Ôn tập An toàn mạng

Chương 1

- **Định nghĩa An toàn thông tin (NIST):**

Bảo vệ thông tin và hệ thống thông tin khỏi việc truy cập, sử dụng, tiết lộ, làm gián đoạn, sửa đổi hoặc phá hủy không được ủy quyền để đảm bảo tính bảo mật, toàn vẹn và khả dụng.

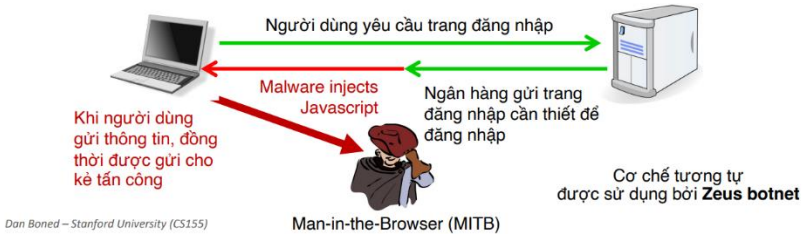
- Xem xét 3 **mục tiêu** lớn của ATTT:
  - **Tính bảo mật:** Tránh tiết lộ thông tin trái phép
  - **Tính toàn vẹn:** Tránh sửa đổi trái phép thông tin
  - **Tính sẵn có:** Đảm bảo thông tin và hệ thống kịp thời bởi những người được cho phép
- **Cybersecurity** (aka. Computer Security)
  - là tập hợp con của information security
  - là hoạt động bảo vệ mạng, máy tính và dữ liệu của tổ chức khỏi các cuộc truy cập trái phép, tấn công hoặc thiệt hại do các hoạt động khác.
- **Network Security**
  - là tập hợp con của cybersecurity
  - nhằm mục đích bảo vệ bất kỳ dữ liệu nào đang được gửi qua các thiết bị trong mạng để đảm bảo rằng thông tin không bị thay đổi hoặc bị chặn.
- **Hacker** - tội phạm mạng là những người liên quan đến việc phá vỡ/vượt qua bảo mật máy tính
- Có bao nhiêu loại hacker ?
  - Black Hat hackers (crackers)
  - White Hat hackers (ethical hackers)
  - Gray Hat hackers
- Hacker trẻ nhất thế giới: Kristoffer von Hassel - 2009
- “Hacking has evolved from teenage mischief into a billion-dollar growth business.”

"Hacking đã phát triển từ những trò tinh nghịch của tuổi teen thành một doanh nghiệp tăng trưởng có giá trị tỷ đô."

- Các cuộc tấn công mạng hàng đầu thế giới:
  - 1988: Morris Worm – the first Internet worm
  - 1994: Mitnick attack • 2000: MafiaBoy attack
  - 2008: Kaminsky attack
  - …
  - 2014: Heartbleed attack
  - 2016: Mirai Botnet: The fall of the Internet
  - 2017: **WannaCry**: A real epidemic – Sâu lan truyền qua một lỗ hổng hệ thống trong SMB (port 445)

- **Mục đích của hacker**:
  - Đánh cắp địa chỉ IP và băng thông
    - Mục tiêu của kẻ tấn công: trông giống như một người dùng Internet bình thường
    - Sử dụng địa chỉ IP của máy hoặc điện thoại bị nhiễm để: • Spam • Denial of Service • Click fraud
  - Đánh cắp thông tin đăng nhập của người dùng
    - Mục tiêu của kẻ tấn công: mật khẩu ngân hàng, mật khẩu công ty, mật khẩu chơi game...
    - Ví dụ: Silent Banker trojan

Silent Banker trojan



- Các cuộc tấn công trên điện thoại: **FinSpy spyware.**

Hoạt động trên iOS and Android (và Windows)

- **Tấn công Server-side**
  - Đánh cắp dữ liệu: số thẻ tín dụng, sở hữu trí tuệ, thông tin khách hang.
  - Ví dụ: Equifax (Tháng 7 2017), ≈ 143M dữ lieu “khách hàng” bị ảnh hưởng
  - Động cơ chính trị
    - DNC, Tunisia Facebook (Tháng 2. 2011), GitHub (Tháng 3. 2015)
  - Lây nhiễm cho người dùng đang truy cập
- **Vulnerabilities**
  - Có thể đến từ các lỗi trong phần mềm (thiết kế hoặc triển khai)
    - Ví dụ: Buffer overflow (the Morris worm vào năm 1988)
  - Các ứng dụng dễ bị tổn thương đang được khai thác
    - Tấn công độc hại được chia nhỏ theo loại ứng dụng mục tiêu (Kaspersky Security Bulletin 2019)
  - Có thể đến từ việc cấu hình không tốt (Misconfiguration)
  - Đến từ yếu tố con người (đào tạo kém)
- Bán lỗ hỏng 0-day: chương trình **bug bounty**.
- **Zero payment** – khoản chi trả:
  - **RCE**: Remote Code Execution (Thực thi Mã từ xa) - Chi trả cho khả năng thực thi mã từ xa trên hệ thống mục tiêu.
  - **LPE**: Local Privilege Escalation (Nâng cao đặc quyền cục bộ) - Chi trả khi có khả năng tăng cường đặc quyền trên máy tính cục bộ.
  - **SBX**: Sandbox Escape (Thoát khỏi Hộp cát) - Chi trả khi có khả năng vượt qua cơ chế cô lập bảo vệ (sandbox) trên các ứng dụng hay hệ thống.

