

ĐẠI HỌC THÁI NGUYÊN
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

BÙI THỊ HƯƠNG THƠM

**NGHIÊN CỨU XÂY DỰNG CÔNG CỤ HỖ TRỢ
PHÂN TÍCH GÓI TIN TRONG ĐIỀU TRA MẠNG**

LUẬN VĂN THẠC SĨ KHOA HỌC MÁY TÍNH

Thái Nguyên, năm 2015

**ĐẠI HỌC THÁI NGUYÊN
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**

BÙI THỊ HƯƠNG THƠM

**NGHIÊN CỨU XÂY DỰNG CÔNG CỤ HỖ TRỢ
PHÂN TÍCH GÓI TIN TRONG ĐIỀU TRA MẠNG**

**Chuyên ngành : Khoa học máy tính
Mã số chuyên ngành: 60 48 01 01**

LUẬN VĂN THẠC SĨ KHOA HỌC MÁY TÍNH

**NGƯỜI HƯỚNG DẪN KHOA HỌC
TS. TRẦN ĐỨC SỰ**

Thái Nguyên, tháng 8 năm 2015

LỜI CAM ĐOAN

Tôi là: **Bùi Thị Hương Thơm**

Lớp: CK12I

Khoá học: 2014 - 2015

Chuyên ngành: Khoa học máy tính

Mã số chuyên ngành: 60 48 0101

Cơ sở đào tạo: Trường Đại học Công nghệ thông tin và Truyền thông Thái Nguyên.

Giáo viên hướng dẫn: **TS. Trần Đức Sự**

Tôi xin cam đoan luận văn “**Nghiên cứu xây dựng công cụ hỗ trợ phân tích gói tin trong điều tra mạng**” này là công trình nghiên cứu của riêng tôi. Các số liệu sử dụng trong luận văn là trung thực. Các kết quả nghiên cứu được trình bày trong luận văn chưa từng được công bố tại bất kỳ công trình nào khác.

Thái Nguyên, ngày 15 tháng 07 năm 2015

HỌC VIÊN

Bùi Thị Hương Thơm

LỜI CẢM ƠN

Để hoàn thành chương trình cao học và viết luận văn này, tôi đã nhận được sự hướng dẫn, giúp đỡ và chỉ bảo nhiệt tình của quý thầy cô trường Đại học Công nghệ thông tin và Truyền thông. Đặc biệt là những thầy cô ở Viện công nghệ thông tin Hà Nội đã tận tình dạy bảo cho tôi trong suốt thời gian học tập tại trường.

Tôi xin gửi lời cảm ơn sâu sắc đến TS. Trần Đức Sự đã dành nhiều thời gian và tâm huyết hướng dẫn tôi hoàn thành luận văn này.

Mặc dù tôi đã cố gắng hoàn thiện luận văn bằng tất cả năng lực của mình, song không thể tránh khỏi những thiếu sót, rất mong nhận được sự đóng góp quý báu của quý thầy cô và các bạn.

Tôi xin chân thành cảm ơn!

MỤC LỤC

CHƯƠNG 1: TỔNG QUAN VỀ KỸ THUẬT ĐIỀU TRA SỐ VÀ ĐIỀU TRA MẠNG.....	3
1.1. GIỚI THIỆU VỀ ĐIỀU TRA SỐ	3
1.1.1. Lịch sử điều tra số.....	3
1.1.2. Ứng dụng của điều tra số.....	5
1.1.3. Quy trình thực hiện điều tra số.....	6
1.1.4. Các loại hình điều tra số phổ biến.....	7
1.2. GIỚI THIỆU VỀ PHÂN TÍCH ĐIỀU TRA MẠNG (NETWORK FORENSICS).....	13
1.2.1. Vai trò và ứng dụng của phân tích điều tra mạng.....	15
1.2.2. Nền tảng kỹ thuật cho phân tích điều tra mạng.....	16
1.2.3. Các kỹ thuật tấn công mạng máy tính.....	28
CHƯƠNG 2. PHÂN TÍCH ĐIỀU TRA MẠNG VÀ PHÂN TÍCH GÓI TIN TRONG ĐIỀU TRA MẠNG.....	33
2.1. QUY TRÌNH TỔNG QUAN TRONG PHÂN TÍCH ĐIỀU TRA MẠNG	33
2.1.1. Giai đoạn 1: Chuẩn bị và ủy quyền	33
2.1.2. Giai đoạn 2: Phát hiện sự cố hoặc hành vi phạm tội	34
2.1.3. Giai đoạn 3: Ứng phó sự cố.....	34
2.1.4. Giai đoạn 4: Thu thập các vết tích mạng.....	35
2.1.5. Giai đoạn 5: Duy trì và bảo vệ	35
2.1.6. Giai đoạn 6: Kiểm tra	35
2.1.7. Giai đoạn 7: Phân tích.....	36
2.1.8. Giai đoạn 8: Điều tra và quy kết trách nhiệm	36
2.1.9. Giai đoạn 9: Tổng kết đánh giá	37
2.2. KỸ THUẬT PHÂN TÍCH ĐIỀU TRA MẠNG	37
2.2.1. Phân tích gói tin.....	37
2.2.2. Phân tích thống kê lưu lượng.....	38
Số hóa bởi Trung tâm Học liệu - ĐHTN	
http://www.lrc-tnu.edu.vn/	

2.2.3. Phân tích nhật ký, sự kiện	39
2.3. CÔNG CỤ SỬ DỤNG TRONG PHÂN TÍCH ĐIỀU TRA MẠNG	40
2.3.1. Wireshark	40
2.3.2. NetworkMiner	40
2.3.3. Snort	41
2.3.4. Tcpxtract & TCPflow	42
2.3.5. Foremost	42
2.3.6. Scapy	43
2.4. CÁCH THỨC PHÂN TÍCH GÓI TIN TRONG ĐIỀU TRA MẠNG	43
2.4.1. Đặc điểm gói tin mạng	43
2.4.2. Cách thức phân tích gói tin mạng	53
CHƯƠNG 3: XÂY DỰNG CÔNG CỤ HỖ TRỢ PHÂN TÍCH GÓI TIN	62
3.1. MỤC TIÊU CÔNG CỤ HỖ TRỢ PHÂN TÍCH GÓI TIN	63
3.2. PHÂN TÍCH, THIẾT KẾ CÔNG CỤ HỖ TRỢ PHÂN TÍCH GÓI TIN THEO GIAO THỨC MẠNG 63	
KẾT LUẬN	71
TÀI LIỆU THAM KHẢO	72
PHỤ LỤC	73

DANH MỤC CÁC TỪ VIẾT TẮT

STT	Tên viết tắt	Tên tiếng Anh
1	ARP	Address resolution protocol
2	CPU	Central Processing Unit
3	DHCP	Dynamic Host Configuration Protocol
4	DNS	Domain Name System
5	DoS	Denial of Service
6	HTTP	Hypertext Transfer Protocol
7	ICMP	Internet control message protocol
8	IDS	Intrusion Detection System
9	IP	Internet Protocol
10	TCP	Tranmission Control Protocol
11	RARP	Reserve address resolution protocol
12	OSI	Open Systems Interconnection Reference Model
13	UDP	User Datagram Protocol
14	URL	Uniform Resource Locator

DANH MỤC CÁC HÌNH VẼ, ĐỒ THỊ

<i>Hình 1.1. Các bước thực hiện điều tra số.....</i>	<i>6</i>
<i>Hình 1.2. Các bước thực hiện điều tra di động.....</i>	<i>10</i>
<i>Hình 1.3. Network Forensics trong Forensics Sciences.....</i>	<i>13</i>
<i>Hình 2.1. Quy trình chung trong phân tích điều tra mạng.....</i>	<i>33</i>
<i>Hình 2.2. Tcp header.....</i>	<i>44</i>
<i>Hình 2.3. UDP header.....</i>	<i>46</i>
<i>Hình 2.4. IP Header.....</i>	<i>47</i>
<i>Hình 2.5. Type of Services.....</i>	<i>47</i>
<i>Hình 2.6. Vị trí gói ICMP header.....</i>	<i>50</i>
<i>Hình 2.7. ICMP header.....</i>	<i>51</i>
<i>Hình 2.8. ARP Header.....</i>	<i>52</i>
<i>Hình 2.9. Nghe trong mạng hub.....</i>	<i>55</i>
<i>Hình 2.10. Xung đột trong mạng hub.....</i>	<i>56</i>
<i>Hình 2.11. Nghe trong mạng Switch.....</i>	<i>56</i>
<i>Hình 2.12. Bắt lưu lượng của thiết bị mục tiêu trên mạng Switch bằng Port Mirroring.....</i>	<i>57</i>
<i>Hình 2.13. Bắt lưu lượng của thiết bị mục tiêu trên mạng Switch bằng Hubbing Out.....</i>	<i>58</i>
<i>Hình 2.14. Bắt lưu lượng của thiết bị mục tiêu trên mạng Switch bằng ARP Cache Poisoning.....</i>	<i>60</i>
<i>Hình 2.15. Nghe trong mạng sử dụng Router.....</i>	<i>61</i>
<i>Hình 3.1. Mô hình hoạt động.....</i>	<i>64</i>
<i>Hình 3.2. Các bước hoạt động của công cụ.....</i>	<i>64</i>
<i>Hình 3.3. Thống kê ban đầu của các gói tin.....</i>	<i>65</i>
<i>Hình 3.4. Thống kê gói tin theo địa chỉ IP của tất cả các giao thức.....</i>	<i>66</i>

<i>Hình 3.5. Thống kê gói tin theo địa chỉ MAC của tất cả các giao thức</i>	<i>67</i>
<i>Hình 3.6. Thống kê gói tin theo địa chỉ IP của giao thức TCP</i>	<i>69</i>
<i>Hình 3.7. Thống kê gói tin theo địa chỉ MAC của giao thức TCP.....</i>	<i>69</i>

MỞ ĐẦU

Sự phát triển mạnh mẽ của Công nghệ thông tin nói chung và mạng Internet nói riêng đã tạo điều kiện thuận lợi cho việc cung cấp đa dạng các dịch vụ hữu ích đến với con người. Trong vài năm gần đây, nó không ngừng phát triển để phù hợp với một cộng đồng rộng lớn hơn nhiều, đem lại rất nhiều dịch vụ với các lợi ích thương mại, kinh tế, xã hội... Tuy nhiên, nó cũng trở thành môi trường cho các cuộc chiến tranh không gian số, nơi mà các cuộc tấn công của nhiều loại hình khác nhau (liên quan tài chính, tư tưởng, hành vi trả đũa...) đang được phát động. Các giao dịch thương mại điện tử được thực hiện trực tuyến là mối quan tâm chính của tội phạm mạng. Những hacker ăn cắp tài khoản của người dùng để thực hiện ý đồ xấu như mua bán trực tuyến, thỏa hiệp với một website hay máy chủ, phát động tấn công lên các hệ thống khác. Chính vì thế, hệ thống máy tính cần phải được bảo vệ khỏi các cuộc tấn công và phản ứng một cách thích hợp để tạo ra những xử lý nhằm giảm thiểu thiệt hại do tội phạm gây ra. Quá trình xử lý sự cố, phục hồi chứng cứ và truy tìm dấu vết tội phạm liên quan đến ngành khoa học điều tra số (digital forensics).

Phân tích điều tra mạng(Network Forensics) là một nhánh của ngành khoa học điều tra số đề cập đến việc chặn bắt, ghi âm và phân tích lưu lượng mạng cho mục đích điều tra và ứng phó sự cố. Có rất nhiều kỹ thuật cũng như công cụ hỗ trợ trong việc chặn bắt các dữ liệu lan truyền trên mạng để một cuộc tấn công hay một ý đồ xấu có thể bị điều tra, ngăn chặn.

Công cụ hỗ trợ phân tích gói tin trong điều tra mạng là một vấn đề rất quan trọng và luôn cấp thiết. Để cho quá trình điều tra mạng được nhanh và chính xác thì một chương trình hỗ trợ cần phải được xây dựng một cách chính xác cung cấp nhiều thông tin cần thiết cho người điều tra.

Nhận thấy được mức độ cấp thiết của vấn đề, học viên đã triển khai nghiên cứu thực hiện luận văn: ***“Nghiên cứu xây dựng công cụ hỗ trợ phân tích gói tin trong điều tra mạng”*** nhằm đưa ra những hiểu biết chung về ngành khoa học điều tra, cùng với chương trình phục vụ quá trình điều tra mong một phần nào đó sẽ giúp cho quá trình điều tra phân tích mạng được hỗ trợ một cách dễ dàng và nhanh chóng hơn.

Luận văn được triển khai thành 3 chương với nội dung như sau:

Chương I – Tổng quan về kỹ thuật điều tra số và điều tra mạng

Chương II – Phân tích điều tra mạng và phân tích gói tin trong điều tra mạng

Chương III – Xây dựng công cụ hỗ trợ phân tích gói tin

Xây dựng công cụ hỗ trợ phân tích gói tin trong điều tra mạng (network forensic) là một đề tài còn khá mới mẻ, mang tính chất thời đại, cần được cập nhật, chỉnh sửa và bổ sung thường xuyên. Với thời gian tìm hiểu và kiến thức còn hạn chế nên đề tài khó tránh khỏi những thiếu sót. Em rất mong được sự góp ý của thầy cô để luận văn thêm hoàn thiện.

Em xin gửi lời cảm ơn chân thành nhất đến thầy giáo hướng dẫn thực hiện đồ án, thầy **Ts. Trần Đức Sự** đã dành nhiều thời gian quan tâm, đôn đốc và giúp đỡ em trong quá trình làm luận văn.

Em xin được bày tỏ lòng tri ân sâu sắc đến tất cả các thầy cô giảng dạy lớp cao học CK12I đã giúp em tích lũy được nhiều kinh nghiệm cùng những kiến thức chuyên môn trong quá trình dài học tập, nghiên cứu.

CHƯƠNG 1: TỔNG QUAN VỀ KỸ THUẬT ĐIỀU TRA SỐ VÀ ĐIỀU TRA MẠNG

1.1. Giới thiệu về điều tra số

Điều tra số (đôi khi còn gọi là Khoa học điều tra số) là một nhánh của ngành Khoa học điều tra đề cập đến việc phục hồi và điều tra các tài liệu tìm thấy trong các thiết bị kỹ thuật số, thường có liên quan đến tội phạm máy tính. Thuật ngữ điều tra số ban đầu được sử dụng tương đương với điều tra máy tính nhưng sau đó được mở rộng để bao quát toàn bộ việc điều tra của tất cả các thiết bị có khả năng lưu trữ dữ liệu số.

Điều tra số có thể được định nghĩa là việc sử dụng các phương pháp, công cụ kỹ thuật khoa học đã được chứng minh để bảo quản, thu thập, xác nhận, chứng thực, phân tích, giải thích, lập báo cáo và trình bày lại những thông tin thực tế từ các nguồn kỹ thuật số với mục đích tạo điều kiện hoặc thúc đẩy việc tái hiện lại các sự kiện nhằm tìm ra hành vi phạm tội hay hỗ trợ cho việc dự đoán các hoạt động trái phép gây gián đoạn quá trình làm việc của hệ thống[11].

1.1.1. Lịch sử điều tra số

Trước những năm 1980, tội phạm liên quan đến máy tính đã được xử lý bằng pháp luật hiện hành. Tội phạm máy tính lần đầu tiên được ghi nhận trong Luật Tội phạm Máy tính Florida vào năm 1978, trong đó có bao gồm luật quy định về việc chống sửa đổi trái phép hay xóa dữ liệu trên một hệ thống máy tính. Trong những năm tiếp theo, phạm vi hoạt động của tội phạm máy tính tăng lên đáng kể, và pháp luật đã được thông qua để đối phó với vấn đề bản quyền tác giả, quyền riêng tư, hành vi quấy rối (như đe dọa, rình rập trên mạng hay kẻ thù trực tuyến) và khiêu dâm trẻ em. Mãi cho đến những năm 1980, luật liên bang mới bắt đầu kết hợp chặt chẽ với các hành vi phạm tội liên quan máy tính. Canada là quốc gia đầu tiên thực thi các luật về tội

phạm máy tính vào năm 1983. Sau đó là tổ chức chống Gian lận và Lạm dụng Máy tính của liên bang Mỹ vào năm 1986, Úc sửa đổi luật về tội phạm máy tính vào 1989 và Đạo luật của Anh vào 1990 quy định về các hành vi lạm dụng máy tính.

Giai đoạn năm 1980 – Đến 1990:

Sự phát triển gia tăng trong tội phạm máy tính những năm 1980 và 1990 là nguyên nhân để các cơ quan thực thi pháp luật bắt đầu thành lập các nhóm chuyên ngành cấp quốc gia để xử lý các khía cạnh kỹ thuật điều tra. Ví dụ năm 1984, FBI thành lập một nhóm ứng phó và phân tích các sự cố máy tính, sau đó một năm cục tội phạm máy tính được thành lập trực thuộc đội cảnh sát chống gian lận Anh.

Trong suốt những năm 1990 yêu cầu về nguồn lực điều tra để đáp ứng với sự gia tăng của tội phạm máy tính. Các đơn vị điều tra tội phạm công nghệ cao được thành lập ở Anh vào năm 2001 để cung cấp cơ sở hạ tầng quốc gia về tội phạm máy tính, bao gồm các nhân viên ở trung tâm London với các lực lượng cảnh sát nhiều vùng khác.

Trong thời gian này các kỹ thuật điều tra số đã phát triển, thuật ngữ “Computer Forensics” đã được sử dụng trong các tài liệu học thuật.

Việc thu giữ, bảo quản và phân tích chứng cứ được lưu trữ trên một máy tính là một trong những thách thức đối với việc điều tra khi phải đối mặt với việc đưa nó ra để làm bằng chứng phục vụ việc thực thi pháp luật trong những năm 1990. Mặc dù hầu hết các phân tích pháp y chẳng hạn như dấu vân tay, xét nghiệm AND, đều được thực hiện bởi các chuyên gia có nhiệm vụ thu thập và phân tích các chứng cứ máy tính thường được chuyển đến cho nhân viên điều tra và các thám tử.

Năm 2000: Phát triển các tiêu chuẩn

Từ năm 2000 để đáp ứng yêu cầu tiêu chuẩn hóa, các cơ quan và các hội đồng khác nhau đã công bố hướng dẫn kỹ thuật điều tra số. Nhóm công tác khoa học về chứng cứ số đã xuất bản một bài báo năm 2002 với tiêu đề “Best practices for Computer Forensics”. Đến năm 2005 công bố tiêu chuẩn ISO 17025 – đề cập đến các yêu cầu chung về thẩm quyền giám định và phòng thí nghiệm kiểm chuẩn. Năm 2004 hiệp định về tội phạm máy tính có hiệu lực, nhằm liên kết giữa các quốc gia với nhau trong việc điều tra các tội phạm liên quan đến công nghệ cao. Hiệp định đã được ký kết bởi 43 quốc gia[6].

1.1.2. Ứng dụng của điều tra số

Trong thời đại công nghệ phát triển mạnh như hiện nay. Song song với các ngành khoa học khác, điều tra số đã có những đóng góp rất quan trọng trong việc ứng cứu nhanh các sự cố xảy ra đối với máy tính, giúp các chuyên gia có thể phát hiện nhanh các dấu hiệu khi một hệ thống có nguy cơ bị xâm nhập, cũng như việc xác định được các hành vi, nguồn gốc của các vi phạm xảy ra đối với hệ thống.

Về mặt kỹ thuật thì điều tra số như: Điều tra mạng, điều tra bộ nhớ, điều tra các thiết bị điện thoại có thể giúp cho tổ chức xác định nhanh những gì đang xảy ra làm ảnh hưởng tới hệ thống, qua đó xác định được các điểm yếu để khắc phục, kiện toàn

Về mặt pháp lý thì điều tra số giúp cho cơ quan điều tra khi tố giác tội phạm công nghệ cao có được những chứng cứ số thuyết phục để áp dụng các chế tài xử phạt với các hành vi phạm pháp.

Một cuộc điều tra số thường bao gồm 3 giai đoạn: Tiếp nhận dữ liệu hoặc ảnh hóa tang vật, sau đó tiến hành phân tích và cuối cùng là báo cáo lại kết quả điều tra được.

Việc tiếp nhận dữ liệu đòi hỏi tạo ra một bản copy chính xác các sector hay còn gọi là nhân bản điều tra, của các phương tiện truyền thông, và để đảm bảo tính toàn vẹn của chứng cứ thu được thì những gì có được phải được băm sử dụng SHA1 hoặc MD5, và khi điều tra thì cần phải xác minh độ chính xác của các bản sao thu được nhờ giá trị đã băm trước đó.

Trong giai đoạn phân tích, thì các chuyên gia sử dụng các phương pháp nghiệp vụ, các kỹ thuật cũng như công cụ khác nhau để hỗ trợ điều tra, những kỹ thuật này sẽ được đề cập chi tiết ở chương 3 của đồ án.

Sau khi thu thập được những chứng cứ có giá trị và có tính thuyết phục thì tất cả phải được tài liệu hóa lại rõ ràng, chi tiết và báo cáo lại cho bộ phận có trách nhiệm xử lý chứng cứ thu được.

1.1.3. Quy trình thực hiện điều tra số

Một cuộc điều tra số thường bao gồm 4 gian đoạn: Chuẩn bị (Preparation), tiếp nhận dữ liệu hay còn gọi là ảnh hóa tang vật (Acquisition), phân tích (analysis) và lập báo cáo (Reporting)



Hình 1.1. Các bước thực hiện điều tra số

- Preparation: Bước này thực hiện việc mô tả lại thông tin hệ thống, những gì đã xảy ra, các dấu hiệu, để xác định phạm vi điều tra, mục đích cũng như các tài nguyên cần thiết sẽ sử dụng trong suốt quá trình điều tra.

- Acquisition: Đây là bước tạo ra một bản sao chính xác các sector hay còn gọi là nhân bản điều tra các phương tiện truyền thông, xác định rõ các nguồn chứng cứ sau đó thu thập và bảo vệ tính toàn vẹn của chứng cứ bằng việc sử dụng hàm băm mật mã. Tiếp nhận tang vật liên quan đến việc tạo ra một bản sao chính xác của các phương tiện truyền thông, thường sử dụng một

thiết bị cắm ghi đề để ngăn ngừa sự thay đổi so với bản gốc. Cả bản sao lẫn bản gốc đều được băm (sử dụng SHA1 hoặc MD5) để so sánh với nhau nhằm xác minh bản sao là chính xác.

- Analysis: Đây là giai đoạn các chuyên gia sử dụng các phương pháp nghiệp vụ, các kỹ thuật cũng như công cụ khác nhau để trích xuất, thu thập và phân tích các bằng chứng thu được. Trong giai đoạn phân tích, điều tra viên sẽ sử dụng các phương pháp và công cụ khác nhau. Năm 2002, một bài báo trên Tạp chí Quốc tế về tang chứng kỹ thuật số gọi bước này là “một hệ thống tìm kiếm chuyên sâu về bằng chứng liên quan đến các kẻ tình nghi”. Năm 2006, nhà nghiên cứu pháp y Brian Carrie mô tả một “thủ tục trực quan” trong đó bằng chứng rõ ràng sẽ được xác định đầu tiên và sau đó “tìm kiếm toàn diện được tiến hành để bắt đầu làm đầy các chỗ trống”.

Quá trình thực tế của phân tích có thể khác nhau giữa các cuộc điều tra, nhưng các phương pháp thông thường bao gồm tiến hành tìm kiếm từ khóa trên các phương tiện truyền thông số (trong tập tin cũng như không gian lỏng và chưa phân bổ), phục hồi các tập tin đã xóa và khai thác các thông tin dẫn kí (ví dụ để liệt kê danh sách tài khoản người dùng, các thiết bị USB kèm theo...)

- Reporting: Sau khi thu thập được những chứng cứ có giá trị và có tính thuyết phục thì tất cả phải được tài liệu hóa lại rõ ràng, chi tiết và báo cáo lại cho bộ phận có trách nhiệm xử lý chứng cứ thu được[6].

1.1.4. Các loại hình điều tra số phổ biến

- Điều tra máy tính

Điều tra máy tính (Computer Forensics) là một nhánh của khoa học điều tra số liên quan đến việc phân tích các bằng chứng pháp lý được tìm thấy trong máy tính và các phương tiện lưu trữ kỹ thuật số như:

- Điều tra bản ghi (Registry Forensics) là việc trích xuất thông tin và

ngữ cảnh từ một nguồn dữ liệu chưa được khai thác qua đó biết được những thay đổi (chỉnh sửa, thêm bớt...) dữ liệu trong bản ghi (Register).

- Điều tra bộ nhớ (Memory Forensics) là việc ghi lại bộ nhớ khả biến (bộ nhớ RAM) của hệ thống sau đó tiến hành phân tích làm rõ các hành vi đã xảy ra trên hệ thống. Để xác định các hành vi đã xảy ra trong hệ thống, người ta thường sử dụng kiến trúc quản lý bộ nhớ trong máy tính để ánh xạ, trích xuất các tập tin đang thực thi và cư trú trong bộ nhớ.

- Điều tra phương tiện lưu trữ (Disk Forensics) là việc thu thập, phân tích dữ liệu được lưu trữ trên phương tiện lưu trữ vật lý, nhằm trích xuất dữ liệu ẩn, khôi phục các tập tin bị xóa, qua đó xác định người đã tạo ra những thay đổi dữ liệu trên thiết bị được phân tích.

