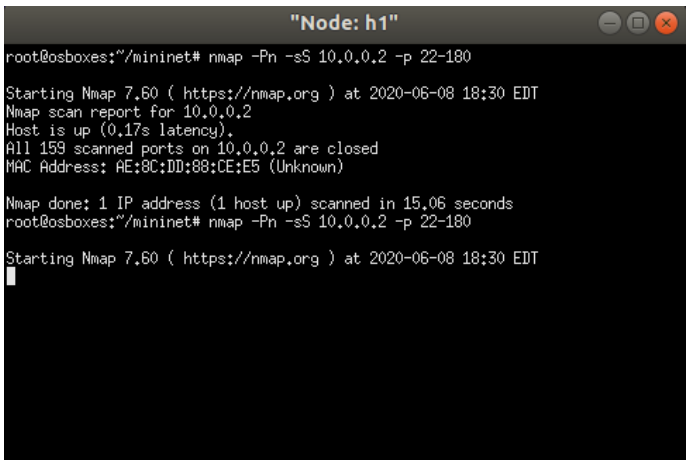
Chương trình sẽ thực hiện tương tự như lúc thu thập dữ liệu cho học máy, tuy nhiên lúc này các dữ liệu là không được dán nhãn, sẽ được đưa qua một bộ phận là Flow-Classifier, bộ phận này sẽ tiến hành dùng mô hình đã học ở trên, dự đoán phân loại cho flow và đưa ra cảnh báo nếu đó là lỗi.

4.2.6 Kết quả

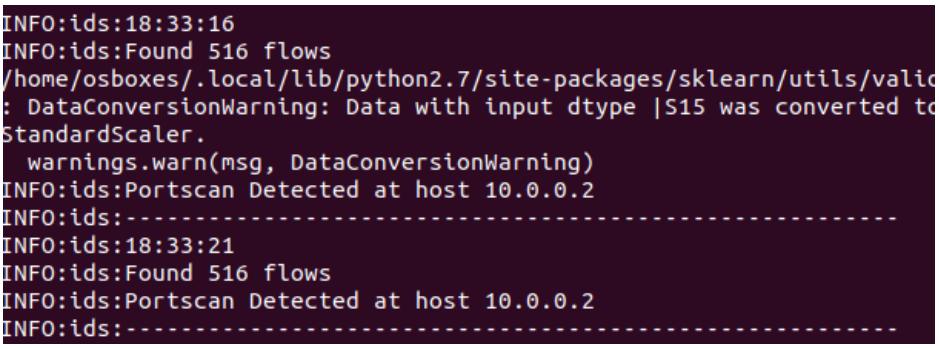
Thực hiện tấn công thử trên một máy host trong mạng đến máy khác bằng nmap:

Nmap -Pn -sS 10.0.0.2 -p 22-180

Thực hiện quét các cổng 22 – 180 với giao thức TCP trên máy có địa chỉ 10.0.0.2.



Kết quả thu được tại chương trình của SDN-IDS:



CHƯƠNG 5. KẾT LUẬN

Trong quá trình thực hiện đề tài, chúng em đã thực hiện được:

- Tìm hiểu về mục đích, nguyên lý và đặc tính của tấn công quét cổng
- Tìm hiểu các kỹ thuật và công cụ được sử dụng để tấn công
- Tìm hiểu các mô hình phát hiện đã được đưa ra và triển khai
- Thực hiện thử hai mô hình phát hiện tấn công và thực hiện thành công

Tuy nhiên, do thời gian có hạn, chúng em chưa thể tìm hiểu hết được các phương thức tấn công quét cổng đã có. Chúng em sẽ cố gắng để tìm hiểu thêm và triển khai thử nghiệm nhiều hệ thống hơn để hiểu rõ hơn về hoạt động của các hệ thống IDS nói chung và module phát hiện tấn công quét cổng nói riêng.

TÀI LIỆU THAM KHẢO

[1] James F. Kurose, Keith W. Ross, Computer Networking: A Top-Down Approach, 2008.

[2] Pranab Bandhu Nath, Md.Mofiz Uddin, "TCP-IP Model in Data Communication and Networking," *American Journal of Engineering Research (AJER),* vol. 4, pp. 102-107, 2015.

[3] "SQL map," [Online]. Available: http://sqlmap.org/.

[4] Bhuyan, Monowar and Bhattacharyya, Dhruba K and Kalita, Jugal, "Surveying Port Scans and Their Detection Methodologies," *The Computer Journal*, vol. 54, pp. 1565-1581, 2011.

[5] "Nmap Reference Guide," [Online]. Available: https://nmap.org/book/man.html.

[6] Marty Roesch, Joel Esler, Christine Council, Sammi Seaman el. at., "Snort," [Online]. Available: snort.org.

[7] S. Staniford-Chen, S. Cheung, R. Crawford, M. Dilger, J. Frank, J. Hoagland, K. Levitt, C. Wee, R. Yip, D. Zerkle, "GrIDS{A GRAPH BASED INTRUSION DETECTION SYSTEM FOR LARGE NETWORKS," Department of Computer Science, University of California.

[8] Shan, R., Li, X.D. & Li, J. , "An adaptive algorithm to detect port scans," J. of Shanghai Univ, 2004.

[9] Robertson, S., Siegel, E. V., Miller, M., and Stolfo,, "Surveillance detection in high bandwidth," *IEEE Computer Society*, p. 130–139, 2003.

[10] Paxson, V. , "A system for detecting network intruders in real-time," *Proceedings of USENIX Security Symposium'98,* p. 2435–2463, 1998.

[11] Bernardo Damele Assumpcao Guimaraes and Miroslav Stampar, "sqlmap," [Online]. Available: http://sqlmap.org/.