Chương 2

- **Computer Security định nghĩa là (NIST):**

Biện pháp và kiểm soát nhằm đảm bảo tính bảo mật, tính toàn vẹn và khả năng sẵn có của tài sản hệ thống thông tin, bao gồm phần cứng, phần mềm, firmware, và thông tin đang được xử lý, lưu trữ và truyền thông.

- Ngoài "ba mục tiêu lớn" (bộ ba CIA), **2 mục tiêu phổ biến nhất** là:
  - **Tính xác thực (Authenticity)**: xác minh rằng người dùng có đúng như họ nói hay không và mỗi đầu vào đến hệ thống đều đến từ một nguồn đáng tin cậy.
  - **Trách nhiệm giải trình (Accountability):** lưu giữ hồ sơ về các hoạt động của người dùng để cho phép phân tích điều tra số sau này nhằm theo dõi các vi phạm an ninh hoặc hỗ trợ trong các tranh chấp giao dịch.
- Lỗ hổng, Đe doạ và Rủi ro:
  - **Lỗ hổng bảo mật**: một điểm yếu đã biết của tài nguyên hệ thống có thể bị khai thác bởi một hoặc nhiều kẻ tấn công.
  - **Đe doạ:** một sự cố mới hoặc mới được phát hiện có khả năng gây hại cho toàn bộ hệ thống hoặc công ty của bạn.
  - **Rủi ro**: khả năng mất mát hoặc thiệt hại khi một mối đe dọa khai thác lỗ hổng.
- **Tấn công (Attack):** một mối đe dọa được thực hiện (bởi tác nhân đe dọa hoặc kẻ thù), nếu thành công, dẫn đến vi phạm không mong muốn về bảo mật hoặc hậu quả của mối đe dọa.
  - **Tấn công chủ động (Active):** Cố gắng thay đổi tài nguyên hệ thống hoặc ảnh hưởng đến hoạt động của chúng, liên quan đến một số sửa đổi luồng dữ liệu hoặc tạo luồng giả (phát lại, giả mạo, sửa đổi thông báo, DOS).
  - **Tấn công thụ động (Passive):** Cố gắng tìm hiểu hoặc sử dụng thông tin từ hệ thống mà không ảnh hưởng đến tài nguyên hệ thống (phát hành nội dung thông báo, phân tích lưu lượng) => khó bị phát hiện.
  - Phân loại khác: tấn công bên trong (insider) và tấn công bên ngoài (outsider)

- **Countermeasure (biện pháp đối phó):** bất kỳ phương tiện nào được thực hiện để đối phó với một cuộc tấn công bảo mật.
    - Lý tưởng nhất: ngăn chặn các cuộc tấn công thành công.
    - Nếu không thể: phát hiện cuộc tấn công => phục hồi hoặc giảm nhẹ tác động của cuộc tấn công.
- Trusted Computing Base (TCB): giả sử một số phần nhỏ nhất của hệ thống không bị xâm phạm. Sau đó, xây dựng một môi trường an toàn trên đó.

Threat consequences (Hệ quả)	Threat Action (attack)
Unauthorized Disclosure - tiết lộ không chấp thuận: Một thực thể có được quyền truy cập vào dữ liệu mà thực thể đó không được phép.	**Exposure** – tiếp xúc **Interception** – chặn đứng **Inference** – rút ra kết luận, thông tin **Intrusion** – xâm nhập
Deception - lừa dối: Một thực thể được ủy quyền nhận dữ liệu giả mạo và tin rằng nó là đúng.	**Masquerade** – giả mạo **Falsification** – chỉnh sửa **Repudiation** – phủ nhận
Disruption - phá hoại: Làm gián đoạn hoặc ngăn chặn hoạt động đúng đắn của các dịch vụ và chức năng hệ thống.	**Incapacitation** – tàn phá khả năng hoạt động **Corruption** – phá hủy **Obstruction** – cản trở