Mục đích của điều tra máy tính là nhằm xác định, bảo quản, phục hồi, phân tích, trình bày lại sự việc và ý kiến về các thông tin thu được từ thiết bị kỹ thuật số. Mặc dù thường được kết hợp với việc điều tra một loạt các tội phạm máy tính, điều tra máy tính cũng có thể được sử dụng trong tố tụng dân sự. Bằng chứng thu được từ các cuộc điều tra máy tính thường phải tuân theo những nguyên tắc và thông lệ như những bằng chứng kỹ thuật số khác. Nó đã được sử dụng trong một số trường hợp có hồ sơ cao cấp và đang được chấp nhận rộng rãi trong các hệ thống tòa án Mỹ và Châu Âu.

Các bước thực hiện điều tra máy tính:

- Chuẩn bị: Kiểm soát hệ thống máy tính để chắc chắn rằng thiết bị và dữ liệu được an toàn. Điều này có nghĩa điều tra viên cần phải nắm quyền bảo mật để không có một cá nhân nào có thể truy cập máy tính và thiết bị lưu trữ đang được kiểm tra. Nếu hệ thống máy tính có kết nối với Internet, điều tra viên phải kiểm soát được kết nối này.

- Thu thập dữ liệu: Tìm kiếm tất cả các tập tin có trong hệ thống máy tính, bao gồm các tập tin đã được mã hóa, được bảo vệ bằng mật khẩu, được

ẩn hoặc bị xóa nhưng chưa bị ghi đè. Nhân viên điều tra nên sao chép lại tất cả các tập tin của hệ thống, bao gồm các tập tin có trong ổ đĩa của máy tính hay tập tin từ các ổ cứng cắm ngoài. Bởi khi truy cập các tập tin có thể thay đổi nó nên nhân viên điều tra chỉ nên làm việc với các bản copy của các tập tin khi tìm kiếm bằng chứng. Bản nguyên gốc cần được bảo quản và không được động đến. Khôi phục lại càng nhiều thông tin bị xóa càng tốt bằng cách sử dụng các ứng dụng có thể tìm kiếm và truy hồi dữ liệu bị xóa. Tìm kiếm thông tin của tất cả các tập tin ẩn.

- Phân tích: Giải mã và truy cập các tập tin được bảo vệ. Phân tích các khu vực đặc biệt trên ổ đĩa máy tính, bao gồm các phần thường khó có thể tiếp cận.

- Báo cáo: Ghi lại tất cả các bước của quá trình. Điều này rất quan trọng đối với nhân viên điều tra để cung cấp bằng chứng rằng công việc điều tra của họ thực hiện có bảo vệ thông tin của hệ thống máy tính mà không làm thay đổi hoặc làm hỏng chúng. Những tài liệu xác thực này không chỉ bao gồm các tập tin và dữ liệu được khôi phục từ hệ thống mà còn bao gồm cả bản vẽ của hệ thống và nơi các tập tin được mã hóa hoặc được ẩn[1],[4].

- Điều tra mạng

Điều tra mạng (Network Forensics) là một nhánh của khoa học điều tra số liên quan đến việc giám sát và phân tích lưu lượng mạng máy tính nhằm phục vụ cho việc thu thập thông tin, chứng cứ pháp lý hay phát hiện các xâm nhập. Điều tra mạng cũng được hiểu như điều tra số trong *môi-trường-mạng*.

Điều tra mạng là một lĩnh vực tương đối mới của khoa học pháp y. Sự phát triển mỗi ngày của Internet đồng nghĩa với việc máy tính đã trở thành mạng lưới trung tâm và dữ liệu bây giờ đã khả dụng trên các chứng cứ số nằm trên đĩa. Điều tra mạng có thể được thực hiện như một cuộc điều tra độc lập hoặc kết hợp với việc phân tích pháp y máy tính (*computer forensics*) –

thường được sử dụng để phát hiện mối liên kết giữa các thiết bị kỹ thuật số hay tái tạo lại quy trình phạm tội.

- Điều tra thiết bị di động

Điều tra thiết bị di động (Mobile device Forensics) là một nhánh của khoa học điều tra số liên quan đến việc thu hồi bằng chứng kỹ thuật số hoặc dữ liệu từ các thiết bị di động. Thiết bị di động ở đây không chỉ đề cập đến điện thoại di động mà còn là bất kỳ thiết bị kỹ thuật số nào có bộ nhớ trong và khả năng giao tiếp, bao gồm các thiết bị PDA, GPS và máy tính bảng.

Việc sử dụng điện thoại với mục đích phạm tội đã phát triển rộng rãi trong những năm gần đây, nhưng các nghiên cứu điều tra về thiết bị di động là một lĩnh vực tương đối mới, có niên đại từ những năm 2000. Sự gia tăng các loại hình điện thoại di động trên thị trường (đặc biệt là điện thoại thông minh) đòi hỏi nhu cầu giám định các thiết bị này mà không thể đáp ứng bằng các kỹ thuật điều tra máy tính hiện tại.

Các bước thực hiện điều tra thiết bị di động:

Theo tài liệu của Viện Tiêu chuẩn và Kỹ thuật Quốc gia Hoa Kỳ (NIST), điều tra thiết bị di động được chia làm 5 giai đoạn chính: Chuẩn bị (Preparation), thu thập dữ liệu (Acquisition), kiểm tra (Examination), phân tích (Analysis) và lập báo cáo (Reporting).



Hình 1.2. Các bước thực hiện điều tra di động

- Chuẩn bị: Giai đoạn này bao gồm các bước trao đổi thông tin ban

đầu, xây dựng kế hoạch và chuẩn bị cho các bước điều tra. Trước khi điều tra một đối tượng phải được sự thoả thuận và ký kết chính thức từ các bên tham gia, nhằm tạo cơ sở pháp lý đảm bảo trong quá trình điều tra, những thông tin quan trọng không bị rò rỉ. Những mô tả lại thông tin hệ thống, những hành vi đã xảy ra, các dấu hiệu để xác định phạm vi điều tra, mục đích cũng như các tài nguyên cần thiết sẽ được sử dụng trong suốt quá trình điều tra.

- Thu thập dữ liệu: Qua tiếp xúc trực tiếp với thiết bị, dữ liệu thu thập càng nhiều thì khả năng thu được những bằng chứng số càng lớn. Đối với thiết bị không yêu cầu về mật khẩu hoặc các kỹ thuật xác thực truy cập, người điều tra có thể truy cập dữ liệu người dùng ở những vùng nhớ khác nhau, từ đó thực hiện hướng điều tra tiếp theo. Nhưng với các thiết bị yêu cầu mật khẩu và các kỹ thuật xác thực người dùng, thì người điều tra cần phải vượt qua cơ chế xác thực. Nếu việc này không thành công thì dễ dẫn đến hiện tượng bị mất hoàn toàn dữ liệu và việc khôi phục thông tin sẽ rất khó khăn. Khi đó, người điều tra cần phải sử dụng phần mềm khác hoặc can thiệp tới nền tảng phần cứng để vượt qua lớp xác thực trên thiết bị. Sau khi tiếp cận được dữ liệu, cần tiến hành thực hiện nhận dạng và kiểm tra bộ nhớ thiết bị di động.

Nhận dạng thiết bị di động: Để nhận dạng được thiết bị di động, cần tiến hành thực hiện việc kiểm tra đặc điểm của thiết bị, đặc điểm của phụ kiện thiết bị, nhãn hiệu thiết bị và thông tin nhà cung cấp, nhà mạng xác định vị trí thiết bị. Trên mỗi thiết bị di động thường có chứa các thông số định danh duy nhất, mà có thể dễ nhận thấy trên bản thân thiết bị hoặc các phụ kiện đi kèm như: dãy số nhận dạng thiết bị di động (IMEI), Model, ESN, thông tin thẻ SIM,... từ đó thực hiện tra cứu các thông tin về thông số kỹ thuật, tính năng, loại và nhà sản xuất của điện thoại cần điều tra.

Kiểm tra bộ nhớ: Cần phải tiến hành kiểm tra toàn bộ các bộ nhớ của thiết bị di động nhằm thu được nhiều chứng cứ số. Việc kiểm tra có thể được tiến hành trên 2 bộ nhớ phổ biến như:

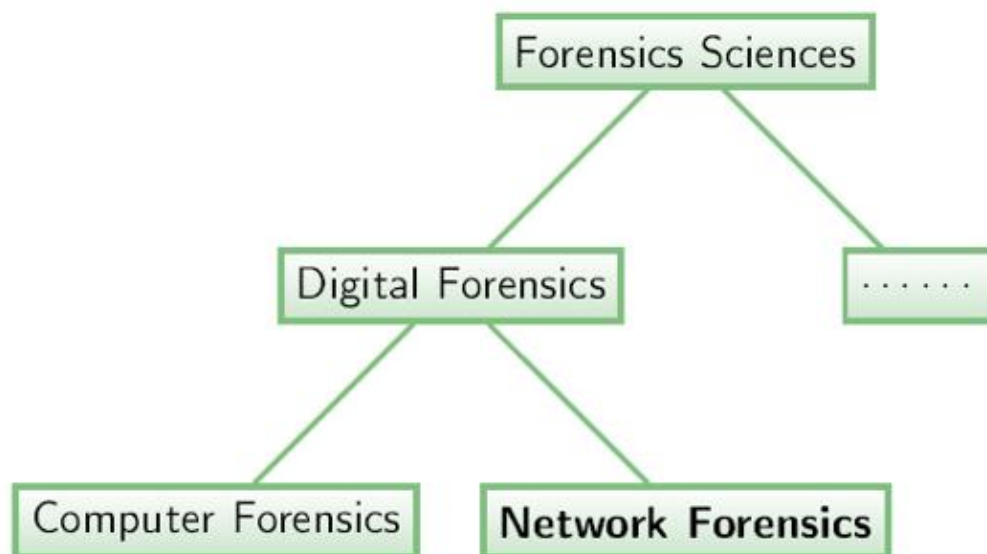
Bộ nhớ điện thoại dùng để lưu trữ hệ điều hành, bao gồm: nhân, trình điều khiển và hệ thống hàm thư viện dùng để thực thi ứng dụng, hệ điều hành. Ngoài ra, nó còn được dùng để lưu trữ ứng dụng của người dùng và các dữ liệu khác (văn bản, hình ảnh, âm thanh, video....).

Bộ nhớ SIM gồm các dữ liệu về các dịch vụ, định danh duy nhất cho SIM, số thuê bao, danh bạ và thông tin về các liên lạc đã được thực hiện.

- **Kiểm tra và phân tích:** Quá trình kiểm tra và phân tích thường được tiến hành song song. Kiểm tra nhằm phát hiện ra bằng chứng số, gồm các bằng chứng bị ẩn hoặc bị che khuất, nhằm hiển thị nội dung và trạng thái của các dữ liệu một cách đầy đủ cả nguồn thông tin và ý nghĩa tiềm ẩn. Quá trình phân tích dựa trên kết quả kiểm tra để xác định các thông tin có ý nghĩa trực tiếp đối với từng trường hợp điều tra. Kết quả của giai đoạn này gồm các bằng chứng tiềm năng và hồ sơ thuê bao.

- **Báo cáo:** Đây là quá trình chuẩn bị một bản báo cáo chi tiết tất cả các bước đã thực hiện và kết luận đạt được trong cuộc điều tra của một trường hợp cụ thể. Báo cáo kết quả điều tra số phải mang tính khách quan, phản ánh trung thực những tình tiết xảy ra, có liên quan trực tiếp hoặc gián tiếp tới vụ việc và mang tính hợp pháp. Nội dung báo cáo phải cung cấp đầy đủ các thông tin cần thiết để xác định nguồn gốc, tình tiết, đưa ra những chứng cứ số được phát hiện trong quá trình điều tra[2].

1.2. Giới thiệu về phân tích điều tra mạng (Network Forensics)



Hình 1.3. Network Forensics trong Forensics Sciences

Thuật ngữ *Network Forensics* (điều tra mạng) được đưa ra bởi chuyên gia bảo mật máy tính Marcus Ranum vào đầu những năm 90, vay mượn từ các lĩnh vực pháp luật và tội phạm nơi mà “*forensics*” gắn liền với việc điều tra các hành vi phạm tội.

Network Forensics là một nhánh của điều tra số liên quan đến việc giám sát và phân tích lưu lượng mạng máy tính nhằm phục vụ cho việc thu thập thông tin, chứng cứ pháp lý hay phát hiện các xâm nhập. Network Forensics cũng được hiểu như Digital Forensics trong môi trường mạng.

Về cơ bản, điều tra mạng là việc chặn bắt, ghi âm và phân tích các sự kiện mạng để khám phá nguồn gốc của các cuộc tấn công hoặc sự cố của một vấn đề nào đó.

Không giống các mảng khác của điều tra số, điều tra mạng giải quyết những thông tin dễ thay đổi và biến động. Lưu lượng mạng được truyền đi và sau đó bị mất, do đó điều tra mạng thường là cuộc điều tra rất linh hoạt, chủ động.

Trong môi trường hiện nay, điều tra mạng thường được thực hiện để phân tích sự xung đột diễn ra giữa những kẻ tấn công và người phòng thủ. Thông thường, các điều tra viên cố gắng ngăn chặn sự bùng phát sâu máy tính, điều tra hành vi vi phạm, thu thập chứng cứ cho tòa án. Các kỹ năng, kỹ thuật cần thiết cho việc phân tích pháp y mạng rất sâu rộng và nâng cao, cùng một nhà điều tra có thể được kêu gọi để khai thác bộ nhớ cache từ web proxy hay sniff thụ động lưu lượng truy cập mạng và xác định các hoạt động đáng ngờ...

Hầu hết các kỹ thuật hiện này là giám sát thụ động, chủ yếu dựa trên lưu lượng mạng, hiệu năng CPU hoặc quá trình nhập/ xuất (Input/Output) với sự can thiệp của con người. Trong đa số các trường hợp, dấu hiệu của cuộc tấn công mới được phát hiện thủ công hoặc trong một số trường hợp nó không bị phát hiện cho đến khi vụ việc được báo cáo. Trọng tâm của lĩnh vực pháp y mạng là để tự động hóa quá trình phát hiện tất cả các cuộc tấn công và thêm vào đó ngăn chặn các thiệt hại do vi phạm an ninh. Ý tưởng chính của network forensics là xác định tất cả các vi phạm an ninh có thể xảy ra và xây dựng các dấu hiệu vào cơ chế phát hiện và ngăn chặn để hạn chế những mất mát về sau[3],[4],[5].

Một số điểm lưu ý khi nói đến Network Forensics

- Nó không phải là một sản phẩm (product) mà là một tiến trình (process) phức tạp (bao gồm các công cụ kỹ thuật, trí tuệ con người, luật pháp...)
- Nó không thay thế cho tường lửa, IDS, IPS...
- Nó sử dụng các cảnh báo IDS, nhật ký của tường lửa, các gói tin...

1.2.1. Vai trò và ứng dụng của phân tích điều tra mạng

- Vai trò: Sự tăng trưởng của các kết nối mạng và sự phức tạp trong các hoạt động trên mạng đã đi kèm với sự gia tăng số lượng tội phạm mạng buộc cả doanh nghiệp cũng như cơ quan thực thi pháp luật phải vào cuộc để thực hiện các điều tra, phân tích. Công việc này có những khó khăn đặc biệt trong thế giới ảo, vấn đề lớn đối với một điều tra viên là hiểu được những dữ liệu số ở mức thấp nhất cũng như việc sắp xếp, tái tạo lại chúng.

Thuật ngữ điều tra mạng được đưa ra bởi chuyên gia bảo mật máy tính Marcus Ranum vào đầu những năm 90 thế kỷ XX. Điều tra mạng là một loại hình của điều tra số liên quan đến việc giám sát và phân tích lưu lượng mạng máy tính nhằm phục vụ cho việc thu thập thông tin, chứng cứ pháp lý hay phát hiện các xâm nhập. Không giống các loại hình khác của điều tra số, điều tra mạng xử lý những thông tin dễ thay đổi và biến động, khó dự đoán. Lưu lượng mạng được truyền đi và sau đó bị mất, do đó việc điều tra được diễn ra rất linh hoạt, chủ động. Các điều tra viên chỉ có thể dựa vào thông tin từ các thiết bị an toàn như bộ lọc gói, tường lửa, hệ thống phát hiện xâm nhập đã được triển khai để dự đoán hành vi vi phạm. Các kỹ năng, kỹ thuật cần thiết cho việc điều tra mạng phức tạp và chuyên sâu, sử dụng thông tin được khai thác từ bộ nhớ đệm (cache) của web, proxy hay chặn bắt thụ động lưu lượng truy cập mạng và xác định các hành vi bất thường.

- Ứng dụng: Tại Việt Nam, vấn đề khắc phục sự cố về an toàn thông tin cũng như điều tra tìm hiểu nguồn gốc tấn công đang ở giai đoạn bắt đầu phát triển. Các nghiên cứu về điều tra mạng ở Việt Nam vẫn còn nhiều hạn chế, chưa tiếp cận được trình độ khoa học - kỹ thuật của các nước phát triển cũng như chưa xây dựng được bộ công cụ riêng phục vụ công tác điều tra mạng.

Mục tiêu quan trọng nhất của phân tích điều tra mạng là cung cấp đầy đủ chứng cứ để có thể khởi tố một tội phạm hình sự. Ứng dụng thực tế của phân tích điều tra mạng có thể là trong các lĩnh vực như hacking, lừa đảo, các công ty bảo hiểm, trộm cắp thông tin nhạy cảm, xuyên tạc, sao chép thẻ tín dụng, vi phạm bản quyền phần mềm, can thiệp vào quá trình bầu cử, phát tán những văn hóa phẩm đồi trụy, khai man, quấy rối tình dục, phân biệt chủng tộc và thậm chí là cả giết người.

1.2.2. Nền tảng kỹ thuật cho phân tích điều tra mạng

- Hệ điều hành và các dịch vụ mạng phổ biến

- *Các dạng hệ điều hành*

Hệ điều hành là một phần mềm chạy trên máy tính, dùng để điều hành, quản lý các thiết bị phần cứng và các tài nguyên phần mềm trên máy tính.

Hệ điều hành đóng vai trò trung gian trong việc giao tiếp giữa người sử dụng và phần cứng máy tính, cung cấp một môi trường cho phép người sử dụng phát triển và thực hiện các ứng dụng của họ một cách dễ dàng.

Hệ điều hành theo hình thức xử lý được chia làm 5 loại chính:

Hệ đa xử lý (Multiprocessor Systems), các CPU dùng chung bộ nhớ và thiết bị, gồm:

- Hệ xử lý đối xứng - Các CPU ngang hàng về chức năng (OS: Solaris, Linux, Microsoft Windows NT trở lên, OS/2)
- Hệ xử lý phi đối xứng - Các CPU được ấn định chức năng riêng, có 1 CPU master điều khiển các CPU phụ (Slaves) (OS: SunOS 4.x)

Hệ phân tán (Distributed Systems)

- Kết nối với nhau qua giao tiếp mạng
- Phân loại theo khoảng cách (LAN, WAN, MAN)
- Phân loại theo phương thức phục vụ (File-Server, Peer-to-peer, Client-Server)

Hệ gom cụm (Clustered Systems), nhiều máy nối mạng để làm chung một công việc, phân loại:

- Gom cụm đối xứng (Symmetric Clustering) - Các máy ngang hàng về chức năng
- Gom cụm phi đối xứng (Asymmetric Clustering) - Có máy chạy trong Hot Standby Mode giám sát các máy khác

Hệ thời gian thực (Real-Time Systems)

- Thời gian thực chặt (Hard Real-Time) - Có thời gian giới tuyến Deadline đã định, quá thời gian này sẽ hư hỏng
- Thời gian thực lỏng (Soft Real-Time) - Trung bình thì đáp ứng được thời gian, nhưng trong một số trường hợp đặc biệt sẽ bị chậm một chút, nhưng ko bị hư hỏng và ảnh hưởng đến toàn hệ

Hệ cầm tay (Handheld Systems) - Các OS cho điện thoại, hoặc PDA (OS: Palm, Sysbian, iOS, Windows Pocket PC, Windows Mobile, Windows Mobile, Android,...)

• *Các định dạng file của hệ điều hành*

Mỗi hệ điều hành có những quy định riêng về định dạng file, thường dựa vào phần mở rộng của tên file. Phần mở rộng của tên file có thể được coi là một loại siêu dữ liệu (metadata). Chúng thường được dùng để bao hàm thông tin về cách thức dữ liệu được lưu trữ trong tệp tin. Việc định nghĩa chính xác đưa ra các tiêu chí quyết định phần nào của tên file là phần mở rộng; thường phần mở rộng là phần xuất hiện sau cùng (nếu có) của tên tệp tin (ví dụ txt là phần mở rộng của tệp tin readme.txt, html là phần mở rộng của mysite.index.html). Trên hệ thống tệp tin của những máy tính lớn (mainframe) như MVS, VMS hay CP/M, MS-DOS, phần mở rộng là chuỗi kí tự tính từ sau khoảng trống được phân tách từ tên tệp tin. Đối với hệ điều hành như Windows, phần mở rộng như .exe, .com hoặc .bat chỉ ra một tệp tin

là một chương trình thực thi.

Các hệ thống tệp tin thuộc họ Unix sử dụng một mô hình khác mà không có kiểu siêu dữ liệu với phần mở rộng tách biệt. Dấu chấm chỉ là một kí tự trong tên tệp tin chính và tên tệp tin có thể có nhiều phần mở rộng, thường đại diện cho những sự chuyển đổi lồng nhau, chẳng hạn như files.tar.gz. Mô hình này thường đòi hỏi tên tệp tin phải đầy đủ để cung cấp trong dòng lệnh, nơi mà các siêu dữ liệu thường được cho phép bỏ qua phần mở rộng.

Những phiên bản OS X trước hệ điều hành MacOS bỏ hoàn toàn việc sử dụng phần mở rộng dựa vào tên tệp tin của siêu dữ liệu, thay vào đó sử dụng một mã tệp tin riêng để xác định các định dạng tệp tin. Thêm vào đó, một mã khởi tạo được chỉ định để xác định ứng dụng nào sẽ được gọi khi nhấp đúp vào tệp tin. Tuy nhiên Mac OS X sử dụng hậu tố tên tệp tin, cũng như mã tệp tin và mã khởi tạo, nó có nguồn gốc từ Unix – tương tự như hệ điều hành NeXTSTEP.

- *Các dịch vụ mạng phổ biến*

Dịch vụ xác thực: cung cấp cơ chế xác thực cho người sử dụng hoặc các hệ thống thông qua mạng. Người sử dụng và các máy chủ sẽ nhận vé mã hóa, những vé này sau đó được trao đổi với nhau để xác minh danh tính.

Dịch vụ thư mục: là hệ thống phần mềm lưu trữ, tổ chức và cung cấp quyền truy cập vào thông tin trong một thư mục. Trong công nghệ phần mềm, một thư mục là một ánh xạ giữa tên với giá trị. Nó cho phép tra cứu các giá trị cho một cái tên, tương tự như một từ điển.

DHCP: Là một giao thức cấu hình tự động địa chỉ IP. Máy tính được cấu hình một cách tự động vì thế sẽ giảm việc can thiệp vào hệ thống mạng. Nó cung cấp một database trung tâm để theo dõi tất cả các máy tính trong hệ thống mạng. Mục đích quan trọng nhất là tránh trường hợp hai máy tính khác nhau lại có cùng địa chỉ IP.

Nếu không có DHCP, các máy có thể cấu hình IP thủ công. Ngoài việc cung cấp địa chỉ IP, DHCP còn cung cấp thông tin cấu hình khác, cụ thể như DNS. Hiện nay DHCP có 2 version: cho IPv4 và IPv6.

DNS: Được phát minh vào năm 1984 cho Internet, là một hệ thống cho phép thiết lập tương ứng giữa địa chỉ IP và tên miền. Hệ thống tên miền (DNS) là một hệ thống đặt tên theo thứ tự cho máy vi tính, dịch vụ, hoặc bất kỳ nguồn lực tham gia vào Internet. Nó liên kết nhiều thông tin đa dạng với tên miền được gán cho những người tham gia. Quan trọng nhất là nó chuyển tên miền có ý nghĩa cho con người vào số định danh (nhị phân), liên kết với các trang thiết bị mạng cho các mục đích định vị và địa chỉ hóa các thiết bị khắp thế giới.

DNS phục vụ như một “Danh bạ điện thoại” để tìm trên Internet bằng cách dịch tên máy chủ máy tính thành địa chỉ IP. Ví dụ, www.example.com dịch thành 208.77.188.166.

Mọi người tận dụng lợi thế này khi họ sử dụng các URL có nghĩa và địa chỉ email mà không cần phải biết làm thế nào các máy sẽ thực sự tìm ra chúng.

Hệ thống tên miền cũng lưu trữ các loại thông tin khác, chẳng hạn như danh sách các máy chủ email chấp nhận thư điện tử cho một tên miền Internet. Bằng cách cung cấp cho một thế giới rộng lớn, phân phối từ khóa – cơ sở của dịch vụ đổi hướng, Hệ thống tên miền là một thành phần thiết yếu cho các chức năng của Internet. Các định dạng khác như các thẻ RFID, mã số UPC, kí tự Quốc tế trong địa chỉ email và tên máy chủ, và một loạt các định dạng khác có thể có khả năng sử dụng DNS

Email (electronic mail – Thư điện tử) là một hệ thống chuyển nhận thư qua các mạng máy tính.

Email là một phương tiện thông tin rất nhanh. Một mẫu thông tin có thể

được gửi đi ở dạng mã hoá hay dạng thông thường và được chuyển qua các mạng máy tính đặc biệt là mạng Internet. Nó có thể chuyển mẫu thông tin từ một máy nguồn tới một hay rất nhiều máy nhận trong cùng lúc.

Ngày nay, email chẳng những có thể truyền gửi được chữ, nó còn có thể truyền được các dạng thông tin khác như hình ảnh, âm thanh, phim, và đặc biệt các phần mềm thư điện tử kiểu mới còn có thể hiển thị các email dạng sống động tương thích với kiểu tệp HTML.

File sharing (chia sẻ tệp tin) là việc phân phối hoặc cung cấp quyền truy cập vào các thông tin được lưu trữ dạng số, chẳng hạn như các chương trình máy tính, đa phương tiện (âm thanh, hình ảnh, video), tài liệu hoặc sách điện tử. Nó có thể được thực hiện thông qua nhiều cách khác nhau. Phương pháp phổ biến của lưu trữ, truyền tải và phân tán bao gồm chia sẻ thủ công bằng các phương tiện di động, các máy chủ tập trung trên mạng máy tính, các tài liệu siêu liên kết trên nền web và việc sử dụng mạng phân phối ngang hàng.

IM - Instant Messaging (tin nhắn nhanh hay trò chuyện trực tuyến, chat) là dịch vụ cho phép hai người trở lên nói chuyện trực tuyến với nhau qua một mạng máy tính.

Mới hơn IRC (Chat chuyển tiếp internet), nhắn tin nhanh là trò chuyện mạng, phương pháp nói chuyện phổ biến hiện nay. Nhắn tin nhanh dễ dùng hơn IRC, và có nhiều tính năng hay, như khả năng trò chuyện nhóm, dùng biểu tượng xúc cảm, truyền tệp tin, tìm dịch vụ và cấu hình dễ dàng bản liệt kê bạn bè.

Nhắn tin nhanh đã thúc đẩy sự phát triển của Internet trong đầu thập niên 2000.

File server (Máy chủ tệp tin) là một máy tính nằm trên mạng có chức năng chính là cung cấp một vị trí để truy cập vào ổ đĩa chia sẻ, nghĩa là lưu

trữ các tệp tin được chia sẻ trên máy tính (chẳng hạn như tài liệu, tệp tin âm thanh, hình ảnh, phim, cơ sở dữ liệu...) có thể được truy cập bởi các máy trạm có kết nối với máy chủ này.

Thuật ngữ server nêu lên vai trò của máy tính trong mô hình client-server, nơi mà các máy khách là các máy trạm sử dụng dữ liệu lưu trữ. Một file server không thực hiện nhiệm vụ tính toán và không chạy các chương trình thay cho máy khách. Nó được thiết kế chủ yếu để lưu trữ và cho phép truy xuất dữ liệu trong khi các tính toán được thực hiện ở phía máy trạm.

VoIP (Voice over IP) dùng để chỉ các giao thức truyền thông, phương pháp và kỹ thuật truyền dẫn liên quan đến việc cung cấp các thông tin liên lạc thoại và các phiên đa phương tiện qua giao thức Internet (IP). Các thuật ngữ khác liên quan đến VoIP là điện thoại IP, điện thoại Internet, thoại qua băng thông rộng (VoBB), truyền thông IP và điện thoại băng thông rộng.

VoIP có sẵn trên nhiều điện thoại thông minh và các thiết bị kết nối Internet giúp người dùng có thể thực hiện các cuộc gọi hoặc gửi tin nhắn văn bản qua mạng 3G hoặc Wi-Fi.

World Wide Web (hay Web hoặc WWW - mạng lưới toàn cầu) là một không gian thông tin toàn cầu mà mọi người có thể truy nhập (đọc và viết) qua các máy tính nối với mạng Internet. Thuật ngữ này thường được hiểu nhầm là từ đồng nghĩa với chính thuật ngữ Internet. Nhưng Web thực ra chỉ là một trong các dịch vụ chạy trên Internet, chẳng hạn như dịch vụ thư điện tử. Web được phát minh và đưa vào sử dụng vào khoảng năm 1990, 1991 bởi viện sĩ Viện Hàn lâm Anh Tim Berners-Lee và Robert Cailliau (Bỉ) tại CERN, Geneva, Switzerland

Các tài liệu trên World Wide Web được lưu trữ trong một hệ thống siêu văn bản (hypertext), đặt tại các máy tính trong mạng Internet. Người dùng phải sử dụng một chương trình được gọi là trình duyệt web (web browser) để

xem siêu văn bản. Chương trình này sẽ nhận thông tin (documents) tại ô địa chỉ (address) do người sử dụng yêu cầu (thông tin trong ô địa chỉ được gọi là tên miền (domain name)), rồi sau đó chương trình sẽ tự động gửi thông tin đến máy chủ (web server) và hiển thị trên màn hình máy tính của người xem. Người dùng có thể theo các liên kết siêu văn bản (hyperlink) trên mỗi trang web để nối với các tài liệu khác hoặc gửi thông tin phản hồi theo máy chủ trong một quá trình tương tác. Hoạt động truy tìm theo các siêu liên kết thường được gọi là duyệt Web.

Quá trình này cho phép người dùng có thể lướt các trang web để lấy thông tin. Tuy nhiên độ chính xác và chứng thực của thông tin không được đảm bảo.

- Giao thức mạng phổ biến

- *Giao thức IP*: Là một giao thức hướng dữ liệu được sử dụng bởi các máy chủ nguồn và đích để truyền dữ liệu trong một liên mạng chuyển mạch gói.

Dữ liệu trong một liên mạng IP được gửi theo các khối được gọi là các gói (packet hoặc datagram). Cụ thể, IP không cần thiết lập các đường truyền trước khi một máy chủ gửi các gói tin cho một máy khác mà trước đó nó chưa từng liên lạc với.

Giao thức IP cung cấp một dịch vụ gửi dữ liệu không đảm bảo (còn gọi là cố gắng cao nhất), nghĩa là nó hầu như không đảm bảo gì về gói dữ liệu. Gói dữ liệu có thể đến nơi mà không còn nguyên vẹn, nó có thể đến không theo thứ tự (so với các gói khác được gửi giữa hai máy nguồn và đích đó), nó có thể bị trùng lặp hoặc bị mất hoàn toàn. Nếu một phần mềm ứng dụng cần được bảo đảm, nó có thể được cung cấp từ nơi khác, thường từ các giao thức giao vận nằm phía trên IP.

IP v4

Giao thức Internet phiên bản 4 (viết tắt IPv4, từ tiếng Anh Internet Protocol version 4) là phiên bản thứ tư trong quá trình phát triển của các giao thức Internet (IP). Đây là phiên bản đầu tiên của IP được sử dụng rộng rãi. IPv4 cùng với IPv6 (giao thức Internet phiên bản 6) là nòng cốt của giao tiếp internet. Hiện tại, IPv4 vẫn là giao thức được triển khai rộng rãi nhất trong bộ giao thức của lớp internet.

Giao thức này được công bố bởi IETF trong phiên bản RFC 791 (tháng 9 năm 1981), thay thế cho phiên bản RFC 760 (công bố vào tháng giêng năm 1980). Giao thức này cũng được chuẩn hóa bởi bộ quốc phòng Mỹ trong phiên bản MIL-STD-1777.

IPv4 là giao thức hướng dữ liệu, được sử dụng cho hệ thống chuyển mạch gói (tương tự như chuẩn mạng Ethernet). Đây là giao thức truyền dữ liệu hoạt động dựa trên nguyên tắc tốt nhất có thể, trong đó, nó không quan tâm đến thứ tự truyền gói tin cũng như không đảm bảo gói tin sẽ đến đích hay việc gây ra tình trạng lặp gói tin ở đích đến. Việc xử lý vấn đề này dành cho lớp trên của chồng giao thức TCP/IP. Tuy nhiên, IPv4 có cơ chế đảm bảo tính toàn vẹn dữ liệu thông qua sử dụng những gói kiểm tra (checksum).

IPv4 sử dụng 32 bits để đánh địa chỉ, theo đó, số địa chỉ tối đa có thể sử dụng là 4.294.967.296 (2³²). Tuy nhiên, do một số được sử dụng cho các mục đích khác như: cấp cho mạng cá nhân (xấp xỉ 18 triệu địa chỉ), hoặc sử dụng làm địa chỉ quảng bá (xấp xỉ 16 triệu), nên số lượng địa chỉ thực tế có thể sử dụng cho mạng Internet công cộng bị giảm xuống. Với sự phát triển không ngừng của mạng Internet, nguy cơ thiếu hụt địa chỉ đã được dự báo, tuy nhiên, nhờ công nghệ NAT (Network Address Translation - Chuyển dịch địa chỉ mạng) tạo nên hai vùng mạng riêng biệt: Mạng riêng và Mạng công cộng, địa chỉ mạng sử dụng ở mạng riêng có thể dùng lại ở mạng công cộng mà

không hề bị xung đột, qua đó trì hoãn được vấn đề thiếu hụt địa chỉ

Chuẩn IPv6, với số lượng bits dùng để đánh địa chỉ nhiều hơn đã được xây dựng nhằm thay thế IPv4 trong tương lai.

IP v6

IPv6, viết tắt tiếng Anh: "Internet Protocol version 6", là "Giao thức liên mạng thế hệ 6", một phiên bản của giao thức liên mạng (IP) nhằm mục đích nâng cấp giao thức liên mạng phiên bản 4 (IPv4) hiện đang truyền dẫn cho hầu hết lưu lượng truy cập Internet nhưng đã hết địa chỉ. IPv6 cho phép tăng lên đến 2¹²⁸ địa chỉ, một sự gia tăng khổng lồ so với 2³² (khoảng 4.3 tỷ) địa chỉ của IPv4.

Để đưa IPv6 vào sử dụng, hầu hết các máy chủ trên mạng Internet cũng như các mạng lưới kết nối với chúng sẽ cần phải triển khai giao thức này với một quá trình chuyển đổi khó khăn. Trong khi các nước đang tăng tốc triển khai IPv6, đặc biệt là ở khu vực Châu Á - Thái Bình Dương và một số nước Châu Âu, thì ở Châu Mỹ và Châu Phi tương đối chậm trong quá trình này. Mỗi máy tính cần ít nhất một địa chỉ IP để có thể truy cập Internet; Địa chỉ IP hiện nay đang sử dụng thuộc thế hệ 4 (IPv4) sử dụng 32 bit để mã hóa địa chỉ. Theo lý thuyết thì IPv4 chứa hơn 4 tỷ địa chỉ và có thể cấp phát hết trong năm 2011. Chính điều này thúc đẩy sự ra đời một thế hệ địa chỉ Internet mới IPv6.

IPv6 được thiết kế với hi vọng khắc phục những hạn chế vốn có của địa chỉ IPv4 như hạn chế về không gian địa chỉ, cấu trúc định tuyến và bảo mật đồng thời đem lại những đặc tính mới thỏa mãn các nhu cầu dịch vụ của thế hệ mạng mới như khả năng tự động cấu hình mà không cần hỗ trợ của máy chủ DHCP, cấu trúc định tuyến tốt hơn, hỗ trợ Multicast, hỗ trợ bảo mật và di động tốt hơn.

Hiện IPv6 đang được chuẩn hóa từng bước và đưa vào sử dụng thực tế tuy nhiên quá trình chuyển đổi hệ thống mạng từ IPv4 sang IPv6 còn gặp

nhiều vấn đề từ thiết bị không đồng bộ, các nhà cung cấp dịch vụ Internet, kiến thức người sử dụng và quản lý mạng.

- *Giao thức TCP*: Là một trong các giao thức cốt lõi của bộ giao thức TCP/IP. Sử dụng TCP, các ứng dụng trên các máy chủ được nối mạng có thể tạo các "kết nối" với nhau, mà qua đó chúng có thể trao đổi dữ liệu hoặc các gói tin. Giao thức này đảm bảo chuyển giao dữ liệu tới nơi nhận một cách đáng tin cậy và đúng thứ tự. TCP còn phân biệt giữa dữ liệu của nhiều ứng dụng (chẳng hạn, dịch vụ Web và dịch vụ thư điện tử) đồng thời chạy trên cùng một máy chủ.

TCP hỗ trợ nhiều giao thức ứng dụng phổ biến nhất trên Internet và các ứng dụng kết quả, trong đó có WWW, thư điện tử và Secure Shell.

Trong bộ giao thức TCP/IP, TCP là tầng trung gian giữa giao thức IP bên dưới và một ứng dụng bên trên. Các ứng dụng thường cần các kết nối đáng tin cậy kiểu đường ống để liên lạc với nhau, trong khi đó, giao thức IP không cung cấp những dòng kiểu đó, mà chỉ cung cấp dịch vụ chuyển gói tin không đáng tin cậy. TCP làm nhiệm vụ của tầng giao vận trong mô hình OSI đơn giản của các mạng máy tính.

- *Giao thức UDP*: Là một trong những giao thức cốt lõi của giao thức TCP/IP. Dùng UDP, chương trình trên mạng máy tính có thể gửi những dữ liệu ngắn được gọi là datagram tới máy khác. UDP không cung cấp sự tin cậy và thứ tự truyền nhận mà TCP làm; các gói dữ liệu có thể đến không đúng thứ tự hoặc bị mất mà không có thông báo. Tuy nhiên UDP nhanh và hiệu quả hơn đối với các mục tiêu như kích thước nhỏ và yêu cầu khắt khe về thời gian. Do bản chất không trạng thái của nó nên nó hữu dụng đối với việc trả lời các truy vấn nhỏ với số lượng lớn người yêu cầu.

Những ứng dụng phổ biến sử dụng UDP như DNS (Domain Name System), ứng dụng streaming media, Voice over IP, Trivial File Transfer

Protocol (TFTP), và game trực tuyến.

- *Giao thức FTP* (File Transfer Protocol - Giao thức truyền tập tin) thường được dùng để trao đổi tập tin qua mạng lưới truyền thông dùng giao thức TCP/IP (chẳng hạn như Internet - mạng ngoại bộ - hoặc intranet - mạng nội bộ). Hoạt động của FTP cần có hai máy tính, một máy chủ và một máy khách). Máy chủ FTP, dùng chạy phần mềm cung cấp dịch vụ FTP, gọi là trình chủ, lắng nghe yêu cầu về dịch vụ của các máy tính khác trên mạng lưới. Máy khách chạy phần mềm FTP dành cho người sử dụng dịch vụ, gọi là trình khách, thì khởi đầu một liên kết với máy chủ. Một khi hai máy đã liên kết với nhau, máy khách có thể xử lý một số thao tác về tập tin, như tải tập tin lên máy chủ, tải tập tin từ máy chủ xuống máy của mình, đổi tên của tập tin, hoặc xóa tập tin ở máy chủ v.v. Vì giao thức FTP là một giao thức chuẩn công khai, cho nên bất cứ một công ty phần mềm nào, hay một lập trình viên nào cũng có thể viết trình chủ FTP hoặc trình khách FTP. Hầu như bất cứ một nền tảng hệ điều hành máy tính nào cũng hỗ trợ giao thức FTP. Điều này cho phép tất cả các máy tính kết nối với một mạng lưới có nền TCP/IP, xử lý tập tin trên một máy tính khác trên cùng một mạng lưới với mình, bất kể máy tính ấy dùng hệ điều hành nào (nếu các máy tính ấy đều cho phép sự truy cập của các máy tính khác, dùng giao thức FTP). Hiện nay trên thị trường có rất nhiều các trình khách và trình chủ FTP, và phân đông các trình ứng dụng này cho phép người dùng được lấy tự do, không mất tiền.

- *Giao thức SMTP* (Simple Mail Transfer Protocol - giao thức truyền tải thư tin đơn giản) là một chuẩn truyền tải thư điện tử qua mạng Internet. SMTP dùng cổng 25 của giao thức TCP. Để xác định trình chủ SMTP của một tên miền nào đấy (domain name), người ta dùng một mẫu tin MX (Mail eXchange - Trao đổi thư) của DNS (Domain Name System - Hệ thống tên miền).

SMTP định nghĩa tất cả những gì đã làm với email. Nó xác định cấu trúc của các địa chỉ, yêu cầu tên miền và bất cứ điều gì liên quan đến email. SMTP cũng xác định các yêu cầu cho Post Office Protocol (POP) và truy cập Internet Message Protocol (IMAP) máy chủ, do đó email được gửi đúng cách.

- *Giao thức HTTP* (HyperText Transfer Protocol - Giao thức truyền tải siêu văn bản) là một trong năm giao thức chuẩn về mạng Internet, được dùng để liên hệ thông tin giữa Máy cung cấp dịch vụ (Web server) và Máy sử dụng dịch vụ (Web client) là giao thức Client/Server dùng cho World Wide Web-WWW, HTTP là một giao thức ứng dụng của bộ giao thức TCP/IP (các giao thức nền tảng cho Internet).

- *Giao thức HTTPS* (Hypertext Transfer Protocol Secure) là một sự kết hợp giữa giao thức HTTP và giao thức bảo mật SSL hay TLS cho phép trao đổi thông tin một cách bảo mật trên Internet. Giao thức HTTPS thường được dùng trong các giao dịch nhạy cảm cần tính bảo mật cao.

- *Giao thức TELNET* (TErminaL NETwork) là một giao thức mạng được dùng trên các kết nối với Internet hoặc các kết nối tại mạng máy tính cục bộ LAN. TELNET thường được dùng để cung cấp những phiên giao dịch đăng nhập, giữa các máy trên mạng Internet, dùng dòng lệnh có tính định hướng người dùng. Tên của nó có nguồn gốc từ hai chữ tiếng Anh "telephone network" (mạng điện thoại), vì chương trình phần mềm được thiết kế, tạo cảm giác như một thiết bị cuối được gắn vào một máy tính khác.

- *Giao thức SSH* (Secure Shell) là một giao thức mạng dùng để thiết lập kết nối mạng một cách bảo mật. SSH hoạt động ở lớp trên trong mô hình phân lớp TCP/IP. Các công cụ SSH (như là OpenSSH, ...) cung cấp cho người dùng cách thức để thiết lập kết nối mạng được mã hoá để tạo một kênh kết nối riêng tư. Hơn nữa tính năng tunneling của các công cụ này cho phép chuyển tải các giao vận theo các giao thức khác.

- *Giao thức ICMP* (Internet Control Message Protocol) cho phép việc thử nghiệm và khắc phục các sự cố của giao thức TCP/IP. ICMP định nghĩa các thông điệp được dùng để xác định khi nào một hệ thống mạng có thể phân phối các gói tin. Thật ra, ICMP là một thành phần bắt buộc của mọi hiện thực IP. Trong một vài trường hợp, một gateway hoặc một máy đích sẽ cần giao tiếp với máy nguồn để báo cáo lại các lỗi xảy ra trong quá trình xử lý gói tin. Trong trường hợp đó, ICMP sẽ được dùng. ICMP sử dụng IP như thể nó nằm ở một mức cao hơn

1.2.3. Các kỹ thuật tấn công mạng máy tính

- Nghe trộm (Eavesdropping)

Nhìn chung, phần lớn các thông tin liên lạc mạng diễn ra ở dạng rõ (cleartext) - định dạng không bảo đảm an toàn, cho phép kẻ tấn công có thể can thiệp vào dữ liệu trên mạng như nghe lén, chỉnh sửa nội dung thông tin... Nếu không có các dịch vụ mã hóa mạnh mẽ dựa trên mật mã, dữ liệu trên mạng có thể bị đọc bởi những kẻ có ý đồ xấu và gây ra tổn thất lớn cho cá nhân cũng như các doanh nghiệp.

Việc nghe trộm thông tin trên đường truyền có thể được thực hiện bằng việc cài keylog, phần mềm chặn bắt gói tin, phân tích giao thức hay thậm chí là các thiết bị phần cứng hỗ trợ việc “lắng nghe” các thông tin liên lạc trên mạng.

- Giả mạo (Spoofing)

Hầu hết các mạng và hệ điều hành sử dụng địa chỉ IP để xác nhận một đối tượng là hợp lệ. Trong một số trường hợp, một địa chỉ IP có thể bị giả mạo, kẻ tấn công cũng có thể sử dụng những chương trình đặc biệt để xây dựng các gói tin IP có vẻ như xuất phát từ những địa chỉ hợp lệ thuộc mạng nội bộ của một công ty. Sau khi đoạt được quyền truy cập vào mạng bằng IP

hợp lệ, kẻ tấn công có thể thực hiện các ý đồ xấu như sửa đổi, định tuyến lại hay xóa dữ liệu hệ thống.

- *Tấn công từ chối dịch vụ (Denial of Service)*

Đây là dạng tấn công trong đó kẻ tấn công làm cho tài nguyên của bộ nhớ trở nên quá tải không thể xử lý các yêu cầu hợp lệ hoặc từ chối người dùng hợp pháp truy cập vào máy tính hay mạng máy tính.

Các loại tấn công từ chối dịch vụ phổ biến:

Tear drop: Tất cả các dữ liệu chuyển đi trên mạng từ hệ thống nguồn đến hệ thống đích đều phải trải qua 2 quá trình: dữ liệu sẽ được chia ra thành các mảnh nhỏ ở hệ thống nguồn, mỗi mảnh đều phải có một giá trị offset định để xác định vị trí của mảnh đó trong gói dữ liệu được chuyển đi. Khi các mảnh này đến hệ thống đích, hệ thống đích sẽ dựa vào giá trị offset để sắp xếp các mảnh lại với nhau theo thứ tự đúng như ban đầu. Lợi dụng sơ hở đó, ta chỉ cần gửi đến hệ thống đích một loạt gói packets với giá trị offset chồng chéo lên nhau. Hệ thống đích sẽ không thể nào sắp xếp lại các packets này, nó không điều khiển được và có thể bị crash, reboot hoặc ngừng hoạt động nếu số lượng gói packets với giá trị offset chồng chéo lên nhau quá lớn.

SYN Attack: Trong SYN Attack, hacker sẽ gửi đến hệ thống đích một loạt SYN packets với địa chỉ ip nguồn không có thực. Hệ thống đích khi nhận được các SYN packets này sẽ gửi trở lại các địa chỉ không có thực đó và chờ đợi để nhận thông tin phản hồi từ các địa chỉ ip giả. Vì đây là các địa chỉ IP không có thực, nên hệ thống đích sẽ chờ đợi vô ích và còn đưa các “request” chờ đợi này vào bộ nhớ, gây lãng phí một lượng đáng kể bộ nhớ trên máy chủ mà đúng ra là phải dùng vào việc khác thay cho phải chờ đợi thông tin phản hồi không có thực này. Nếu ta gửi cùng một lúc nhiều gói tin có địa chỉ IP giả như vậy thì hệ thống sẽ bị quá tải dẫn đến bị crash hoặc boot máy tính[1].

Smurf Attack: Trong Smurf Attack, hacker sẽ gửi các gói tin ICMP đến địa chỉ broadcast của mạng khuếch đại. Điều đặc biệt là các gói tin ICMP packets này có địa chỉ ip nguồn chính là địa chỉ IP của nạn nhân. Khi các packets đó đến được địa chỉ broadcast của mạng khuếch đại, các máy tính trong mạng khuếch đại sẽ tưởng rằng máy tính nạn nhân đã gửi gói tin ICMP packets đến và chúng sẽ đồng loạt gửi trả lại hệ thống nạn nhân các gói tin phản hồi ICMP packets. Hệ thống máy nạn nhân sẽ không chịu nổi một khối lượng khổng lồ các gói tin này và nhanh chóng bị ngừng hoạt động, crash hoặc reboot.

UDP Flooding: Cách tấn công UDP đòi hỏi phải có 2 hệ thống máy cùng tham gia. Hackers sẽ làm cho hệ thống của mình đi vào một vòng lặp trao đổi các dữ liệu qua giao thức UDP. Và giả mạo địa chỉ IP của các gói tin là địa chỉ loopback (127.0.0.1) , rồi gửi gói tin này đến hệ thống của nạn nhân trên cổng UDP echo (7). Hệ thống của nạn nhân sẽ trả lời lại các messages do 127.0.0.1 (chính nó) gửi đến, kết quả là nó sẽ đi vòng một vòng lặp vô tận. Tuy nhiên, có nhiều hệ thống không cho dùng địa chỉ loopback nên hacker sẽ giả mạo một địa chỉ IP của một máy tính nào đó trên mạng nạn nhân và tiến hành ngập lụt UDP trên hệ thống của nạn nhân.

Tấn công DNS: Hacker có thể đổi một lối vào trên Domain Name Server của hệ thống nạn nhân rồi cho chỉ đến một website nào đó của hacker. Khi máy khách yêu cầu DNS phân tích địa chỉ bị xâm nhập thành địa chỉ IP, lập tức DNS (đã bị hacker thay đổi cache tạm thời) sẽ đổi thành địa chỉ IP mà hacker đã cho chỉ đến đó. Kết quả là thay vì phải vào trang Web muốn vào thì các nạn nhân sẽ vào trang Web do chính hacker tạo ra. Một cách tấn công từ chối dịch vụ thật hữu hiệu[1].