	Tính sẵn sàng	Tính bí mật	Tính toàn vẹn
Phần cứng	Thiết bị bị đánh cắp hoặc bị vô hiệu hóa, do đó từ chối dịch vụ.	Một ổ USB không được mã hóa đã bị đánh cắp.	
Phần mềm	Các chương trình bị xóa, từ chối quyền truy cập của người dùng.	Một bản sao trái phép của phần mềm được tạo ra.	Một chương trình làm việc bị sửa đổi, có thể khiến nó bị lỗi trong khi thực thi hoặc khiến nó thực hiện một số tác vụ ngoài ý muốn.
Dữ liệu	Các tệp bị xóa, từ chối quyền truy cập của người dùng.	Việc đọc dữ liệu trái phép được thực hiện.	Các tập tin hiện có được sửa đổi hoặc các tập tin mới được tạo ra.
Communication Lines and Networks	Tin nhắn bị hủy hoặc bị xóa. Các đường dây hoặc mạng liên lạc không khả dụng	Tin nhắn được đọc. Lưu lượng tin nhắn được quan sát.	Tin nhắn được sửa đổi, trì hoãn, sắp xếp lại hoặc trùng lặp. Thông điệp sai là bịa đặt.

Câu hỏi	Câu trả lời
Chúng ta có thể tin tưởng vào chương trình "đăng nhập - login" trong bản phân phối Linux không? (ví dụ: Ubuntu)	=> Không! Chương trình đăng nhập có thể có backdoor. => ghi lại mật khẩu khi nhập. => Giải pháp: biên dịch lại chương trình đăng nhập từ mã nguồn
Chúng ta có thể tin tưởng mã nguồn đăng nhập không?	=> Không! Nhưng chúng ta có thể kiểm tra đoạn mã, sau đó biên dịch lại.
Tôi đặt một chiếc Laptop qua đường bưu điện. Khi nó đến, tôi có thể tin tưởng vào điều gì?	=> Các ứng dụng hoặc hệ điều hành có thể được cài sẵn backdoor. => Giải pháp: cài đặt lại OS và ứng dụng
Làm thế nào để cài đặt lại? Không thể tin tưởng OS để cài lại OS.	=> Boot Tails từ ổ USB (Debian)
Cần tin tưởng pre-boot BIOS, UEFI code. Chúng ta có thể tin tưởng?	=> Không (e.g. ShadowHammer operation in 2018)

Bài tập: [NT140.O11.ANTT]Nhóm 6 - Nộp bài tập buổi 2

Chương 3

- Malware is malicious software - hành động chống lại chủ sở hữu hoặc người dùng.

Một chương trình được chèn vào hệ thống, thường là một cách âm thầm, với ý định làm suy giảm tính bảo mật, tính toàn vẹn hoặc khả năng sẵn có của dữ liệu, ứng dụng hoặc hệ điều hành của nạn nhân hoặc gây phiền toái hoặc gây rối cho nạn nhân.

- **Cơ chế lan truyền:**
  - Infected content – Viruses
  - Vulnerability exploit – Worms
  - Social engineering – Spam email, trojans
- **Payload:**
  - System corruption – Logic booms
  - Tác nhân tấn công – Zombie, Bots
  - Đánh cắp thông tin – Keylogger, Phishing, Spyware
  - Trộm cắp – Backdoors, Rootkits