Distributed DoS Attacks (DDoS): DDoS yêu cầu phải có ít vài hackers cùng tham gia. Đầu tiên các hackers sẽ cố thâm nhập vào các mạng máy tính

được bảo mật kém, sau đó cài lên các hệ thống này chương trình DDoS server. Bây giờ các hackers sẽ hẹn nhau đến thời gian đã định sẽ dùng DDoS client kết nối đến các DDoS servers, sau đó đồng loạt ra lệnh cho các DDoS servers này tiến hành tấn công DDoS đến hệ thống nạn nhân.

DRDoS (The Distributed Reflection Denial of Service Attack): Đây có lẽ là kiểu tấn công lợi hại và làm boot máy tính của đối phương nhanh gọn . Cách làm thì cũng tương tự như DDos nhưng thay vì tấn công bằng nhiều máy tính thì người tấn công chỉ cần dùng một máy tấn công thông qua các server lớn trên thế giới. Vẫn với phương pháp giả mạo địa chỉ IP của victim , kẻ tấn công sẽ gửi các gói tin đến các server mạnh, nhanh và có đường truyền rộng như Yahoo .v.v... , các server này sẽ phản hồi các gói tin đó đến địa chỉ của victim. Việc cùng một lúc nhận được nhiều gói tin thông qua các server lớn này sẽ nhanh chóng làm nghẽn đường truyền của máy tính nạn nhân và làm crash, reboot máy tính đó. Cách tấn công này lợi hại ở chỗ chỉ cần một máy có kết nối Internet đơn giản với đường truyền bình thường cũng có thể đánh bật được hệ thống có đường truyền tốt thế giới nếu như ta không kịp ngăn chặn.

- *Tấn công kẻ đứng giữa (MITM - Man-in-the-middle)*

Man-in-the-Middle (MITM) là hình thức tấn công mà kẻ tấn công nằm vùng trên đường truyền với vai trò là máy trung gian trong việc trao đổi thông tin giữa hai máy tính, hai thiết bị, hay giữa một máy tính và server, nhằm nghe trộm, thông dịch dữ liệu nhạy cảm, đánh cắp thông tin hoặc thay đổi luồng dữ liệu trao đổi giữa các nạn nhân.

Hiện nay có các hình thức tấn công MITM phổ biến như:

- Tấn công giả mạo ARP cache (ARP Cache Poisoning).
- Tấn công giả mạo DNS (DNS Spoofing hay DNS Cache Poisoning).
- Chiếm quyền điều khiển Session (Session Hijacking).

- Chiếm quyền điều khiển SSL.

- *Tấn công chặn bắt (Sniffer)*

Sniffer là một ứng dụng hoặc một thiết bị có thể đọc, theo dõi và chặn bắt dữ liệu trao đổi và các gói tin trên mạng. Nếu các gói tin không được mã hóa, sniffer sẽ cung cấp một cái nhìn đầy đủ về các dữ liệu bên trong gói tin. Thậm chí các gói tin đã được đóng gói cũng có thể bị phá vỡ và đọc trừ khi chúng được mã hóa và kẻ tấn công không khai thác được khóa giải mã. Bằng cách sử dụng Sniffer, kẻ tấn công có thể:

- Phân tích mạng của đối phương và thu thập thông tin nhằm khiến cho hệ thống bị trì trệ hoặc dính lỗi.

- Đọc các thông tin liên lạc

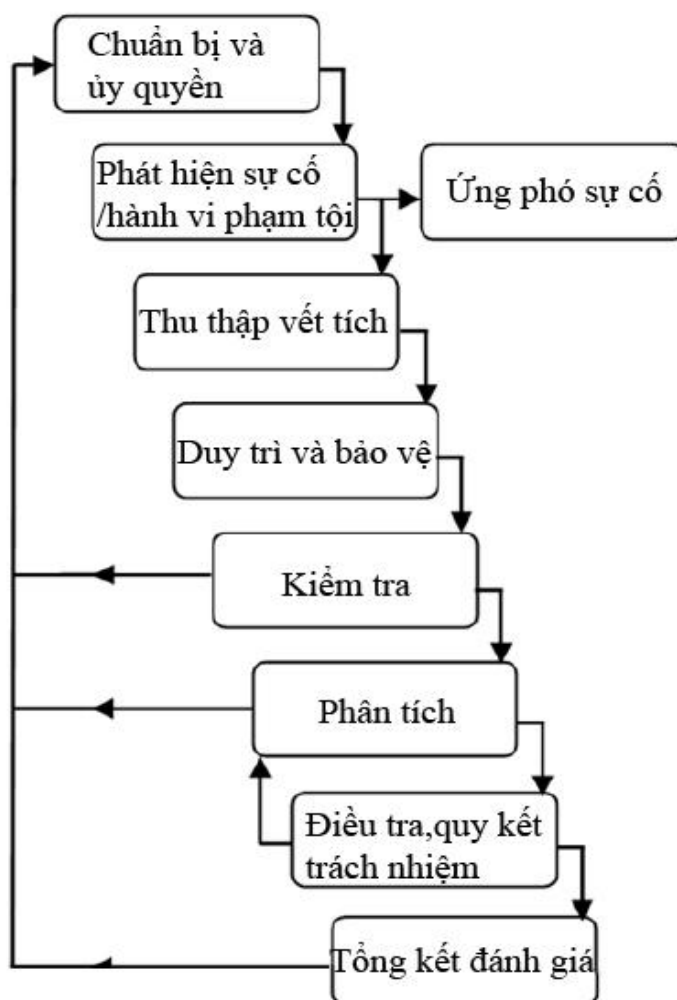
- *Tấn công lớp ứng dụng (Application-layer)*

Mục tiêu của một cuộc tấn công lên lớp ứng dụng là các máy chủ ứng dụng, nó được thực hiện bằng cách cố tình gây ra lỗi trong hệ điều hành của máy chủ hoặc các ứng dụng chạy trên máy chủ. Điều này sẽ dẫn đến việc kẻ tấn công có khả năng vượt qua các kiểm soát truy cập bình thường. Những kẻ tấn công lợi dụng kẽ hở này để giành quyền kiểm soát các ứng dụng lẫn hệ thống mạng và có thể thực hiện các hành vi sau:

- Đọc, thêm, xóa, sửa dữ liệu hoặc hệ điều hành
- Cài đặt và lây nhiễm các chương trình virus lên hệ thống
- Kết hợp cài đặt các chương trình chặn bắt, nghe lén thông tin trên mạng
- Can thiệp đến quá trình hoạt động của ứng dụng hoặc hệ điều hành như ngắt kết nối, tắt máy...
- Vô hiệu hóa các kiểm soát an toàn để thực hiện các cuộc tấn công trong tương lai[1].

CHƯƠNG 2. PHÂN TÍCH ĐIỀU TRA MẠNG VÀ PHÂN TÍCH GÓI TIN TRONG ĐIỀU TRA MẠNG

2.1. Quy trình tổng quan trong phân tích điều tra mạng



Hình 2.1. Quy trình chung trong phân tích điều tra mạng

Phần này trình bày quy trình chung cho việc phân tích điều tra mạng nhằm xác định một cách cụ thể các bước thực hiện từ những mô hình đã được đề xuất cho điều tra số.

2.1.1. Giai đoạn 1: Chuẩn bị và ủy quyền

Network Forensics chỉ áp dụng cho các môi trường mà ở đó những công cụ an ninh mạng như hệ thống phát hiện xâm nhập IDS, hệ thống phân tích gói tin, tường lửa, phần mềm đo đặc lưu lượng,... được triển khai tại những

điểm chiến lược trên mạng. Đội ngũ quản lý những công cụ này phải được đào tạo để đảm bảo có thể thu thập số bằng chứng tối đa và chất lượng nhất nhằm tạo điều kiện thuận lợi cho việc quy kết hành vi phạm tội. Ngoài ra việc giám sát lưu lượng mạng còn có những chính sách an toàn được thiết lập nhằm hạn chế sự vi phạm đến yếu tố riêng tư của các cá nhân và tổ chức. Honeynets và network telescope cũng có thể được triển khai để thu hút kẻ tấn công, nghiên cứu các hành vi và tìm hiểu chiến thuật của chúng.

2.1.2. Giai đoạn 2: Phát hiện sự cố hoặc hành vi phạm tội

Những cảnh báo được tạo ra bởi các công cụ bảo mật khác nhau chỉ ra các vi phạm về an ninh hay chính sách sẽ được theo dõi. Bất kỳ một sự việc trái phép hay hành động dị thường bị phát hiện sẽ được phân tích. Sự hiện diện và tính chất của cuộc tấn công được xác định dựa vào các thông số khác nhau. Một sự xác minh nhanh chóng được thực hiện để đánh giá và xác nhận tấn công khả nghi. Điều này tạo điều kiện thuận lợi cho việc quyết định quan trọng liệu có tiếp tục điều tra hay bỏ qua các cảnh báo như là báo động sai. Cần thực hiện các biện pháp phòng ngừa để chứng cứ không bị sửa đổi trong quá trình này. Việc xác nhận vụ việc chia ra làm hai hướng - ứng phó sự cố và thu thập dữ liệu.

2.1.3. Giai đoạn 3: Ứng phó sự cố

Việc đối phó với tội phạm hay các xâm nhập đã phát hiện bắt đầu dựa trên thông tin được thu thập nhằm xác nhận và đánh giá vụ việc. Các phản ứng ban đầu phụ thuộc vào các loại hình tấn công được xác định và hướng dẫn bởi chính sách tổ chức, pháp luật và thương mại. Một kế hoạch hành động về việc làm thế nào để ngăn chặn các cuộc tấn công trong tương lai và phục hồi từ các tổn thất tồn tại ban đầu. Đồng thời, quyết định liệu có tiếp tục điều tra và thu thập thêm thông tin hay không. Giai đoạn này được áp dụng đối với những trường hợp mà cuộc điều tra được khởi tạo trong khi tấn công đang thực hiện và không có thông báo phạm tội.

2.1.4. Giai đoạn 4: Thu thập các vết tích mạng

Dữ liệu thu được từ các bộ cảm biến (*sensor*) được sử dụng để thu thập lưu lượng mạng. Các cảm biến sử dụng phải an toàn, có khả năng chịu lỗi, giới hạn quyền truy cập và phải có khả năng tránh sự thỏa hiệp. Một thủ tục được xác định rõ bằng cách sử dụng những công cụ tin cậy, phần cứng và phần mềm, phải được dùng để thu thập chứng cứ tối đa nhưng lại gây ra tác động tối thiểu đến nạn nhân. Mạng phải được giám sát để xác định các tấn công trong tương lai. Tính toàn vẹn của dữ liệu được ghi lại và các bản ghi sự kiện mạng phải được đảm bảo. Việc thu thập là khó khăn nhất đối với dữ liệu của lưu lượng mạng thay đổi một cách nhanh chóng và nó không có khả năng tạo ra cùng các dấu vết ở những lần sau. Số lượng dữ liệu được ghi lại sẽ rất lớn, yêu cầu một không gian bộ nhớ tương đương và hệ thống phải có khả năng để xử lý các định dạng khác nhau một cách thích hợp.

2.1.5. Giai đoạn 5: Duy trì và bảo vệ

Các dữ liệu ban đầu lấy từ các dấu vết (*traces*) hay các bản ghi (*records*) sẽ được lưu trữ trên một thiết bị sao lưu. Việc băm giá trị của dữ liệu thu được sẽ đảm bảo an toàn cho dữ liệu, nó đảm bảo tính chính xác và độ tin cậy của dữ liệu được bảo quản. Chuỗi giám sát được thực hiện chính xác để không xảy ra việc sử dụng trái phép hay giả mạo. Một bản sao lưu khác của dữ liệu sẽ được sử dụng cho phân tích và lưu lượng mạng ban đầu thu được sẽ được bảo quản. Việc này được thực hiện để quá trình điều tra có thể chứng minh một lần nữa trên dữ liệu gốc được bảo quản để đáp ứng các yêu cầu pháp lý.

2.1.6. Giai đoạn 6: Kiểm tra

Các dấu vết thu được từ các cảm biến an toàn khác nhau được tích hợp và kết hợp để tạo ra một tập dữ liệu lớn mà việc phân tích có thể thực hiện được. Việc sắp xếp và dán nhãn thời gian cũng được thực hiện đồng thời. Điều này nhằm đảm bảo thông tin quan trọng không bị mất hoặc lẫn lộn. Dữ liệu ẩn hoặc nguy trang của kẻ tấn công cần phải được phục hồi. Dữ liệu thu

thập được phân loại và nhóm thành các nhóm để dễ dàng trong khâu quản lý. Thông tin dự phòng và dữ liệu không liên quan bị loại bỏ còn các thuộc tính đại diện tối thiểu được xác định để hạn chế lượng thông tin nhằm phân tích các bằng chứng có khả năng nhất.

2.1.7. Giai đoạn 7: Phân tích

Chứng cứ sau khi thu thập được tìm kiếm cách thức phù hợp để khai thác dấu hiệu đặc biệt của tội phạm. Các dấu hiệu này được phân loại và bằng suy luận tương quan để đưa ra những nhận xét quan trọng thông qua các mẫu tấn công đã có. Tiếp cận bằng phương pháp thống kê và khai phá dữ liệu được dùng để tìm kiếm dữ liệu và kết hợp với mẫu tấn công phù hợp. Một vài thông số quan trọng có liên quan đến sự thiết lập các kết nối mạng, truy vấn DNS, phân mảnh gói tin, kỹ thuật in dấu giao thức và hệ điều hành, các tiến trình giả mạo, phần mềm hay rootkit được cài đặt. Các mẫu tấn công được xâu chuỗi với nhau và tấn công sẽ được xây dựng và thực hiện lại nhằm nắm được ý định và phương thức hành động của kẻ tấn công. Các kết quả của giai đoạn này là sự xác nhận các hoạt động đáng ngờ.

2.1.8. Giai đoạn 8: Điều tra và quy kết trách nhiệm

Các thông tin có từ dấu vết bằng chứng được dùng để xác định ai? cái gì? ở đâu? khi nào? như thế nào? và tại sao gây ra sự cố. Việc này sẽ giúp cho việc xây dựng lại kịch bản tấn công và quy kết trách nhiệm. Phần khó khăn nhất của việc phân tích pháp y là xác định danh tính kẻ tấn công. Hai cách thức đơn giản của kẻ tấn công để che giấu bản thân là giả mạo IP và tấn công kiểu bàn đạp. Các nhà nghiên cứu đã đề xuất nhiều giải pháp xác định IP để tìm kiếm địa chỉ chính xác của kẻ tấn công đầu tiên nhưng vẫn còn một vấn đề mở. Kẻ tấn công sử dụng bàn đạp tức là các hệ thống đã bị thỏa hiệp để thực hiện tấn công. Chúng có thể bị phát hiện sử dụng phương pháp tiếp cận dựa vào sự tương tự và bất thường trong số liệu thống kê gói tin. Cách tiếp cận của việc điều tra phụ thuộc vào dạng tấn công.

2.1.9. Giai đoạn 9: Tổng kết đánh giá

Kết quả điều tra được trình bày theo ngôn từ dễ hiểu để cán bộ quản lý tổ chức và cán bộ pháp chế thuận lợi trong khi cung cấp các giải thích của những thủ tục tiêu chuẩn khác nhau dùng để đi đến kết luận. Các tài liệu có hệ thống cũng được bao gồm để đáp ứng các yêu cầu. Những kết luận cũng được trình bày sử dụng trực quan để họ có thể dễ dàng nắm bắt. Các dữ liệu thống kê được giải thích với sự hỗ trợ của các kết luận đến. Một báo cáo toàn diện của vụ việc được thực hiện và các biện pháp được khuyến nghị để ngăn ngừa những sự cố tương tự xảy ra trong tương lai. Các kết quả được tài liệu hóa để sử dụng trong việc điều tra tương lai và cải thiện các sản phẩm bảo mật[2],[4].

2.2. Kỹ thuật phân tích điều tra mạng

2.2.1. Phân tích gói tin

Phân tích gói tin thông thường được quy vào việc nghe các gói tin và phân tích giao thức, mô tả quá trình bắt và phiên dịch các dữ liệu sống như là các luồng đang lưu chuyển trong mạng với mục tiêu hiểu rõ hơn điều gì đang diễn ra trên mạng. Phân tích gói tin thường được thực hiện bởi một packet sniffer, một công cụ được sử dụng để bắt dữ liệu thô đang lưu chuyển trên đường dây. Phân tích gói tin có thể giúp chúng ta hiểu cấu tạo mạng, ai đang ở trên mạng, xác định ai hoặc cái gì đang sử dụng băng thông, chỉ ra những thời điểm mà việc sử dụng mạng đạt cao điểm, chỉ ra các khả năng tấn công và các hành vi phá hoại, và tìm ra các ứng dụng không được bảo mật.

Để thực hiện việc bắt các gói tin trên mạng, phải chỉ ra những vị trí tương ứng để đặt “máy nghe” vào hệ thống đường truyền của mạng. Quá trình này đơn giản là đặt “máy nghe” vào đúng vị trí vật lý nào trong một mạng máy tính. Việc nghe các gói tin không đơn giản chỉ là cắm một máy xách tay vào mạng và bắt gói. Thực tế, nhiều khi việc đặt máy nghe vào mạng khó hơn việc phân tích các gói tin. Thách thức của việc này là ở chỗ là có một số

lượng lớn các thiết bị mạng phần cứng được sử dụng để kết nối các thiết bị với nhau. Lý do là vì 3 loại thiết bị chính (hub, switch, router) có nguyên lý hoạt động rất khác nhau. Và điều này đòi hỏi ta phải nắm rõ được cấu trúc vật lý của mạng mà ta đang phân tích.

2.2.2. Phân tích thống kê lưu lượng

Thông lượng của một mạng có thể được đo bằng các công cụ có sẵn trên các nền tảng khác nhau. Lý do để đo thông lượng trong mạng là mọi người thường quan tâm đến dữ liệu tối đa trong mỗi giây của một liên kết thông tin liên lạc hay một truy cập mạng. Một phương pháp điển hình thực hiện việc đo đạc này là chuyển một tập tin lớn từ một hệ thống sang một hệ thống khác và đo thời gian cần thiết để hoàn tất việc chuyển giao hay sao chép tập tin. Thông lượng sau đó được tính bằng cách chia kích thước tập tin theo thời gian để có được kết quả theo megabit, kilobit hay bit trên mỗi giây...

Tuy nhiên, kết quả của một lần tính như vậy sẽ dẫn đến việc thông lượng trên thực tế ít hơn thông lượng dữ liệu tối đa trên lý thuyết, làm người ta tin rằng liên kết thông tin liên lạc của họ là không chính xác. Trên thực tế, có rất nhiều các chi phí chiếm trong thông lượng ngoài các chi phí truyền tải, bao gồm cả độ trễ, kích thước cửa sổ và hạn chế của hệ thống, có nghĩa là các kết quả không phản ánh được thông lượng tối đa đạt được.

Phần mềm kiểm tra băng thông được sử dụng để xác định băng thông tối đa của một mạng hoặc kết nối internet. Nó thường được thực hiện bằng cách cố gắng tải về hoặc tải lên số dữ liệu tối đa trong thời gian ngắn nhất. Vì lý do này, kiểm tra băng thông có thể trì hoãn tốc độ truyền của mạng và gây ra chi phí dữ liệu tăng cao.

Một phương pháp chính xác hơn là sử dụng phần mềm chuyên dụng như Netcps, JDSU QT600, Spirent Test Center, IxChariot, Iperf, Ttcp, netperf hay bwping để đo thông lượng tối đa cho một truy cập mạng.

2.2.3. Phân tích nhật ký, sự kiện

Một tệp tin nhật ký là một bản ghi của các sự kiện xảy ra trong hệ thống hay trong một mạng bất kì. Tệp tin nhật ký bao gồm các mục nhập vào, mỗi mục chứa các thông tin liên quan đến một sự kiện cụ thể đã xảy ra trong hệ thống. Ban đầu, các tệp tin nhật ký được sử dụng chủ yếu cho vấn đề xử lý sự cố nhưng bây giờ nó phục vụ cho rất nhiều chức năng bên trong các tổ chức như tối ưu hóa hệ thống và hiệu năng mạng, cung cấp các dữ liệu hữu ích trong việc điều tra những hoạt động phạm tội.

Các tệp tin nhật ký được phát triển để chứa thêm các thông tin liên quan đến nhiều loại sự kiện khác nhau xảy ra trên mạng hay trong một hệ thống. Trong một tổ chức, các tệp tin nhật ký chứa những bản ghi liên quan đến an ninh máy tính, ví dụ phổ biến về các bản ghi này là bản ghi kiểm toán, theo dõi nỗ lực xác thực người dùng và nhật ký của các thiết bị an toàn ghi lại những cuộc tấn công vào hệ thống.

Việc triển khai rộng rãi các máy chủ, máy trạm, các thiết bị máy tính cùng mạng lưới internet đã làm gia tăng mối đe dọa đối với mạng và hệ thống, số lượng, khối lượng và sự đa dạng của các tệp tin nhật ký làm các bản ghi bảo mật tăng lên rất nhiều. Điều này đã tạo ra sự cần thiết của việc phân tích các tệp tin nhật ký dùng cho những mục đích riêng, đặc biệt là trong điều tra tấn công mạng.

Tệp tin nhật ký có thể chứa nhiều thông tin về các sự kiện xảy ra trong hệ thống, có thể phân loại thành các dạng đặc biệt sau:

- Nhật ký phần mềm bảo mật (*security software logs*) chủ yếu chứa các thông tin liên quan đến an ninh máy tính và các thiết bị an toàn.
- Nhật ký hệ điều hành (*operating system logs*) liên quan đến các sự kiện xảy ra trong quá trình vận hành.

- Nhật ký ứng dụng (*application logs*) chứa nhiều thông tin về dữ liệu của hệ thống.

2.3. Công cụ sử dụng trong phân tích điều tra mạng

2.3.1. Wireshark

WireShark có một bề dày lịch sử, Gerald Combs là người đầu tiên phát triển phần mềm này. Phiên bản đầu tiên được gọi là Ethernal được phát hành năm 1998. Tám năm sau kể từ khi phiên bản đầu tiên ra đời, Combs từ bỏ công việc hiện tại để theo đuổi một cơ hội nghề nghiệp khác. Thật không may, tại thời điểm đó, ông không thể đạt được thỏa thuận với công ty đã thuê ông về việc bán quyền của thương hiệu Ethernal. Thay vào đó, Combs và phần còn lại của đội phát triển đã xây dựng một thương hiệu mới cho sản phẩm “Ethernal” vào năm 2006, dự án tên là WireShark.

WireShark đã phát triển mạnh mẽ và đến nay, nhóm phát triển cho đến nay đã lên tới 500 cộng tác viên. Sản phẩm đã tồn tại dưới cái tên Ethernal không được phát triển thêm.

Lợi ích Wireshark đem lại đã giúp cho nó trở nên phổ biến như hiện nay. Nó có thể đáp ứng nhu cầu của cả các nhà phân tích chuyên nghiệp lẫn nghiệp dư và nó đưa ra nhiều tính năng để thu hút mỗi đối tượng khác nhau[9].

2.3.2. NetworkMiner

NetworkMiner là một công cụ phân tích điều tra mạng (Network Forensics Analysis Tool – NFAT) cho Windows. NetworkMiner có thể được sử dụng như một công cụ chặn bắt gói tin thụ động nhằm nhận biết các hệ điều hành, các phiên làm việc, tên host, các port mở... mà không cần đặt bất cứ luồng dữ liệu nào lên mạng.

NetworkMiner cũng có thể phân tích các tệp tin .pcap trong trường hợp ngoại tuyến và tái tạo các tệp tin truyền tải, cấu trúc thư mục hay chứng chỉ từ tệp tin .pcap. Mục đích của NetworkMiner là thu thập dữ liệu (chẳng hạn như

chúng cứ pháp lý) về các host trên mạng chứ không phải thu thập dữ liệu về lưu lượng truy cập, là quan tâm đến trung tâm máy chủ (nhóm các thông tin trên từng máy) chứ không phải là trung tâm gói tin (thông tin về danh sách các gói tin, khung nhìn...). NetworkMiner cũng rất tiện dụng khi phân tích mã độc như C&C (command & control – ra lệnh và điều khiển) kiểm soát lưu lượng truy cập từ mạng lưới botnet.

2.3.3. Snort

Snort là một hệ thống phát hiện xâm nhập mạng (NIDS) mã nguồn mở miễn phí. NIDS là một kiểu của hệ thống phát hiện xâm nhập (IDS), được sử dụng để giám sát dữ liệu di chuyển trên mạng. Cũng có thể các hệ thống phát hiện xâm nhập Host-based, được cài đặt trên một Host cụ thể và chỉ để phát hiện các sự tấn công nhắm đến Host đó. Mặc dù tất cả các phương pháp phát hiện xâm nhập vẫn còn mới nhưng Snort được đánh giá là hệ thống tốt nhất hiện nay.

Snort chủ yếu là một IDS dựa trên luật, tuy nhiên các Input plug-in cũng tồn tại để phát hiện sự bất thường trong các Header của giao thức. Snort sử dụng các luật được lưu trữ trong các File Text, có thể được chỉnh sửa bởi người quản trị. Các luật thuộc về mỗi loại được lưu trong các File khác nhau. File cấu hình chính của Snort là snort.conf. Snort đọc những luật này vào lúc khởi tạo và xây dựng cấu trúc dữ liệu cung cấp nhằm phân tích các dữ liệu thu được. Tìm ra các dấu hiệu và sử dụng chúng trong các luật là một vấn đề đòi hỏi sự tinh tế, vì càng sử dụng nhiều luật thì năng lực xử lý càng được đòi hỏi để thu thập dữ liệu trong thực tế. Snort có một tập hợp các luật được định nghĩa trước để phát hiện các hành động xâm nhập và chúng ta cũng có thể thêm vào các luật của chính mình. Cũng có thể xóa một vài luật đã được tạo trước để tránh việc báo động sai.

2.3.4. Tcpextract & TCPflow

Tcpextract là công cụ dùng để giải nén các tệp tin từ lưu lượng mạng dựa trên các dấu hiệu, dạng tiêu đề và phụ đề (hay còn gọi là “carving” – chạm khắc), đây là một kỹ thuật khôi phục dữ liệu kiểu cũ. Những công cụ như Foremost sử dụng kỹ thuật này để khôi phục các tệp tin từ bất kỳ luồng dữ liệu nào. Tcpextract đặc biệt sử dụng kỹ thuật này vào việc chặn bắt các tệp tin được truyền qua mạng. Các công cụ khác với chức năng tương tự là driftnet và EtherPEG, 2 công cụ này dùng để theo dõi và giải nén tệp tin hình ảnh trên mạng và thường được sử dụng bởi các nhà quản trị để giám sát các hoạt động trực tuyến của người dùng. Hạn chế lớn của driftnet và EtherPEG là chúng chỉ hỗ trợ ba định dạng tệp tin mà không có cách nào để bổ sung thêm. Các kỹ thuật tìm kiếm chúng sử dụng cũng không có khả năng mở rộng và không thể tìm được ở giới hạn gói tin. Tcpextract có những tính năng nổi bật sau:

- Hỗ trợ 26 định dạng tệp tin phổ biến. Những định dạng mới có thể được thêm bằng việc chỉnh sửa tệp tin cấu hình.
- Có thể sử dụng tệp tin cấu hình của Foremost cho Tcpextract.
- Thuật toán tìm kiếm được tùy chỉnh với phạm vi rộng và tốc độ nhanh.
- Sử dụng libpcap, một thư viện di động phổ biến và ổn định cho mạng lưới thu thập dữ liệu.
- Có thể được dùng đối với một mạng trực tuyến hoặc một tệp tin tcpdump đã được capture.

2.3.5. Foremost

Foremost là một chương trình điều khiển (console) dùng để khôi phục tệp tin dựa vào tiêu đề, phụ đề và các cấu trúc dữ liệu bên trong. Quá trình này thường được gọi là chạm khắc dữ liệu (data carving). Foremost có thể làm việc trên các tệp tin ảnh, chẳng hạn được tạo ra bởi dd, Safeback, Encase,... hoặc trực tiếp từ trên ổ cứng. Tiêu đề và phụ đề có thể được xác định bởi một tệp tin cấu hình hoặc có thể sử dụng một switch dòng lệnh dựa

trên dạng tệp tin tích hợp. Các dạng tích hợp này sẽ tra cứu cấu trúc dữ liệu của định dạng tệp tin được cung cấp được nhằm đảm bảo việc phục hồi sẽ nhanh và đáng tin cậy hơn.

2.3.6. Scapy

Scapy là một công cụ thao tác với gói tin dùng cho mạng máy tính, được viết bằng Python bởi Philippe Biondi. Nó có thể giả mạo hoặc giải mã các gói tin, gửi lại trên đường truyền, chặn bắt và làm khớp các yêu cầu với phản hồi. Nó cũng có thể xử lý những tác vụ khác như quét, truy vết, thăm dò, kiểm thử đơn vị, tấn công và phát hiện mạng.

Scapy cung cấp một giao diện Python vào libpcap (WinPCap trên Windows) theo một cách tương tự như cung cấp trong Wireshark, với giao diện capture trực quan. Nó cũng có thể giao tiếp với một số chương trình khác để cung cấp tính trực quan kể cả Wireshark nhằm giải mã các gói tin, GnuPlot cho việc tạo đồ thị, graphviz hoặc Vpython cho việc hiển thị...

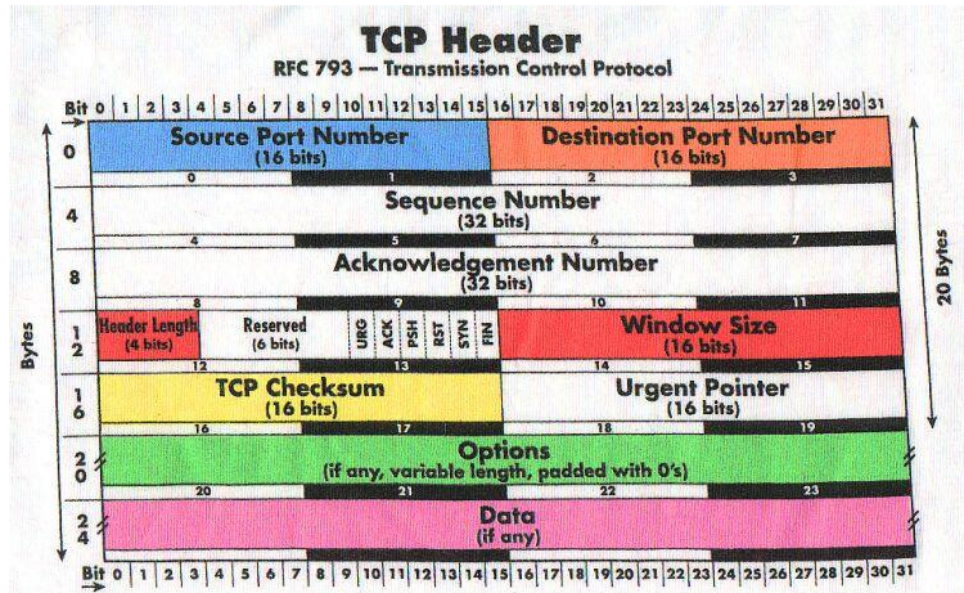
2.4. Cách thức phân tích gói tin trong điều tra mạng

2.4.1. Đặc điểm gói tin mạng

- TCP:

Giao thức điều khiển truyền TCP là một giao thức hoạt động theo phương thức có liên kết (connection – oriented). Trong bộ giao thức TCP/IP, nó là giao thức trung gian giữa IP và một ứng dụng phía trên, đảm bảo dữ liệu được trao đổi một cách tin cậy và đúng thứ tự. Các ứng dụng sẽ gửi các dòng gồm các byte 8 bit tới TCP để gửi qua mạng. TCP sẽ phân chia các dòng này thành các đoạn (segment) có kích thước thích hợp (thường dựa theo kích thước của đơn vị truyền dẫn tối đa MTU của tầng liên kết của mạng mà máy tính đang nằm trong đó. Sau đó TCP chuyển các gói tin thu được tới IP để thực hiện chuyển nó qua liên mạng tới modul TCP tại máy tính đích. Trong quá trình này, nó sẽ có cơ chế bắt tay, điều khiển truyền, đánh số thứ tự và sửa lỗi để việc truyền dẫn diễn ra đúng đắn và chính xác.

Đơn vị dữ liệu của TCP được gọi là segment (đoạn dữ liệu) bao gồm 2 phần: Header và Data, được miêu tả dưới hình sau:



Hình 2.2. Tcp header

Trong đó:

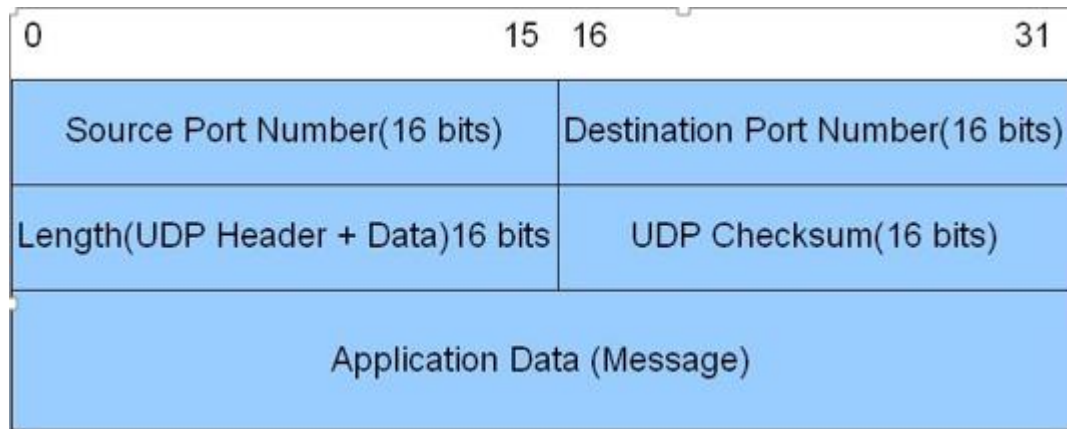
- Source port (16 bit): Số hiệu của cổng của trạm nguồn
- Destination port (16 bit): Số hiệu của cổng của trạm đích.
- Sequence number (32 bit): Trường này có 2 nhiệm vụ. Nếu cờ SYN bật thì nó là số hiệu tuần tự khởi đầu (ISN) và byte dữ liệu đầu tiên là ISN + 1. Nếu không có cờ SYN thì đây là số hiệu byte đầu tiên của segment.
- Acknowledgement number (32 bit): Số hiệu của segment tiếp theo mà trạm nguồn đang chờ để nhận. Ngầm ý báo nhận tốt (các) segment mà trạm đích đã gửi cho trạm nguồn.
- Data offset (4 bit): Qui định độ dài của phần header (tính theo đơn vị từ 32 bit). Phần header có độ dài tối thiểu là 5 từ (160 bit) và tối đa là 15 từ (480 bit).
- Reserved (6 bit): Dành cho tương lai và có giá trị là 0.
- Flags (hay Control bits): Bao gồm 6 cờ từ trái sang phải như sau:

- URG: Cờ cho trường Urgent pointer
 - ACK: Cờ cho trường Acknowledgement
 - PSH: Hàm Push
 - RST: Thiết lập lại đường truyền
 - SYN: Đồng bộ lại số hiệu tuần tự (sequence number).
 - FIN: Không còn dữ liệu từ trạm nguồn.
- Window (16 bit): Số byte trạm nguồn có thể nhận bắt đầu từ giá trị của trường báo nhận (ACK).
- Checksum: 16 bit kiểm tra cho cả phần header và dữ liệu.
- Urgent pointer (16 bit): Trỏ tới số hiệu tuần tự của byte đi theo sau dữ liệu khẩn, cho phép bên nhận biết được độ dài của vùng dữ liệu khẩn. Vùng này chỉ có hiệu lực khi cờ URG được thiết lập.
- Options (độ dài thay đổi): Đây là trường tùy chọn.
- Padding (độ dài thay đổi): Phần chèn thêm vào header để bảo đảm phần header luôn kết thúc ở một mốc 32 bit. Phần thêm này gồm toàn số 0.
- TCP data (độ dài thay đổi): Chứa dữ liệu của tầng trên, có độ dài ngầm định là 536 byte. Giá trị này có thể điều chỉnh bằng cách khai báo trong vùng options.

- UDP

Đây là một giao thức “không liên kết” được sử dụng thay thế trên IP theo yêu cầu của các ứng dụng. Khác với TCP, UDP không có các chức năng thiết lập và giải phóng liên kết. Nó cũng không cung cấp các cơ chế báo nhận, không sắp xếp tuần tự các đơn vị dữ liệu đến và có thể dẫn tới tình trạng dữ liệu mất hoặc trùng mà không hề có thông báo lỗi cho người gửi. Tóm lại nó cung cấp các dịch vụ giao vận không tin cậy như trong TCP. Do ít chức năng phức tạp nên UDP có xu thế hoạt động nhanh hơn so với TCP. Nó thường được dùng cho các ứng dụng không đòi hỏi độ tin cậy cao trong giao vận.

Cấu trúc của một đơn vị dữ liệu UDP như sau:



Hình 2.3. UDP header

Trong đó:

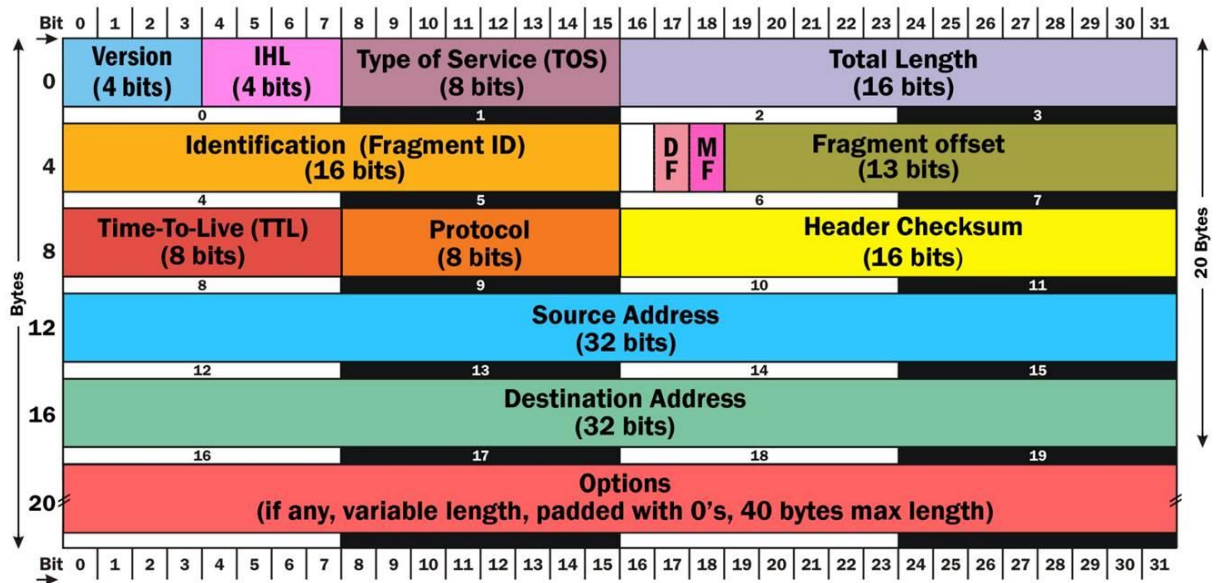
- Source port (16 bit): Trường này xác định cổng của máy gửi đi.
- Destination port (16 bit): Trường xác định cổng của máy nhận thông tin.
- Length (16 bit): Xác định chiều dài của toàn bộ datagram: phần header và dữ liệu. Chiều dài tối thiểu là 8 byte khi gói tin không có dữ liệu, chỉ có header.
- Checksum (16 bit): Trường checksum 16 bit dùng cho việc kiểm tra lỗi của phần header và dữ liệu.

- IP

Giao thức liên mạng IP hạt nhân của bộ giao thức TCP/IP. Trong phạm vi đề tài chúng ta chỉ xét tới IP phiên bản 4 (IPv4). IP là một giao thức hướng dữ liệu được sử dụng trong mạng chuyển mạch gói (ví dụ như Ethernet). IP là một giao thức hoạt động theo phương thức không liên kết (connectionless) và không đảm bảo truyền (không có sự trao đổi thông tin điều khiển). Vai trò của IP tương tự như vai trò của giao thức tầng mạng (network layer) trong mô hình OSI với các chức năng như sau:

- Xác định lược đồ địa chỉ Internet.
- Di chuyển dữ liệu giữa tầng giao vận và tầng liên kết.
- Dẫn đường cho các đơn vị dữ liệu tới các trạm ở xa.

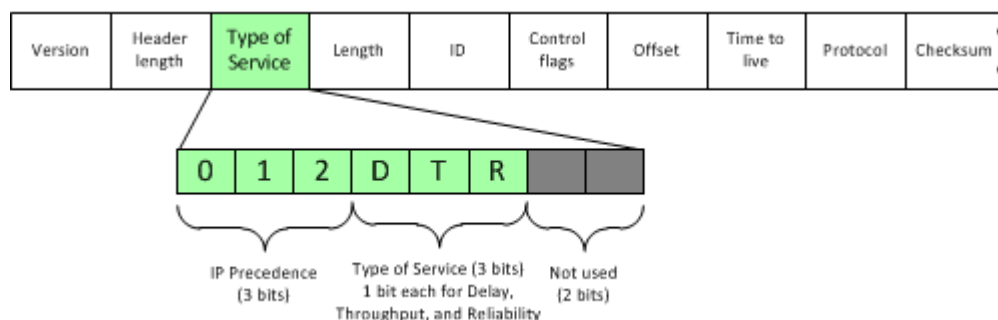
- Thực hiện việc cắt và hợp các đơn vị dữ liệu.



Hình 2.4. IPHeader

Trong đó phần header bao gồm các thành phần:

- Version: chỉ ra phiên bản hiện hành của IP được cài đặt (có giá trị là 4 đối với IPv4).
- Internet Header Length (IHL) Chỉ độ dài phần đầu của IP packet, tính theo đơn vị từ (word = 32 bit). Độ dài tối thiểu là 5 từ (20 byte).
- Differentiated Services (DS): Trước đây còn gọi là Type of Services đặc tả các tham số dịch vụ, có dạng cụ thể như sau:



Hình 2.5. Type of Services

Với ý nghĩa các bit cụ thể:

- Precedence (3 bit): quyền ưu tiên cụ thể là 111 - Network Control, 110 - Internetwork Control, 101 - CRITIC/ECP, 100 - Flash Override, 011 - Flash, 010 - Immediate, 001 - Priority, 000 - Routine.

- D (Delay) (1 bit): chỉ độ trễ yêu cầu $D = 0$ nếu độ trễ bình thường, 1 nếu độ trễ thấp.

- T (Throughput) (1 bit): chỉ thông lượng yêu cầu $T = 0$ thông lượng bình thường, 1 nếu thông lượng cao.

- R (Reliability) (1bit) chỉ độ tin cậy yêu cầu $R = 0$ độ tin cậy bình thường, 1 nếu độ tin cậy cao.

- C (Cost) (1bit) chỉ hao phí $C = 0$ normal cost, 1 nếu minimize cost.

- Reserved (1bit) để dành.

- Total Length trường 16 bit chỉ độ dài toàn bộ datagram bao gồm cả phần header và phần data tính theo byte và có giá trị lớn nhất là 65535 và giá trị nhỏ nhất là 20 byte.

- Identification (16 bit) định danh duy nhất cho 1 datagram khi nó vẫn còn trên liên mạng.

- Flags (3 bit) điều khiển sự phân mảnh. Theo thứ tự từ bit cao xuống bit thấp như sau:

- Reserved: có giá trị 0.

- DF: 0 (May Fragment); 1 (Don't Fragment).

- MF: 0 (Last Fragment); 1 (More Fragment).

- Fragment Offset chỉ vị trí của đoạn (fragment) trong datagram tính theo đơn vị 64 bit, có nghĩa mỗi đoạn (trừ đoạn cuối cùng) phải chứa một vùng dữ liệu có độ dài là bội số của 64 bit.

- Time To Live (TTL) (8 bit): quy định thời gian tồn tại (tính bằng

giây) của datagram trong liên mạng để tránh tình trạng một datagram bị lặp vô hạn trên liên mạng. Thời gian này được cho bởi trạm gửi và được giảm đi (thường quy ước là 1 đơn vị) khi datagram đi qua mỗi router của liên mạng.

- Protocol (8 bit): chỉ ra giao thức tầng trên kế tiếp sẽ nhận vùng dữ liệu ở trạm đích (hiện tại thường là TCP hoặc UDP được cài đặt trên IP).

- Header Checksum (16 bit): mã kiểm soát lỗi 16 bit theo phương pháp CRS, chỉ dành cho phần header.

- Source address (32 bit): địa chỉ trạm nguồn.

- Destination address (16 bit): địa chỉ trạm đích.

- Options (độ dài thay đổi): khai báo các lựa chọn do người dùng yêu cầu (tùy theo từng chương trình).

- Padding (độ dài thay đổi): vùng đệm được dùng để đảm bảo cho phần header luôn kết thúc ở một mốc 32 bits.

- Data (độ dài thay đổi): vùng dữ liệu có độ dài là bội số của 8 bit và tối đa là 65535 byte.

- ICMP

Giao thức ICMP cung cấp cơ chế thông báo lỗi và các tình huống không mong muốn cũng như điều khiển các thông báo trong bộ giao thức TCP/IP. Giao thức này được tạo ra để thông báo các lỗi dẫn đường cho trạm nguồn. ICMP phụ thuộc vào IP để có thể hoạt động và là một phần không thể thiếu của bộ giao thức TCP/IP, tuy nhiên nó không phải giao thức dùng để truyền tải dữ liệu nên thường được coi nằm trong tầng Internet (Internet layer) mà không phải là tầng giao vận (transport layer).

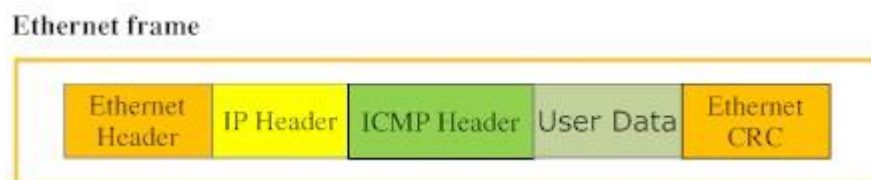
Chức năng của ICMP như sau:

- Cung cấp thông báo phản hồi và trả lời để kiểm tra độ tin cậy của kết nối giữa hai trạm. Điều này được thiết lập bởi câu lệnh PING (Packet internet gropher).

- Định hướng lại lưu lượng để cung cấp việc dẫn đường hiệu quả hơn khi một bộ dẫn đường quá tải do lưu lượng qua nó quá lớn.
- Gửi thông báo về thời gian quá khi datagram của trạm nguồn đã vượt quá TTL và bị loại bỏ.
- Gửi quảng cáo dẫn đường để xác định địa chỉ của các bộ dẫn đường trên đoạn mạng.
- Cung cấp các thông báo quá hạn thời gian.

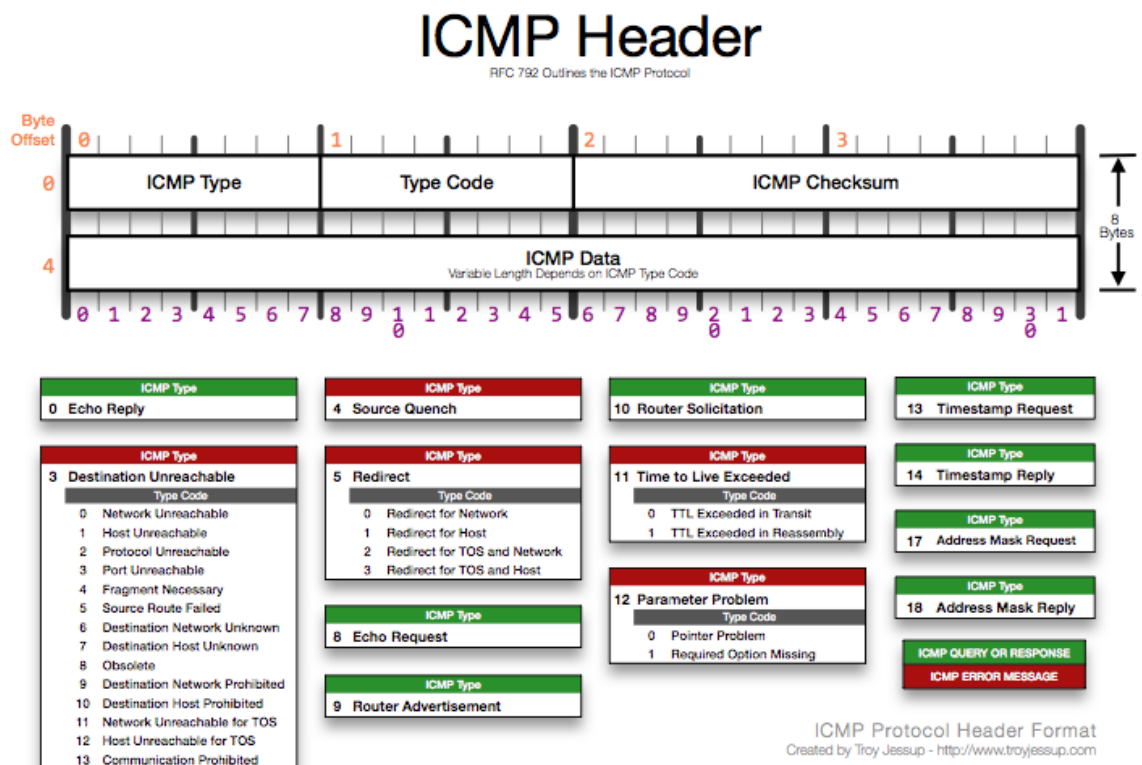
Xác định subnet mask nào được sử dụng trên đoạn mạng.

Dữ liệu của gói ICMP sẽ được đóng gói bởi giao thức IP và Ethernet như trong hình vẽ sau:



Hình 2.6. Vị trí gói ICMP header

Đơn vị dữ liệu của ICMP bao gồm 2 phần: Header và Data. Phần Data trong Window có độ lớn là 32 và theo ngay sau phần Header. Header được bắt đầu sau bit thứ 160 của gói tin IP (trừ khi phần IP Option được sử dụng) có cấu trúc như sau:



Hình 2.7. ICMP header

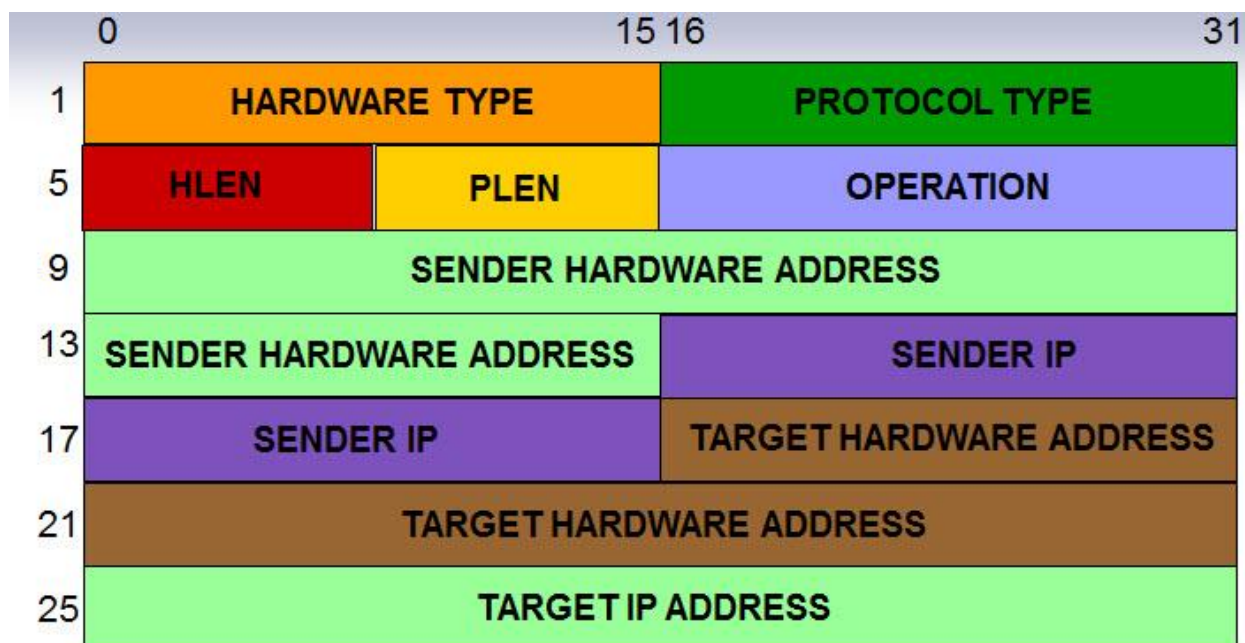
Trong đó:

- Type (8 bit): Cho biết dạng của gói tin, có 8 bit dành cho TYPE ICMP vậy nếu tính tổng có khoảng 255 dạng ICMP nhưng chỉ có 8 dạng hay dùng và cần quan tâm nhất
 - Code (8 bit): Cho biết nhiều thông tin chi tiết về dạng gói tin đó. Một gói tin ICMP được hiểu là "Destination Unreachable" sẽ được thiết lập có code từ 1 tới 15
 - Checksum(16 bit): sử dụng để tính toán tổng cộng gói Header+data của gói ICMP
- ARP

Giao thức phân giải địa chỉ ARP là phương pháp tìm địa chỉ tầng liên kết (hay địa chỉ vật lý) khi biết địa chỉ tầng Internet (IP) hoặc một vài kiểu địa chỉ tầng mạng khác. ARP được sử dụng không chỉ để chuyển đổi địa chỉ đối với IP và Ethernet mà nó được cài đặt để làm việc với nhiều loại địa chỉ của các tầng các loại mạng khác nhau. Tuy nhiên, do sự phổ biến của IPv4 và Ethernet nên ARP chủ yếu được dùng để chuyển đổi từ địa chỉ IP thành địa chỉ MAC. Nó cũng được sử dụng đối với IP dựa trên các công nghệ LAN khác Ethernet như FDDI, Token Ring, IEEE 802.11 hay ATM.

Trong thực tế, khi truyền thông với máy chủ thay vì truy vấn địa chỉ vật lý của máy chủ, giao thức ARP sẽ sử dụng bộ đệm ARP (ARP cache). Bộ đệm lưu trữ các địa chỉ IP gần nhất đã được phân giải. Nếu địa chỉ MAC của địa chỉ IP đích được tìm thấy trong bộ đệm thì địa chỉ này sẽ được sử dụng để truyền thông.

Cấu trúc của một đơn vị dữ liệu giao thức ARP như sau:



Hình 2.8. ARPHeader

Trong đó:

- Hardware type (HTYPE) Mỗi giao thức tầng liên kết (link layer) sẽ

được gán một số để phân biệt (ví dụ như Ethernet là 1)..

- Protocol type (PTYPE) Dùng để phân biệt giao thức tầng Internet, ví dụ như với IP là 0x0800.

- Hardware length (HLEN) Độ dài tính theo byte của địa chỉ vật lý. Đối với Ethernet giá trị này là 6.

- Protocol length (PLEN) Độ dài tính theo byte của địa chỉ logic. Đối với IP giá trị này là 4..

- Operation Xác định hành động mà bên gửi gói tin đang thực hiện: 1 cho request, 2 cho reply, 3 cho RARP request và 4 cho RARP reply.

- Sender hardware address (SHA) Địa chỉ vật lý của trạm gửi.

- Sender protocol address (SPA) Địa chỉ logic của trạm gửi (ví dụ như địa chỉ IP).

- Target hardware address (THA) Địa chỉ vật lý của trạm đích. Trường này được để trống đối với gói tin request.

- Target protocol address (TPA) Địa chỉ logic của trạm đích.

- RARP

Là giao thức ngược lại so với ARP, tìm địa chỉ logic khi biết địa chỉ vật lý. Cấu trúc của một đơn vị dữ liệu của giao thức RARP hoàn toàn tương tự như ARP, ngoại trừ trường Operation. Đối với gói dữ liệu ARP thì Operation có giá trị 1 nếu là request, 2 nếu reply. Đối với gói dữ liệu RARP thì Operation có giá trị 3 nếu là request và 4 nếu là reply.

2.4.2. Cách thức phân tích gói tin mạng

Phân tích gói tin, hay còn thường được gọi là lắng nghe gói tin và phân tích giao thức, mô tả quá trình bắt và diễn giải dữ liệu sống như là các luồng đang lưu chuyển trong mạng với mục tiêu hiểu rõ hơn điều gì đang diễn ra trên mạng. Phân tích gói tin thường được thực hiện bởi 1 packet sniffer, là 1 công cụ được sử dụng để bắt dữ liệu thô đang truyền trên đường dây.

Phân tích gói tin có thể giúp chúng ta:

- Hiểu được các đặc điểm của mạng (cấu tạo của mạng).
- Biết được ai đang ở trên mạng.
- Xác định được ai hay cái gì đang sử dụng băng thông.
- Xác định những thời điểm mà việc sử dụng mạng đạt cao điểm.
- Chỉ ra các khả năng tấn công và các hành vi phá hoại.
- Tìm ra các ứng dụng không được bảo mật.

Quá trình nghe gói tin được chia làm 3 bước:

- Thu thập dữ liệu: Đây là bước đầu tiên, chương trình nghe gói tin chuyển giao diện mạng được lựa chọn sang chế độ Promiscuous. Chế độ này cho phép card mạng có thể nghe tất cả các gói tin đang lưu chuyển trên phân mạng của nó. Chương trình nghe gói sử dụng chế độ này cùng với việc truy nhập ở mức thấp để bắt các dữ liệu nhị phân trên đường truyền.

- Chuyển đổi dữ liệu: Trong bước này, các gói tin nhị phân trên được chuyển đổi thành các khuôn dạng có thể đọc được.

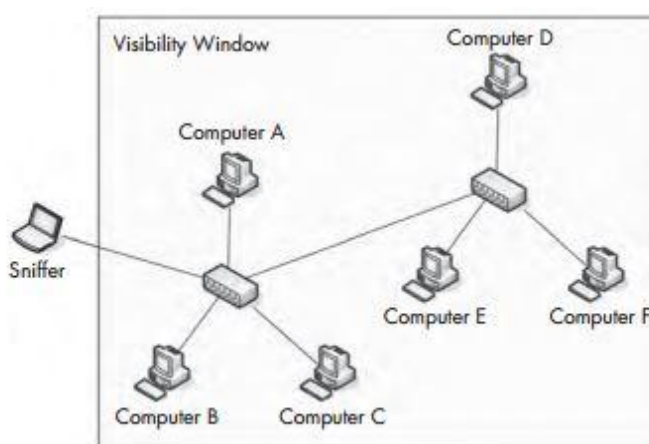
- Phân tích: Phân tích các dữ liệu đã được bắt lại và được chuyển đổi.

Để thực hiện việc bắt các gói tin trên mạng, phải chỉ ra những vị trí tương ứng để đặt “máy nghe” vào hệ thống đường truyền của mạng. Quá trình này đơn giản là đặt “máy nghe” vào đúng vị trí vật lý nào trong một mạng máy tính. Việc nghe các gói tin không đơn giản chỉ là cắm một máy xách tay vào mạng và bắt gói. Thực tế, nhiều khi việc đặt máy nghe vào mạng khó hơn việc phân tích các gói tin. Thách thức của việc này là ở chỗ là có một số lượng lớn các thiết bị mạng phần cứng được sử dụng để kết nối các thiết bị với nhau. Lý do là vì 3 loại thiết bị chính (hub, switch, router) có nguyên lý hoạt động rất khác nhau. Và điều này đòi hỏi ta phải nắm rõ được cấu trúc vật lý của mạng đang phân tích.

Nghiên cứu một số mạng thực tế để chỉ ra cách tốt nhất để bắt các gói tin trong từng môi trường mạng sử dụng Hub, Switch và Router.

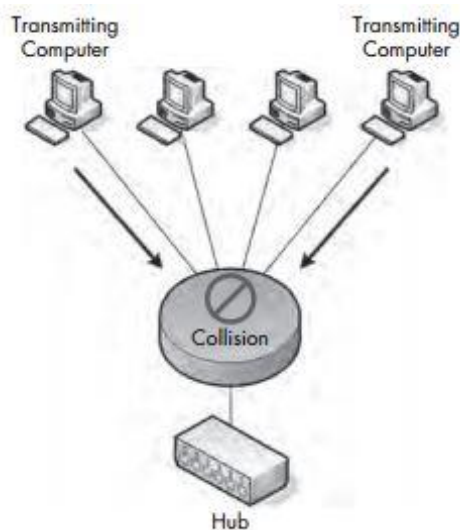
- Nghe trong mạng sử dụng Hub

Lưu lượng được gửi đi qua thiết bị Hub thì sẽ đi qua bất kỳ cổng nào kết nối với Hub. Vì vậy, để phân tích lưu lượng đi qua 1 máy tính kết nối với 1 thiết bị Hub phải kết nối 1 packet sniffer tới 1 cổng còn trống trên Hub. Sẽ thấy được tất cả truyền thông đến và đi từ máy tính đó, cũng như truyền thông giữa các thiết bị khác kết nối với thiết bị Hub đó. Hình dưới đây minh họa việc sniffer trên 1 mạng có Hub (vùng sniffer được giới hạn trong khung):



Hình 2.9. Nghe trong mạng hub

Tuy nhiên, mạng Hub rất hiếm tồn tại bởi vì cấu tạo chỉ có 1 thiết bị duy nhất có thể truyền thông tại 1 thời điểm, 1 thiết bị kết nối qua 1 Hub phải cạnh tranh băng thông với các thiết bị khác cũng đang cố gắng truyền thông qua thiết bị Hub đó. Khi 2 hay nhiều thiết bị truyền thông ngay tại cùng 1 thời điểm, sẽ xảy ra xung đột, như ở hình dưới đây. Kết quả gây ra sẽ là mất mát gói tin, và các thiết bị sẽ phải truyền lại các gói tin đó, khiến cho mạng càng trở nên tắc nghẽn. Khi đến 1 mức xung đột nào đó, thiết bị sẽ phải truyền lại 1 gói tin đến tận 3,4 lần và sẽ làm giảm hiệu năng của mạng. Vì vậy hầu hết các mạng ngày nay đều sử dụng Switch[4].

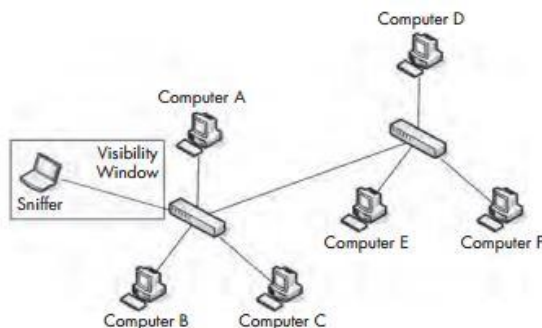


Hình 2.10. Xung đột trong mạng hub

- Nghe trong mạng sử dụng Switch

Switch là loại thiết bị kết nối thông dụng nhất được sử dụng trong môi trường mạng hiện đại. Chúng cung cấp 1 giải pháp hiệu quả để truyền dữ liệu qua các lưu lượng broadcast, unicast và multicast. Ngoài ra, Switch còn cho phép truyền thông song công, có nghĩa là thiết bị có thể gửi và nhận dữ liệu cùng lúc đồng thời.

Tuy nhiên, Switch có 1 sự phức tạp nhất định. Khi kết nối 1 sniffer tới 1 cổng trên Switch, chỉ có thể thấy luồng traffic broadcast và traffic đó được truyền và nhận bởi thiết bị kết nối, được minh họa như hình dưới đây:



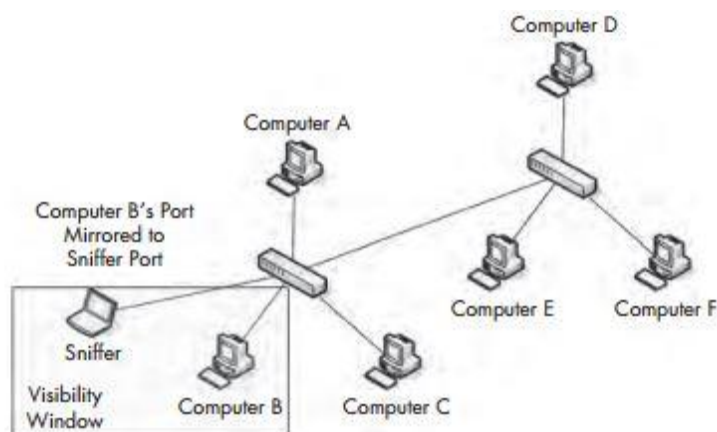
Hình 2.11. Nghe trong mạng Switch

Có 3 cách để bắt luồng traffic từ 1 thiết bị mục tiêu trên 1 mạng chuyển mạch: Port Mirroring, Hubbing Out và ARP Cache Poisoning.

- Port Mirroring

Port Mirroring, hay Port Spanning có lẽ là cách đơn giản nhất để bắt traffic từ 1 thiết bị mục tiêu trên 1 mạng chuyển mạch. Trong loại thiết lập này, phải truy cập vào giao diện dòng lệnh hoặc giao diện quản trị web của Switch kết nối trực tiếp với máy mục tiêu. Switch cũng phải hỗ trợ tính năng Port Mirroring và có 1 cổng còn trống để bạn có thể cắm máy nghe vào.

Khi kích hoạt tính năng Port Mirroring, copy toàn bộ lưu lượng đi qua cổng này sang 1 cổng khác, hay nói cách khác, tính năng Port Mirroring là 1 tính năng cho phép mọi traffic vào và ra 1 cổng sẽ được copy sang 1 cổng đích. Ở trên cổng đích, trên PC gắn vào cổng này, cần cài đặt các chương trình có khả năng đọc và phân tích traffic. Ví dụ, để bắt được lưu lượng trên 1 thiết bị kết nối với cổng 3 của Switch, bạn phải kết nối PC có chương trình phân tích traffic với cổng 4 của Switch đó và ánh xạ các traffic từ cổng 3 sang cổng 4 [4],[6]. Khi đó sẽ thấy được toàn bộ lưu lượng được truyền và nhận bởi máy mục tiêu. Hình dưới đây minh họa điều này:



Hình 2.12. Bắt lưu lượng của thiết bị mục tiêu trên mạng Switch bằng Port Mirroring

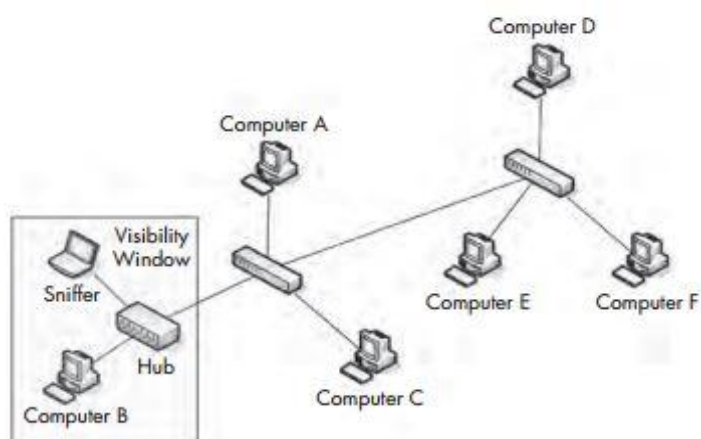
Cách thiết lập tính năng Port Mirroring phụ thuộc vào nhà sản xuất Switch. Với hầu hết Switch, sẽ cần phải đăng nhập vào giao diện dòng lệnh và

sử dụng câu lệnh Port Mirroring. Dưới đây là 1 danh sách các câu lệnh port-mirroring phổ biến.

Manufacturer	Command
Cisco	set span <source port> <destination port>
Enterasys	set port mirroring create <source port> <destination port>
Nortel	port-mirroring mode mirror-port <source port> monitor-port <destination port>

○ Hubbing Out

Một cách đơn giản khác để bắt các lưu lượng của thiết bị mục tiêu trong 1 mạng Switch là Hubbing Out. Hubbing Out là kỹ thuật mà trong đó bạn đặt thiết bị mục tiêu và máy nghe vào cùng 1 phân mạng bằng cách đặt chúng trực tiếp vào 1 thiết bị Hub.



Hình 2.13. Bắt lưu lượng của thiết bị mục tiêu trên mạng Switch bằng Hubbing Out

Rất nhiều người nghĩ rằng Hubbing Out là lừa dối, nhưng nó thật sự là 1 giải pháp hoàn hảo trong các tình huống mà bạn không thể thực hiện Port Mirroring nhưng vẫn có khả năng truy cập vật lý tới Switch mà thiết bị mục tiêu cắm vào.

Trong hầu hết các tình huống, Hubbing Out sẽ giảm tính năng song công của thiết bị mục tiêu (Full to Half). Phương thức này không phải là cách

sạch sẽ nhất để nghe, và nó thường được bạn sử dụng như là 1 lựa chọn khi mà Switch không hỗ trợ Port Mirroring.

Khi Hubbing Out, chắc chắn rằng sử dụng 1 thiết bị Hub chứ không phải là 1 Switch bị gán nhầm nhãn. Khi sử dụng Hub, hãy kiểm tra để chắc chắn rằng nó là 1 thiết bị Hub bằng cách cắm 2 máy tính vào nó và nhìn xem 1 máy có thể nhìn thấy lưu lượng của máy còn lại hay không.

- ARP Cache Poisoning

Tiến trình ARP

Có 2 loại địa chỉ gói tin ở lớp 2 và lớp 3 của mô hình OSI. Các địa chỉ lớp 2, hay còn gọi là địa chỉ MAC, được sử dụng để kết hợp với địa chỉ lớp 3 của hệ thống đang sử dụng. Địa chỉ lớp 3 còn được gọi với cái tên khác là địa chỉ IP [4],[6].

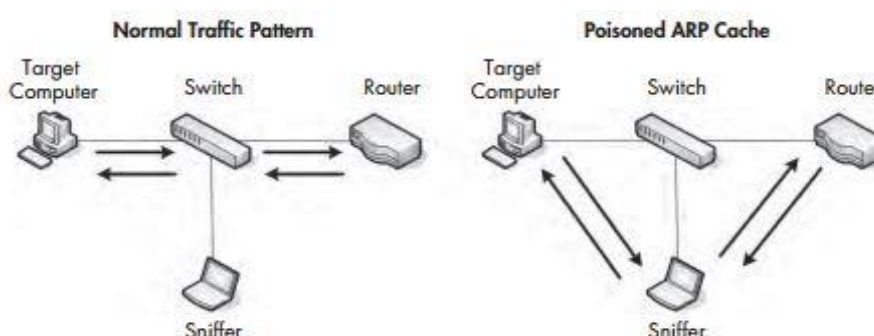
Tất cả các thiết bị trên mạng giao tiếp với thiết bị khác ở lớp 3 sử dụng địa chỉ IP. Bởi vì Switch hoạt động ở lớp 2 của mô hình OSI, nên chúng chỉ có thể hiểu được các địa chỉ MAC, bởi vậy các thiết bị phải chứa cả thông tin này trong những gói tin mà chúng tạo ra. Khi thiết bị không biết địa chỉ MAC, thì chúng sẽ sử dụng địa chỉ IP để có thể chuyển tiếp lưu lượng tới thiết bị thích hợp. Quá trình dịch địa chỉ này được hoàn thành bởi giao thức ARP.

Tiến trình ARP bắt đầu khi 1 máy tính muốn giao tiếp với máy khác. Máy tính truyền tin đó đầu tiên kiểm tra ARP cache của nó để xem liệu nó đã có địa chỉ MAC tương ứng với địa chỉ IP của máy nhận hay chưa. Nếu chưa có, nó sẽ gửi 1 ARP Request tới lớp liên kết dữ liệu 1 địa chỉ broadcast FF:FF:FF:FF:FF:FF. Bởi vì là 1 gói tin broadcast, cho nên gói tin này sẽ được nhận bởi tất cả các máy ở trên cùng phân mạng đó. Về cơ bản thì gói tin quảng bá này sẽ có nhiệm vụ hỏi "địa chỉ IP nào tương ứng với địa chỉ MAC XX:XX:XX:XX:XX:XX".

Các thiết bị có địa chỉ IP khác với địa chỉ IP trên sẽ tự động loại bỏ ARP Request. Máy đích có nhiệm vụ phản hồi lại gói tin địa chỉ MAC của nó thông qua ARP Reply. Ở thời điểm này, máy gửi bây giờ đã có thông tin về địa chỉ lớp 2 mà nó cần để giao tiếp với máy nhận, và nó lưu trữ thông tin đó ở trong bảng ARP cache để lần tiếp theo có thể gửi nhanh hơn mà không cần mất công dò hỏi lần nữa.

Hoạt động của ARP Cache Poisoning

ARP Cache Poisoning, hay còn được gọi là ARP Spoofing, là 1 quá trình gửi thông điệp ARP tới 1 Switch hay 1 Router bằng 1 địa chỉ MAC giả mạo nhằm mục đích nghe lén lưu lượng của thiết bị mục tiêu. Hình dưới đây minh họa quá trình này:



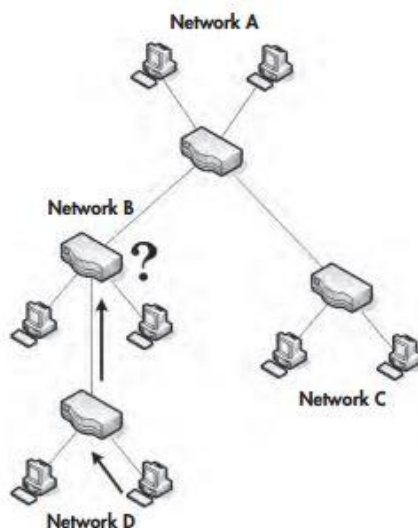
Hình 2.14. Bắt lưu lượng của thiết bị mục tiêu trên mạng Switch bằng ARP Cache Poisoning

ARP Cache Poisoning là 1 kỹ thuật nâng cao trong việc nghe đường truyền trong 1 mạng Switch. Nó được sử dụng phổ biến bởi hacker để gửi các gói tin địa chỉ sai tới máy nhận với mục tiêu để nghe trộm đường truyền hiện tại hoặc tấn công từ chối dịch vụ, nhưng ARP Cache Poisoning chỉ có thể phục vụ như là 1 cách hợp pháp để bắt các gói tin của máy mục tiêu trong mạng Switch.

- Nghe trong mạng sử dụng Router

Tất cả các kỹ thuật nghe trong mạng Switch đều có thể được sử dụng trong mạng Router. Chỉ có 1 việc cần quan tâm khi mà thực hiện với mạng Router là sự quan trọng của việc đặt máy nghe khi mà thực hiện xử lý 1 vấn đề liên quan đến nhiều phân mạng. Broadcast Domain của 1 thiết bị được mở rộng cho đến khi nó gặp Router. Khi đó, lưu lượng sẽ được chuyển giao sang dòng dữ liệu Router tiếp theo và bạn sẽ mất liên lạc với các gói tin đó cho đến khi bạn nhận được 1 ACK của các máy nhận trả về. Trong tình huống này, dữ liệu sẽ lưu chuyển qua nhiều Router, vì vậy rất quan trọng để thực hiện phân tích tất cả lưu lượng trên các giao diện của Router[4],[6].

Ví dụ, liên quan đến vấn đề liên kết, có thể gặp phải 1 mạng với 1 số phân mạng được kết nối với nhau thông qua các Router. Trong mạng đó, 1 phân mạng liên kết với 1 phân mạng với mục đích lưu trữ và tham chiếu dữ liệu. Vấn đề mà chúng ta đang cố gắng giải quyết là phân mạng D không thể kết nối với các thiết bị trong phân mạng A.



Hình 2.15. Nghe trong mạng sử dụng Router

Khi nghe lưu lượng của 1 thiết bị trong phân mạng D. Khi đó, có thể nhìn thấy rõ ràng lưu lượng truyền tới phân mạng A, nhưng không có biên

nhận (ACK) nào được gửi trả lại. Khi nghe luồng lưu lượng ở phân mạng cấp trên để tìm ra nguyên nhân vấn đề, tìm ra rằng lưu lượng bị hủy bởi Router ở phân mạng B. Cuối cùng dẫn đến việc kiểm tra cấu hình của Router, nếu đúng, hãy giải quyết vấn đề đó của. Đó là 1 ví dụ điển hình lý do vì sao cần nghe lưu lượng của nhiều thiết bị trên nhiều phân mạng với mục tiêu xác định chính xác vấn đề.

Sau khi bắt được gói tin, sẽ tiến hành phân tích giao thức của gói tin, mô tả quá trình bắt và diễn giải dữ liệu sống.

Căn cứ vào nội dung các trường trong cấu trúc của các gói tin bắt được, việc phân tích giao thức sẽ chỉ ra được một số thông tin của gói tin như: Giao thức, địa chỉ nguồn, địa chỉ đích, cổng nguồn, cổng đích, thời gian trao đổi gói tin, độ dài gói tin, có thể cả nội dung gói tin.v.v.

CHƯƠNG 3: XÂY DỰNG CÔNG CỤ HỖ TRỢ PHÂN TÍCH GÓI TIN

3.1. Mục tiêu công cụ hỗ trợ phân tích gói tin

Xây dựng công cụ hỗ trợ phân tích gói tin qua việc trích xuất các thông tin cần thiết từ dòng dữ liệu mạng. Công cụ có chức năng đọc và thống kê các giao thức mạng IP, ICMP, TCP, UDP... từ file dữ liệu chuẩn được lưu trữ dưới dạng .pcap. Dữ liệu mạng được bắt từ các chương trình bắt gói tin như Tcpdump hay Wireshark từ đó công cụ có thể trích xuất các trường cần thiết trong các gói tin và có thể thống kê các trường đó theo một mục đích cụ thể nào đó. **Phân tích, thiết kế công cụ hỗ trợ phân tích gói tin theo giao thức mạng**

Đầu vào:

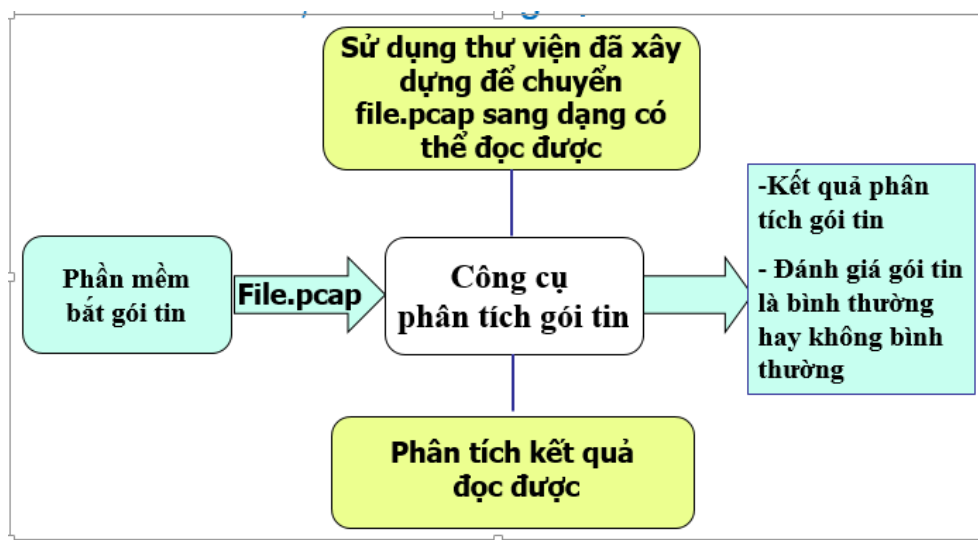
Là luồng dữ liệu thông tin mạng được bắt bằng các chương trình bắt gói tin và được lưu trữ dưới định dạng file .pcap

Đầu ra:

Công cụ sẽ đọc các gói tin .pcap và dựa vào các trường dữ liệu trong các gói tin để trích xuất ra các thông tin cần thiết sau đó sẽ thống kê chúng ở mức cơ bản nhất.

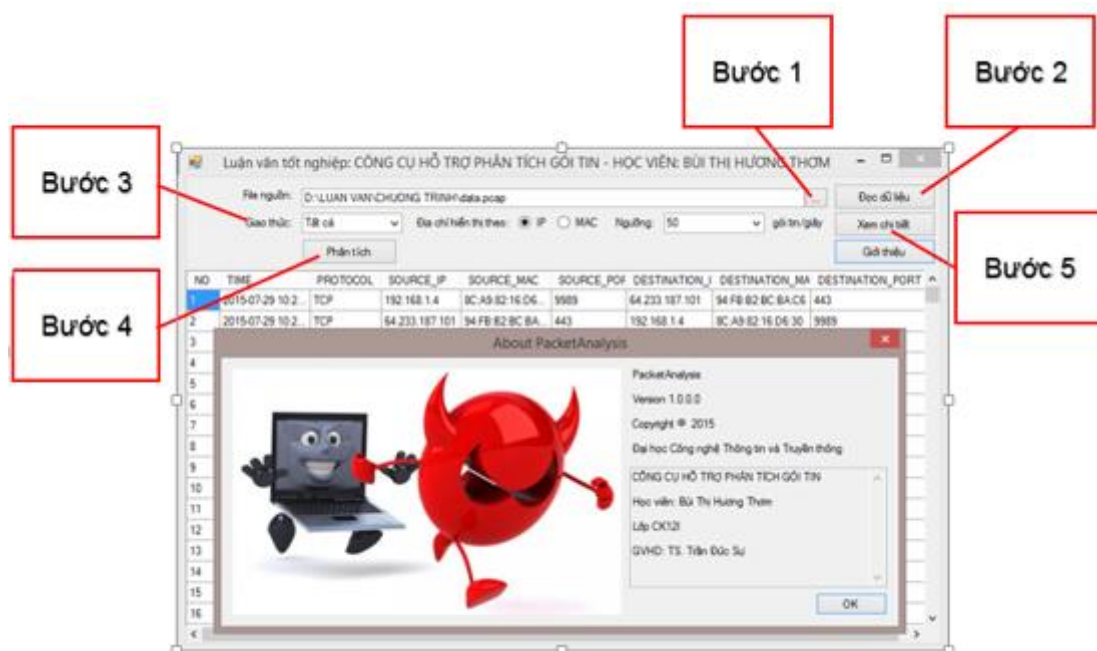
Công cụ có các chức năng thống kê theo tất cả các giao thức, theo từng giao thức với địa chỉ IP hoặc địa chỉ MAC và có nhận xét những gói tin bất thường hoặc không bất thường.

Mô hình hoạt động



Hình 3.1. Mô hình hoạt động

Các bước hoạt động:



Hình 3.2. Các bước hoạt động của công cụ

- Bước 1: Chọn file dữ liệu nguồn là file chứa các gói tin được bắt bởi các chương trình bắt gói tin được lưu dưới định dạng file.pcap.
- Bước 2: Click “Đọc dữ liệu”. Công cụ sẽ lọc và đưa ra các thông tin cơ bản của gói tin thể hiện bởi các cột:
 - NO: Số thứ tự.

- TIME: Hiển thị thời gian trao đổi của gói tin chính xác theo phần nghìn của giây.
- TIME_SEC: Cột này hiển thị thời gian trao đổi của gói tin theo từng giây, nhưng bị ẩn đi vì chỉ sử dụng để tính toán.
- PROTOCOL: Hiển thị giao thức của gói tin (TCP, PTP, UDP,...)
- SOURCE_IP: Hiển thị địa chỉ nguồn của gói tin theo địa chỉ IP
- SOURCE_MAC: Hiển thị địa chỉ nguồn của gói tin theo địa chỉ MAC
- SOURCE_PORT: Hiển thị cổng nguồn của gói tin
- DESTINATION_IP: Hiển thị địa chỉ đích của gói tin theo địa chỉ IP
- DESTINATION _MAC: Hiển thị địa chỉ đích của gói tin theo địa chỉ MAC
- DESTINATION _PORT: Hiển thị cổng đích của gói tin

Luận văn tốt nghiệp: CÔNG CỤ HỖ TRỢ PHÂN TÍCH GÓI TIN - HỌC VIÊN: BÙI THỊ HƯƠNG THƠM

File nguồn: D:\LUAN VAN\CHUONG TRINH\data.pcap

Giao thức: Tất cả Địa chỉ hiển thị theo: ☒ IP ☐ MAC Ngưỡng: 50 gói tin/giây

Phân tích

Đọc dữ liệu
Xem chi tiết
Giới thiệu

NO	TIME	PROTOCOL	SOURCE_IP	SOURCE_MAC	SOURCE_PORT	DESTINATION_IP	DESTINATION_MAC	DESTINATION_PORT
1	2015-07-29 10:24:22.089	TCP	192.168.1.4	8C:A9:82:16:D6:30	9989	64.233.187.101	94:FB:B2:BC:BA:C6	443
2	2015-07-29 10:24:22.166	TCP	64.233.187.101	94:FB:B2:BC:BA:C6	443	192.168.1.4	8C:A9:82:16:D6:30	9989
3	2015-07-29 10:24:25.138	UDP	192.168.1.1	94:FB:B2:BC:BA:C6	1900	239.255.255.250	8C:A9:82:16:D6:30	1900
4	2015-07-29 10:24:25.139	UDP	192.168.1.1	94:FB:B2:BC:BA:C6	1900	239.255.255.250	8C:A9:82:16:D6:30	1900
5	2015-07-29 10:24:25.139	UDP	192.168.1.1	94:FB:B2:BC:BA:C6	1900	239.255.255.250	8C:A9:82:16:D6:30	1900
6	2015-07-29 10:24:25.140	UDP	192.168.1.1	94:FB:B2:BC:BA:C6	1900	239.255.255.250	8C:A9:82:16:D6:30	1900
7	2015-07-29 10:24:25.140	UDP	192.168.1.1	94:FB:B2:BC:BA:C6	1900	239.255.255.250	8C:A9:82:16:D6:30	1900
8	2015-07-29 10:24:25.140	UDP	192.168.1.1	94:FB:B2:BC:BA:C6	1900	239.255.255.250	8C:A9:82:16:D6:30	1900
9	2015-07-29 10:24:25.141	UDP	192.168.1.1	94:FB:B2:BC:BA:C6	1900	239.255.255.250	8C:A9:82:16:D6:30	1900
10	2015-07-29 10:24:25.141	UDP	192.168.1.1	94:FB:B2:BC:BA:C6	1900	239.255.255.250	8C:A9:82:16:D6:30	1900
11	2015-07-29 10:24:25.141	UDP	192.168.1.1	94:FB:B2:BC:BA:C6	1900	239.255.255.250	8C:A9:82:16:D6:30	1900
12	2015-07-29 10:24:25.142	UDP	192.168.1.1	94:FB:B2:BC:BA:C6	1900	239.255.255.250	8C:A9:82:16:D6:30	1900
13	2015-07-29 10:24:25.142	UDP	192.168.1.1	94:FB:B2:BC:BA:C6	1900	239.255.255.250	8C:A9:82:16:D6:30	1900
14	2015-07-29 10:24:25.143	UDP	192.168.1.1	94:FB:B2:BC:BA:C6	1900	239.255.255.250	8C:A9:82:16:D6:30	1900
15	2015-07-29 10:24:25.143	UDP	192.168.1.1	94:FB:B2:BC:BA:C6	1900	239.255.255.250	8C:A9:82:16:D6:30	1900
16	2015-07-29 10:24:25.143	UDP	192.168.1.1	94:FB:B2:BC:BA:C6	1900	239.255.255.250	8C:A9:82:16:D6:30	1900
17	2015-07-29 10:24:25.143	UDP	192.168.1.1	94:FB:B2:BC:BA:C6	1900	239.255.255.250	8C:A9:82:16:D6:30	1900
18	2015-07-29 10:24:25.144	UDP	192.168.1.1	94:FB:B2:BC:BA:C6	1900	239.255.255.250	8C:A9:82:16:D6:30	1900
19	2015-07-29 10:24:27.242	TCP	192.168.1.4	8C:A9:82:16:D6:30	10023	123.30.175.29	94:FB:B2:BC:BA:C6	80
20	2015-07-29 10:24:27.247	UDP	192.168.1.4	8C:A9:82:16:D6:30	53092	123.30.175.88	94:FB:B2:BC:BA:C6	53

Hình 3.3. Thống kê ban đầu của các gói tin

- Bước 3:

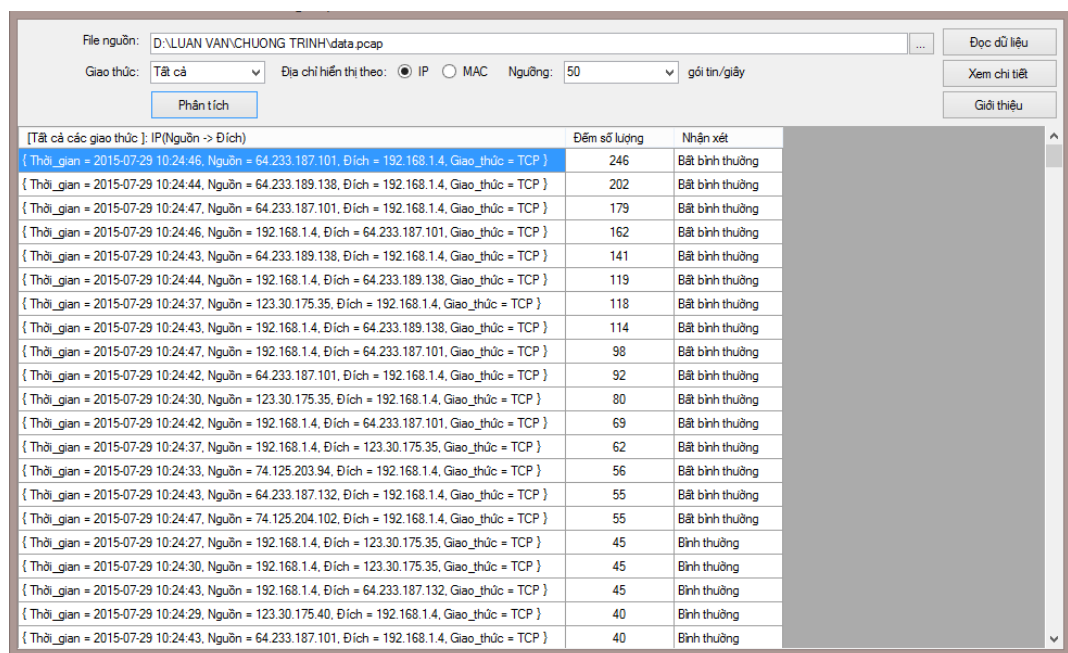
- Lựa chọn “Giao thức” của gói tin để lọc: Có thể lựa chọn 1 giao thức bất kỳ hoặc tất cả các giao thức để công cụ lọc theo
- Lựa chọn “Địa chỉ hiển thị” theo địa chỉ IP hoặc địa chỉ MAC, lựa chọn địa chỉ hiển thị nào thì khi lọc xong, địa chỉ nguồn và địa chỉ đích của gói tin sẽ hiển thị theo lựa chọn đó.
- Lựa chọn “Ngưỡng” (số gói tin/giây): Tùy thuộc vào người sử dụng có thể thay đổi theo mục đích sử dụng.

- Bước 4: Click “Phân tích”. Tùy thuộc vào sự lựa chọn ở bước 3 mà công cụ hiển thị ra các thông tin cần thiết.

- Bước 5: Click “Xem chi tiết” để có thể xem lại file dữ liệu nguồn sau khi lọc bằng công cụ mà không phải thông qua Bước 1.

Các kết quả phân tích gói tin:

- Thống kê gói tin theo địa chỉ IP của tất cả các giao thức



[Tất cả các giao thức]: IP(Nguồn -> Đích)	Đếm số lượng	Nhận xét
{ Thời gian = 2015-07-29 10:24:46, Nguồn = 64.233.187.101, Đích = 192.168.1.4, Giao_thức = TCP }	246	Bất bình thường
{ Thời gian = 2015-07-29 10:24:44, Nguồn = 64.233.189.138, Đích = 192.168.1.4, Giao_thức = TCP }	202	Bất bình thường
{ Thời gian = 2015-07-29 10:24:47, Nguồn = 64.233.187.101, Đích = 192.168.1.4, Giao_thức = TCP }	179	Bất bình thường
{ Thời gian = 2015-07-29 10:24:46, Nguồn = 192.168.1.4, Đích = 64.233.187.101, Giao_thức = TCP }	162	Bất bình thường
{ Thời gian = 2015-07-29 10:24:43, Nguồn = 64.233.189.138, Đích = 192.168.1.4, Giao_thức = TCP }	141	Bất bình thường
{ Thời gian = 2015-07-29 10:24:44, Nguồn = 192.168.1.4, Đích = 64.233.189.138, Giao_thức = TCP }	119	Bất bình thường
{ Thời gian = 2015-07-29 10:24:37, Nguồn = 123.30.175.35, Đích = 192.168.1.4, Giao_thức = TCP }	118	Bất bình thường
{ Thời gian = 2015-07-29 10:24:43, Nguồn = 192.168.1.4, Đích = 64.233.189.138, Giao_thức = TCP }	114	Bất bình thường
{ Thời gian = 2015-07-29 10:24:47, Nguồn = 192.168.1.4, Đích = 64.233.187.101, Giao_thức = TCP }	98	Bất bình thường
{ Thời gian = 2015-07-29 10:24:42, Nguồn = 64.233.187.101, Đích = 192.168.1.4, Giao_thức = TCP }	92	Bất bình thường
{ Thời gian = 2015-07-29 10:24:30, Nguồn = 123.30.175.35, Đích = 192.168.1.4, Giao_thức = TCP }	80	Bất bình thường
{ Thời gian = 2015-07-29 10:24:42, Nguồn = 192.168.1.4, Đích = 64.233.187.101, Giao_thức = TCP }	69	Bất bình thường
{ Thời gian = 2015-07-29 10:24:37, Nguồn = 192.168.1.4, Đích = 123.30.175.35, Giao_thức = TCP }	62	Bất bình thường
{ Thời gian = 2015-07-29 10:24:33, Nguồn = 74.125.203.94, Đích = 192.168.1.4, Giao_thức = TCP }	56	Bất bình thường
{ Thời gian = 2015-07-29 10:24:43, Nguồn = 64.233.187.132, Đích = 192.168.1.4, Giao_thức = TCP }	55	Bất bình thường
{ Thời gian = 2015-07-29 10:24:47, Nguồn = 74.125.204.102, Đích = 192.168.1.4, Giao_thức = TCP }	55	Bất bình thường
{ Thời gian = 2015-07-29 10:24:27, Nguồn = 192.168.1.4, Đích = 123.30.175.35, Giao_thức = TCP }	45	Bình thường
{ Thời gian = 2015-07-29 10:24:30, Nguồn = 192.168.1.4, Đích = 123.30.175.35, Giao_thức = TCP }	45	Bình thường
{ Thời gian = 2015-07-29 10:24:43, Nguồn = 192.168.1.4, Đích = 64.233.187.132, Giao_thức = TCP }	45	Bình thường
{ Thời gian = 2015-07-29 10:24:29, Nguồn = 123.30.175.40, Đích = 192.168.1.4, Giao_thức = TCP }	40	Bình thường
{ Thời gian = 2015-07-29 10:24:43, Nguồn = 64.233.187.101, Đích = 192.168.1.4, Giao_thức = TCP }	40	Bình thường

Hình 3.4. Thống kê gói tin theo địa chỉ IP của tất cả các giao thức
Bảng thống kê sẽ liệt kê ra các thông tin cơ bản sau:

- Thời gian: Được tính trong 1 giây
- Địa chỉ nguồn và địa chỉ đích, giao thức của các gói tin

Công cụ sẽ đọc và liệt kê ra địa chỉ IP nguồn và địa chỉ IP đích của tất cả các gói tin theo các giao thức. Nếu địa chỉ nguồn, địa chỉ đích và giao thức của gói tin trùng với địa chỉ nguồn, địa chỉ đích và giao thức của gói tin sau thì công cụ sẽ chỉ in một lần địa chỉ nguồn, đích và giao thức của gói tin.

- **Đếm số lượng:** Là số lần xuất hiện trùng nhau của địa chỉ nguồn, đích và giao thức của gói tin.
- **Nhận xét:** Hiện thị thông báo bất thường và bình thường. Thông báo bất thường là khi số lượng gói tin gửi đi từ địa chỉ nguồn gửi đến địa chỉ đích trong vòng 1 giây lớn hơn 50 tức là 1 giây gửi đi lớn hơn 50 gói tin.

- **Thống kê gói tin theo địa chỉ MAC của tất cả các giao thức**

Tương tự như việc thống kê gói tin theo địa chỉ IP của các giao thức, nhưng điểm khác đó là địa chỉ IP sẽ được thay bằng địa chỉ MAC

Luận văn tốt nghiệp: CÔNG CỤ HỖ TRỢ PHÂN TÍCH GÓI TIN - HỌC VIÊN: BÙI THỊ HƯƠNG THƠM

File nguồn: D:\LUAN VAN\CHUONG TRINH\data.pcap

Giao thức: Tất cả Địa chỉ hiển thị theo: ☐ IP ☒ MAC Ngưỡng: 50 gói tin/giây

Phân tích

Đọc dữ liệu Xem chi tiết Giới thiệu

[Tất cả các giao thức]: IP(Nguồn -> Đích)	Đếm số lượng	Nhận xét
{ Thời_gian = 2015-07-29 10:24:30, Nguồn = 94:FB:B2:BC:BA:C6, Đích = 8C:A9:82:16:D6:30, Giao_thức = TCP }	98	Bất bình thường
{ Thời_gian = 2015-07-29 10:24:33, Nguồn = 94:FB:B2:BC:BA:C6, Đích = 8C:A9:82:16:D6:30, Giao_thức = TCP }	96	Bất bình thường
{ Thời_gian = 2015-07-29 10:24:27, Nguồn = 94:FB:B2:BC:BA:C6, Đích = 8C:A9:82:16:D6:30, Giao_thức = TCP }	94	Bất bình thường
{ Thời_gian = 2015-07-29 10:24:37, Nguồn = 8C:A9:82:16:D6:30, Đích = 94:FB:B2:BC:BA:C6, Giao_thức = TCP }	93	Bất bình thường
{ Thời_gian = 2015-07-29 10:24:42, Nguồn = 8C:A9:82:16:D6:30, Đích = 94:FB:B2:BC:BA:C6, Giao_thức = TCP }	86	Bất bình thường
{ Thời_gian = 2015-07-29 10:24:29, Nguồn = 94:FB:B2:BC:BA:C6, Đích = 8C:A9:82:16:D6:30, Giao_thức = TCP }	85	Bất bình thường
{ Thời_gian = 2015-07-29 10:24:33, Nguồn = 8C:A9:82:16:D6:30, Đích = 94:FB:B2:BC:BA:C6, Giao_thức = TCP }	76	Bất bình thường
{ Thời_gian = 2015-07-29 10:24:30, Nguồn = 8C:A9:82:16:D6:30, Đích = 94:FB:B2:BC:BA:C6, Giao_thức = TCP }	70	Bất bình thường
{ Thời_gian = 2015-07-29 10:24:36, Nguồn = 94:FB:B2:BC:BA:C6, Đích = 8C:A9:82:16:D6:30, Giao_thức = TCP }	63	Bất bình thường
{ Thời_gian = 2015-07-29 10:24:36, Nguồn = 8C:A9:82:16:D6:30, Đích = 94:FB:B2:BC:BA:C6, Giao_thức = TCP }	59	Bất bình thường
{ Thời_gian = 2015-07-29 10:24:52, Nguồn = 94:FB:B2:BC:BA:C6, Đích = 8C:A9:82:16:D6:30, Giao_thức = TCP }	48	Bình thường
{ Thời_gian = 2015-07-29 10:24:57, Nguồn = 8C:A9:82:16:D6:30, Đích = 94:FB:B2:BC:BA:C6, Giao_thức = TCP }	43	Bình thường
{ Thời_gian = 2015-07-29 10:24:51, Nguồn = 94:FB:B2:BC:BA:C6, Đích = 8C:A9:82:16:D6:30, Giao_thức = TCP }	41	Bình thường
{ Thời_gian = 2015-07-29 10:24:50, Nguồn = 94:FB:B2:BC:BA:C6, Đích = 8C:A9:82:16:D6:30, Giao_thức = TCP }	39	Bình thường
{ Thời_gian = 2015-07-29 10:24:50, Nguồn = 8C:A9:82:16:D6:30, Đích = 94:FB:B2:BC:BA:C6, Giao_thức = TCP }	38	Bình thường
{ Thời_gian = 2015-07-29 10:24:52, Nguồn = 8C:A9:82:16:D6:30, Đích = 94:FB:B2:BC:BA:C6, Giao_thức = TCP }	32	Bình thường
{ Thời_gian = 2015-07-29 10:24:28, Nguồn = 94:FB:B2:BC:BA:C6, Đích = 8C:A9:82:16:D6:30, Giao_thức = TCP }	29	Bình thường
{ Thời_gian = 2015-07-29 10:24:51, Nguồn = 8C:A9:82:16:D6:30, Đích = 94:FB:B2:BC:BA:C6, Giao_thức = UDP }	29	Bình thường
{ Thời_gian = 2015-07-29 10:24:29, Nguồn = 8C:A9:82:16:D6:30, Đích = 94:FB:B2:BC:BA:C6, Giao_thức = UDP }	27	Bình thường
{ Thời_gian = 2015-07-29 10:24:29, Nguồn = 94:FB:B2:BC:BA:C6, Đích = 8C:A9:82:16:D6:30, Giao_thức = UDP }	27	Bình thường
{ Thời_gian = 2015-07-29 10:24:40, Nguồn = 94:FB:B2:BC:BA:C6, Đích = 8C:A9:82:16:D6:30, Giao_thức = TCP }	23	Bình thường

Hình 3.5. Thống kê gói tin theo địa chỉ MAC của tất cả các giao thức

Ngoài ra, công cụ hỗ trợ việc phân tích gói tin theo 1 giao thức cụ thể, để biết được thời gian gửi, địa chỉ nguồn và địa chỉ đích (IP hoặc MAC) của

gói tin, đếm được số lượng trùng lặp khi địa chỉ nguồn và địa chỉ đích trùng nhau, đưa ra được cảnh báo bình thường hoặc bất bình thường với ngưỡng tùy chọn.

Ví dụ với file dữ liệu ban đầu, công cụ thống kê gói tin theo giao thức TCP

- Thống kê gói tin theo địa chỉ IP của giao thức TCP

Bảng thống kê sẽ liệt kê ra các thông tin cơ bản sau:

- Thời gian: Được tính trong 1 giây
- Địa chỉ nguồn và địa chỉ đích của các gói tin

Công cụ sẽ đọc và liệt kê ra địa chỉ IP nguồn và địa chỉ IP đích của tất cả các gói tin theo giao thức TCP. Nếu địa chỉ nguồn, địa chỉ đích của gói tin trước trùng với địa chỉ nguồn, địa chỉ đích của gói tin sau thì công cụ sẽ chỉ in một lần địa chỉ nguồn, đích của gói tin.

- Đếm số lượng: Là số lần xuất hiện trùng nhau của địa chỉ nguồn, đích của gói tin.
- Nhận xét: Hiện thị thông báo bất thường và bình thường. Thông báo bất thường là khi số lượng gói tin gửi đi từ địa chỉ nguồn gửi đến địa chỉ đích trong vòng 1 giây lớn hơn 50 tức là 1 giây gửi đi lớn hơn 50 gói tin.

Luận văn tốt nghiệp: CÔNG CỤ HỖ TRỢ PHÂN TÍCH GÓI TIN - HỌC VIÊN: BÙI THỊ HƯƠNG THƠM

File nguồn: D:\LUAN VAN\CHUONG TRINH\data.pcap

Giao thức: TCP Địa chỉ hiển thị theo: ☒ IP ☐ MAC Ngưỡng: 50 gói tin/giây

Phân tích

Đọc dữ liệu
Xem chi tiết
Giới thiệu

[Giao thức: TCP]: IP(Nguồn -> Đích)	Đếm số lượng	Nhận xét
{ Thời gian = 2015-07-29 10:24:46, Nguồn = 192.168.1.4, Đích = 64.233.187.101 }	162	Bất bình thường
{ Thời gian = 2015-07-29 10:24:43, Nguồn = 64.233.189.138, Đích = 192.168.1.4 }	141	Bất bình thường
{ Thời gian = 2015-07-29 10:24:44, Nguồn = 192.168.1.4, Đích = 64.233.189.138 }	119	Bất bình thường
{ Thời gian = 2015-07-29 10:24:37, Nguồn = 123.30.175.35, Đích = 192.168.1.4 }	118	Bất bình thường
{ Thời gian = 2015-07-29 10:24:43, Nguồn = 192.168.1.4, Đích = 64.233.189.138 }	114	Bất bình thường
{ Thời gian = 2015-07-29 10:24:47, Nguồn = 192.168.1.4, Đích = 64.233.187.101 }	98	Bất bình thường
{ Thời gian = 2015-07-29 10:24:42, Nguồn = 64.233.187.101, Đích = 192.168.1.4 }	92	Bất bình thường
{ Thời gian = 2015-07-29 10:24:30, Nguồn = 123.30.175.35, Đích = 192.168.1.4 }	80	Bất bình thường
{ Thời gian = 2015-07-29 10:24:42, Nguồn = 192.168.1.4, Đích = 64.233.187.101 }	69	Bất bình thường
{ Thời gian = 2015-07-29 10:24:37, Nguồn = 192.168.1.4, Đích = 123.30.175.35 }	62	Bất bình thường
{ Thời gian = 2015-07-29 10:24:33, Nguồn = 74.125.203.94, Đích = 192.168.1.4 }	56	Bất bình thường
{ Thời gian = 2015-07-29 10:24:43, Nguồn = 64.233.187.132, Đích = 192.168.1.4 }	55	Bất bình thường
{ Thời gian = 2015-07-29 10:24:47, Nguồn = 74.125.204.102, Đích = 192.168.1.4 }	55	Bất bình thường
{ Thời gian = 2015-07-29 10:24:27, Nguồn = 192.168.1.4, Đích = 123.30.175.35 }	45	Bình thường
{ Thời gian = 2015-07-29 10:24:30, Nguồn = 192.168.1.4, Đích = 123.30.175.35 }	45	Bình thường
{ Thời gian = 2015-07-29 10:24:43, Nguồn = 192.168.1.4, Đích = 64.233.187.132 }	45	Bình thường
{ Thời gian = 2015-07-29 10:24:29, Nguồn = 123.30.175.40, Đích = 192.168.1.4 }	40	Bình thường
{ Thời gian = 2015-07-29 10:24:43, Nguồn = 64.233.187.101, Đích = 192.168.1.4 }	40	Bình thường
{ Thời gian = 2015-07-29 10:24:27, Nguồn = 123.30.175.35, Đích = 192.168.1.4 }	39	Bình thường
{ Thời gian = 2015-07-29 10:24:33, Nguồn = 192.168.1.4, Đích = 74.125.203.94 }	37	Bình thường
{ Thời gian = 2015-07-29 10:24:36, Nguồn = 123.30.175.40, Đích = 192.168.1.4 }	35	Bình thường

Hình 3.6. Thống kê gói tin theo địa chỉ IP của giao thức TCP

- Thống kê gói tin theo địa chỉ MAC của giao thức TCP

Tương tự như việc thống kê gói tin theo địa chỉ IP của giao thức TCP, nhưng điểm khác đó là địa chỉ IP sẽ được thay bằng địa chỉ MAC.

Luận văn tốt nghiệp: CÔNG CỤ HỖ TRỢ PHÂN TÍCH GÓI TIN - HỌC VIÊN: BÙI THỊ HƯƠNG THƠM

File nguồn: D:\LUAN VAN\CHUONG TRINH\data.pcap

Giao thức: TCP Địa chỉ hiển thị theo: ☐ IP ☒ MAC Ngưỡng: 50 gói tin/giây

Phân tích

Đọc dữ liệu
Xem chi tiết
Giới thiệu

[Giao thức: TCP]: IP(Nguồn -> Đích)	Đếm số lượng	Nhận xét
{ Thời gian = 2015-07-29 10:24:44, Nguồn = 8C:A9:82:16:D6:30, Đích = 94:FB:B2:BC:BA:C6 }	157	Bất bình thường
{ Thời gian = 2015-07-29 10:24:37, Nguồn = 94:FB:B2:BC:BA:C6, Đích = 8C:A9:82:16:D6:30 }	135	Bất bình thường
{ Thời gian = 2015-07-29 10:24:27, Nguồn = 8C:A9:82:16:D6:30, Đích = 94:FB:B2:BC:BA:C6 }	124	Bất bình thường
{ Thời gian = 2015-07-29 10:24:42, Nguồn = 94:FB:B2:BC:BA:C6, Đích = 8C:A9:82:16:D6:30 }	115	Bất bình thường
{ Thời gian = 2015-07-29 10:24:29, Nguồn = 8C:A9:82:16:D6:30, Đích = 94:FB:B2:BC:BA:C6 }	98	Bất bình thường
{ Thời gian = 2015-07-29 10:24:30, Nguồn = 94:FB:B2:BC:BA:C6, Đích = 8C:A9:82:16:D6:30 }	98	Bất bình thường
{ Thời gian = 2015-07-29 10:24:33, Nguồn = 94:FB:B2:BC:BA:C6, Đích = 8C:A9:82:16:D6:30 }	96	Bất bình thường
{ Thời gian = 2015-07-29 10:24:27, Nguồn = 94:FB:B2:BC:BA:C6, Đích = 8C:A9:82:16:D6:30 }	94	Bất bình thường
{ Thời gian = 2015-07-29 10:24:37, Nguồn = 8C:A9:82:16:D6:30, Đích = 94:FB:B2:BC:BA:C6 }	93	Bất bình thường
{ Thời gian = 2015-07-29 10:24:42, Nguồn = 8C:A9:82:16:D6:30, Đích = 94:FB:B2:BC:BA:C6 }	86	Bất bình thường
{ Thời gian = 2015-07-29 10:24:29, Nguồn = 94:FB:B2:BC:BA:C6, Đích = 8C:A9:82:16:D6:30 }	85	Bất bình thường
{ Thời gian = 2015-07-29 10:24:33, Nguồn = 8C:A9:82:16:D6:30, Đích = 94:FB:B2:BC:BA:C6 }	76	Bất bình thường
{ Thời gian = 2015-07-29 10:24:30, Nguồn = 8C:A9:82:16:D6:30, Đích = 94:FB:B2:BC:BA:C6 }	70	Bất bình thường
{ Thời gian = 2015-07-29 10:24:36, Nguồn = 94:FB:B2:BC:BA:C6, Đích = 8C:A9:82:16:D6:30 }	63	Bất bình thường
{ Thời gian = 2015-07-29 10:24:36, Nguồn = 8C:A9:82:16:D6:30, Đích = 94:FB:B2:BC:BA:C6 }	59	Bất bình thường
{ Thời gian = 2015-07-29 10:24:52, Nguồn = 94:FB:B2:BC:BA:C6, Đích = 8C:A9:82:16:D6:30 }	48	Bình thường
{ Thời gian = 2015-07-29 10:24:57, Nguồn = 8C:A9:82:16:D6:30, Đích = 94:FB:B2:BC:BA:C6 }	43	Bình thường
{ Thời gian = 2015-07-29 10:24:51, Nguồn = 94:FB:B2:BC:BA:C6, Đích = 8C:A9:82:16:D6:30 }	41	Bình thường
{ Thời gian = 2015-07-29 10:24:50, Nguồn = 94:FB:B2:BC:BA:C6, Đích = 8C:A9:82:16:D6:30 }	39	Bình thường
{ Thời gian = 2015-07-29 10:24:50, Nguồn = 8C:A9:82:16:D6:30, Đích = 94:FB:B2:BC:BA:C6 }	38	Bình thường
{ Thời gian = 2015-07-29 10:24:52, Nguồn = 8C:A9:82:16:D6:30, Đích = 94:FB:B2:BC:BA:C6 }	32	Bình thường

Hình 3.7. Thống kê gói tin theo địa chỉ MAC của giao thức TCP

Công cụ được xây dựng bằng công cụ hỗ trợ lập trình Microsoft Visual Studio, viết bởi ngôn ngữ C# và có thể chạy trên nền hệ điều hành Window. Trong đó có sự hỗ trợ chủ lực của LINQ (*Language Integrated Query*), là thư viện mở rộng cho các ngôn ngữ lập trình C# và Visual Basic.NET cung cấp khả năng truy vấn trực tiếp dữ liệu Object, CSDL và XML. LINQ là một tập hợp các thành phần mở rộng cho phép viết các câu truy vấn dữ liệu ngay trong một ngôn ngữ lập trình, như C# hoặc VB.NET. Khi tạo một đối tượng LINQ thì Visual Studio sẽ tự động sinh ra các lớp có các thành phần tương ứng với CSDL của chúng ta. Khi muốn truy vấn, làm việc với CSDL ta chỉ việc gọi và truy xuất các hàm, thủ tục tương ứng của LINQ mà không cần quan tâm đến các câu lệnh SQL thông thường

C# (C Sharp) là một ngôn ngữ lập trình bậc cao, hiện đại và hướng đối tượng. C# là sự phát triển của C++ và Java nhằm mục đích đơn giản hóa và giúp việc lập trình dễ dàng hơn. Các chương trình C# được biên dịch thì cần phải cài đặt .NET Framework có version tương ứng với thư viện gốc được chương trình tham chiếu đến để viết mã. Nếu không thì chương trình sẽ báo lỗi. Hiện nay thì tất cả các hệ điều hành Windows đều chứa sẵn .NET Framework.

KẾT LUẬN

- Kết quả của luận văn:

- Trình bày tổng quan về những vấn đề liên quan đến điều tra số bao gồm khái niệm, ứng dụng, quy trình thực hiện cũng như các loại hình điều tra số phổ biến mà trọng tâm là điều tra mạng. Giới thiệu về phân tích điều tra mạng cùng vai trò và ứng dụng của phân tích điều tra mạng. Trình bày chi tiết về nền tảng kỹ thuật của phân tích điều tra mạng từ các dạng hệ điều hành, các loại dịch vụ cho đến các giao thức mạng phổ biến. Từ đó hiểu rõ thêm về các kỹ thuật phân tích như phân tích gói dữ liệu, thống kê lưu lượng, các dạng nhật ký, sự kiện....

- Xây dựng được công cụ hỗ trợ phân tích gói tin trong điều tra mạng với chức năng thống kê theo giao thức TCP, UDP, ICMP, ARP, RARP,...cho các gói tin dưới dạng cơ bản nhất.

Bên cạnh những kết quả thu được học viên tự thấy luận văn còn nhiều hạn chế như: Khả năng diễn đạt và lập luận trong luận văn còn chưa tinh tế, chương trình mô phỏng còn đơn giản, chưa trau chuốt được về hình thức.

- Hướng phát triển của luận văn:

Lĩnh vực điều tra mạng hiện nay còn khá mới mẻ mà nguồn lực lại hạn chế, vì thế việc đẩy mạnh phát triển lĩnh vực này là một vấn đề thiết yếu trong xã hội hiện đại ngày nay.

Dựa vào kết quả ban đầu thu được của luận văn, học viên hi vọng công cụ hỗ trợ phân tích sẽ hoàn thiện hơn, có nhiều chức năng hơn sẽ đi vào thống kê chi tiết từng trường hợp từng khía cạnh của các cuộc tấn công mạng để từ đó nó có thể giúp người quản trị dễ dàng hơn trong việc phát hiện và phong chống được các cuộc tấn công mạng.

TÀI LIỆU THAM KHẢO

Tiếng Việt

1. Trần Đức Sự, Phạm Minh Thuận (2013), *Giáo trình Phòng chống và điều tra tội phạm máy tính*, Học viện mật mã.
2. Thái Hồng Nhi, Phạm Minh Việt (2004), *An toàn thông tin mạng máy tính, truyền số liệu và truyền dữ liệu*, Nxb Khoa học và kỹ thuật.

Tiếng Anh

3. Sherri Davidoff, Jonathan Ham (2012), *Network Forensics Tracking Hackers through Cyberspace*.
4. Emmanuel S. Pilli, R.C. Joshi & Rajdeep Niyogi (2010), *A Generic Framework for Network Forensics*.
5. Natarajan Meghanathan, Sumanth Reddy Allam and Loretta A. Moore, (2009) *Tools and Techniques for Network Forensics*.
6. Siti Rahayu Selamat, Robiah Yusof, Shahrin Sahib (2008), *Mapping Process of Digital Forensic Investigation Framework*.
7. Gary C. Kessler, Champlain College, Center for Digital Investigation, Burlington, Vermont (2006), *The Case for Teaching Network Protocols to Computer Forensics Examiners*.
8. Srinivas Mukkamala & Andrew H. Sung (2003), *Identifying Significant Features for Network Forensic Analysis Using Artificial Intelligent Techniques*.

* Trang web

9. <http://www.binarytides.com/packet-sniffer-code-c-libpcap-linux-sockets/>
10. <http://securitydaily.net/computer-forensics-va-nhung-dieu-can-biet/>
11. <http://antoanthongtin.ictu.edu.vn/vi/di-u-tra-s/134-tong-quan-computer-forensics>
12. <http://antoanthongtin.ictu.edu.vn/vi/chi-n-tranh-khong-gian-m-ng/166-tu-an-ninh-mang-toi-chien-tranh-mang>

PHỤ LỤC

Mã nguồn chương trình:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using PcapDotNet.Core;
using PcapDotNet.Packets;

namespace PacketAnalysis
{
    public partial class Form2 : Form
    {
        public Form2()
        {
            InitializeComponent();
        }

        private void Form2_Load(object sender, EventArgs e)
        {

        }

        private void cmdOpenFile_Click(object sender, EventArgs e)
        {
            OfflinePacketDevice selectedDevice = new
OfflinePacketDevice(txtFileName.Text);
            init_data();
            using (PacketCommunicator communicator =
selectedDevice.Open(65536, PacketDeviceOpenAttributes.Promiscuous,
1000))
            {
                communicator.ReceivePackets(0, DispatcherHandler);
                //MessageBox.Show();
            }
            get_group_protocal();
        }
    }
}
```

```

        dgv.DataSource = dt;
        dgv.Columns[2].Visible = false;
    }
    int stt = 0;
    void get_group_protocal()
    {
        try
        {
            if (dt != null)
            {
                var q = from r in dt.AsEnumerable()
                        group r by r["PROTOCOL"] into g
                        select new
                        {
                            g.Key
                        };
                cboFilter.Items.Clear();
                const string all = "Tất cả";
                cboFilter.Items.Add(all);
                foreach (var i in q.ToList())
                {
                    cboFilter.Items.Add(i.Key.ToString());
                }
                cboFilter.Sorted = true;
                cboFilter.Text = all;
            }
        }
        catch (Exception e)
        {
            MessageBox.Show(e.Message, "Error get group protocal",
                MessageBoxButtons.OK, MessageBoxIcon.Error);
        }
    }
    DataTable dt;
    string[] arrayTenCot = { "NO", "TIME", "TIME_SEC",
                            "PROTOCOL",
                            "SOURCE_IP", "SOURCE_MAC",
SOURCE_PORT",
                            "DESTINATION_IP", "DESTINATION_MAC",
"DESTINATION_PORT" };

```

```

void init_data()
{
    stt = 0;
    dt = new DataTable();
    foreach (string TenCot in arrayTenCot)
    {
        DataColumn cot = new DataColumn(TenCot.ToUpper(),
typeof(string));
        cot.DefaultValue = string.Empty;
        dt.Columns.Add(cot);
    }
}
private void DispatcherHandler(Packet packet)
{
    try
    {
        stt++;
        string no = stt.ToString();
        string time = packet.Timestamp.ToString("yyyy-MM-dd
hh:mm:ss.fff");
        string timeSEC = packet.Timestamp.ToString("yyyy-MM-dd
hh:mm:ss");
        string protocol = packet.Ethernet.IpV4.Protocol.ToString();
        if (protocol == "InternetControlMessageProtocol") protocol =
"ICMP";
        if (protocol == "InternetGroupManagementProtocol") protocol =
"IGMP";
        if (protocol == "PerformanceTransparencyProtocol") protocol =
"PTP";
        if (protocol == "IpsilonFlowManagementProtocol") protocol =
"IFMP";
        protocol = protocol.ToUpper();
        string sourceIP = packet.Ethernet.IpV4.Source.ToString();
        string sourceMAC = packet.Ethernet.Source.ToString();
        string sourcePORT =
packet.Ethernet.IpV4.Tcp.SourcePort.ToString();
        string destinationIP = packet.Ethernet.IpV4.Destination.ToString();
        string destinationMAC = packet.Ethernet.Destination.ToString();
        string destinationPORT =
packet.Ethernet.IpV4.Tcp.DestinationPort.ToString();
    }
}

```

```

        string[] arrayDuLieu = { no, time, timeSEC, protocol, sourceIP,
sourceMAC, sourcePORT, destinationIP, destinationMAC, destinationPORT
};

        DataRow dong = dt.NewRow();
        for (int i = 0; i < arrayTenCot.Length; i++)
            dong[arrayTenCot[i]] = arrayDuLieu[i] == null ? string.Empty :
arrayDuLieu[i].ToString();
        dt.Rows.Add(dong);
        cmdOpen.Text = "Read " + no + " packet";
        Application.DoEvents();
    }
    catch { }
    finally
    {
        cmdOpen.Text = "Đọc dữ liệu";
    }
}

```

```

void do_filter(string filter, int max)
{
    try
    {
        if (dt != null)
        {
            if (rIP.Checked)
            {
                var q = from r in dt.AsEnumerable()
                    where r["PROTOCOL"].ToString() == filter
                    group r by new
                    {
                        Thời_gian = r["TIME_SEC"],
                        Nguồn = r["SOURCE_IP"],
                        Đích = r["DESTINATION_IP"],
                    } into g
                    let count = g.Count()
                    orderby count descending
                    select new
                    {

```

```

        X = g.Key,
        Count = count,
        Noitice = (count < max) ? "Bình thường" : "Bất bình
thường"
    };
    dgv.DataSource = q.ToList();
}
else
{
    var q = from r in dt.AsEnumerable()
    where r["PROTOCOL"].ToString() == filter
    group r by new
    {
        Thời_gian = r["TIME_SEC"],
        Nguồn = r["SOURCE_MAC"],
        Đích = r["DESTINATION_MAC"],
    } into g
    let count = g.Count()
    orderby count descending
    select new
    {
        X = g.Key,
        Count = count,
        Noitice = (count < max) ? "Bình thường" : "Bất bình
thường"
    };
    dgv.DataSource = q.ToList();
}
dgv.Columns[0].AutoSizeMode =
DataGridViewAutoSizeColumnMode.AllCells;
dgv.Columns[0].HeaderText = "[Giao thức: " + filter + "]:
IP(Nguồn -> Đích)";
dgv.Columns[1].HeaderText = "Đếm số lượng";
dgv.Columns[2].HeaderText = "Nhận xét";
dgv.Columns[1].DefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleCenter;
}
else
{

```

```

        MessageBox.Show("Chưa có dữ liệu", "Thông báo",
        MessageBoxButtons.OK, MessageBoxIcon.Warning);
    }
}
catch (Exception e)
{
    MessageBox.Show(e.Message, "Error do filter",
    MessageBoxButtons.OK, MessageBoxIcon.Error);
}
}
void do_filterALL(int max)
{
    try
    {
        if (dt != null)
        {
            if (rIP.Checked)
            {
                var q = from r in dt.AsEnumerable()
                //where r["PROTOCOL"].ToString() == filter
                group r by new
                {
                    Thời_gian = r["TIME_SEC"],
                    Nguồn = r["SOURCE_IP"],
                    Đích = r["DESTINATION_IP"],
                    Giao_thức = r["PROTOCOL"]
                } into g
                let count = g.Count()
                orderby count descending
                select new
                {
                    X = g.Key,
                    Count = count,
                    Noitice = (count < max) ? "Bình thường" : "Bất bình
thường"
                };
                dgv.DataSource = q.ToList();
            }
        }
    }
    else
    {

```

```

var q = from r in dt.AsEnumerable()
//where r["PROTOCOL"].ToString() == filter
group r by new
{
    Thời_gian = r["TIME_SEC"],
    Nguồn = r["SOURCE_MAC"],
    Đích = r["DESTINATION_MAC"],
    Giao_thức = r["PROTOCOL"]
} into g
let count = g.Count()
orderby count descending
select new
{
    X = g.Key,
    Count = count,
    Noitice = (count < max) ? "Bình thường" : "Bất bình
thường"
};
dgv.DataSource = q.ToList();
}

dgv.Columns[0].AutoSizeMode =
DataGridViewAutoSizeColumnMode.AllCells;
dgv.Columns[0].HeaderText = "[Tất cả các giao thức ]:
IP(Nguồn -> Đích)";
dgv.Columns[1].HeaderText = "Đếm số lượng";
dgv.Columns[2].HeaderText = "Nhận xét";
dgv.Columns[1].DefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleCenter;
}
else
{
    MessageBox.Show("Chưa có dữ liệu", "Thông báo",
    MessageBoxButtons.OK, MessageBoxIcon.Warning);
}
}
catch (Exception e)
{
    MessageBox.Show(e.Message, "Error do filter",
    MessageBoxButtons.OK, MessageBoxIcon.Error);
}

```



```

    }
}
private void cmdFilter_Click(object sender, EventArgs e)
{
    string filter = cboFilter.Text;
    if (filter != null && filter.Trim() != "")
    {
        int max = int.Parse(comboBox1.Text);
        if (filter!="Tất cả")
            do_filter(filter, max);
        else
            do_filterALL(max);
    }
}

private void cmdXemLai_Click(object sender, EventArgs e)
{
    dgv.DataSource = dt;
    dgv.Columns[2].Visible = false;
}

private void cmdFile_Click(object sender, EventArgs e)
{
    if (OFD.ShowDialog() == DialogResult.OK)
    {
        txtFileName.Text = OFD.FileName;
    }
}

private void cmdAbout_Click(object sender, EventArgs e)
{
    AboutBox1 f = new AboutBox1();
    f.ShowDialog();
}
}
}

```