Advanced Persistent Threat

- **Tội phạm mạng**, thường là từ một quốc gia hoặc nhóm được tài trợ bởi quốc gia, hướng đến các mục tiêu kinh doanh và chính trị, sử dụng nhiều công nghệ xâm nhập và phần mềm độc hại, được áp dụng một cách kiên trì và hiệu quả đối với các mục tiêu cụ thể trong một khoảng thời gian kéo dài, thường được đặc hiệu là do các tổ chức được tài trợ bởi quốc gia.
  - **Advanced**: đã lựa chọn cẩn thận nhiều loại kỹ thuật thu thập thông tin tình báo và xâm nhập cũng như phần mềm độc hại.
  - **Persistent**: dần dần, và thường lén lút, được áp dụng cho đến khi mục tiêu bị xâm phạm.
  - **Threats**: do mục đích của những kẻ tấn công có tổ chức, có khả năng và được tài trợ tốt. (threat = capability + intent)
- **Hai kiểu malware tự sao chép (self-replicating):**
  - **Virus**: Lây nhiễm các chương trình / hệ thống bằng cách sửa đổi chúng.
    - Loại malware đầu tiên phổ biến rộng rãi.
    - Sự lan truyền được kích hoạt bởi hành động của người dùng (ví dụ: chạy chương trình bị nhiễm)
    - Được sử dụng như một thuật ngữ chung (“anti-virus” SW detects more than viruses!)
  - **Worm**: Phát tán các bản sao qua mạng.
    - Thường là một chương trình độc lập (không được đính kèm với một chương trình như virus)
    - Thường lan truyền tự động (không có sự tương tác của người dùng)
    - Khai thác các lỗ hổng (như buffer overflow) để lây lan
- **Làm thế nào để người dùng không biết về malware?**
  - **Trojan Horse**: Ẩn đằng sau một số chức năng mong muốn.
    - Malicious code included with game, utility, or other “tempting ware”
    - VD: AIDS Trojan (1989 - đĩa mềm được gửi qua đường bưu điện), phần mềm anti-virus giả.
  - **Rootkit**: Che giấu sự tồn tại của phần mềm độc hại
    - User-level rootkits thay thế các lệnh cơ bản của hệ thống
    - Linux: Replace “ps” (hide processes) and replace “ls” (hide files)
    - Windows: Replace Process Monitor and Windows Explorer
  - **Kernel-level rootkits** đi sâu hơn, ẩn mọi thứ khỏi mọi chương trình
  - **Backdoor**: Cho phép người dùng trái phép truy cập vào hệ thống
    - Thường là một cuộc tấn công từ bên trong - ví dụ: để giữ lại quyền truy cập sau khi rời khỏi
  - **Botnets**: Cung cấp cho cuộc tấn công một tài nguyên phân tán
    - Hệ thống bị nhiễm “zombies”
    - Kiểm soát bằng kỹ thuật “command and control”. Thường là một kênh công khai khó theo dõi (IRC, Twitter…)
    - Sử dụng phổ biến: để khởi động các cuộc tấn công khác hoặc gửi thư rác và đào coin!

- **Privacy-Invasive Software**: Gửi thông tin về hệ thống hoặc người dùng
- **Spyware** – có thể xem lịch sử duyệt web
- **Adware** - chạy quảng cáo trên máy nạn nhân
- **Keystroke loggers (keyloggers)** – bắt được mật khẩu, thậm chí khai thác webcam!
- **Ransomware**: Mã hóa các tập tin để người dùng không có quyền truy cập, yêu cầu thanh toán bằng bitcoin để lấy khóa giải mã

- **Hành vi của malware**
  - **Sự khác nhau vật trung gian truyền nhiễm (infection vectors)**
    - Phần mềm phân tán (disk hoặc network)
    - Các dịch vụ mạng có lỗ hổng
    - Ứng dung có lỗ hổng
    - E-mail: Tự động hoặc đánh lừa người dùng
  - **Kiểm soát hành vi độc hại**
    - Có thể thực thi ngay lập tức
    - Có thể "kích hoạt" vào một thời gian cụ thể ("time- bomb") hoặc điều kiện("logic-bomb")
    - Thường do nhân viên cũ bỏ lại - được kích hoạt sau khi bị sa thải
    - Có thể được điều khiển từ xa (như trong mạng botnet)

Virus

- **Virus** (Thuật ngữ do Fred Cohen đặt ra vào năm 1983): Mã độc hại đính kèm với nội dung đang hoạt động (active content - có thể là chương trình, tập lệnh, khu vực khởi động, thư viện,…)
- Quá trình khởi động:
  - Khởi động firmware (BIOS with POST, initialization, …)
  - Tải first-stage bootloader (master boot record - MBR)
  - Chạy mã được tìm thấy ở đó - thường là chuỗi đến second-stage bootloader
- Virus được kiểm soát sớm và hoàn toàn thay thế MBR
  - PC Virus đầu tiền (Brain) là boot-sector virus
  - Những công cụ phức tạp hơn có thể tạo hypervisor (như BluePill)
  - UEFI Secure Boot bảo vệ tốt chống lại điều này
- **1986: Brain virus (PC virus đầu tiền - MS-DOS)** - có nguồn gốc từ Pakistan ("Pakistani Brain"). Những gì nó làm:
  - Nằm trong high memory và thường trú
  - Tự sao chép chính nó vào boot sector
  - Tự sao chép bản gốc boot sector và các bản sao bổ sung của chính nó vào các vị trí đĩa khác nhau, được đánh dầu là "bad sectors"
  - Chặn tất cả yêu cầu đọc/ghi đĩa để giả mạo việc đọc boot sector (thay thế bản sao gốc)
  - Trong quá trình đọc / ghi đĩa, tự lây lan sang tất cả các đĩa không bị nhiễm
  - Không có thiệt hại trực tiếp
- **Macro Viruses**: sớm nhất: Melissa (1999): sử dụng macro để truy cập sổ địa chỉ Outlook
- **Virus Hoaxes**
  - Nhiều virus lừa bịp trong những năm qua:
    - "Virus Flambé": được đồn đại là đặt tốc độ đồng bộ hóa màn hình cao đến mức nó sẽ bùng cháy!
    - Trò lừa virus thiệp chúc mừng Blue Mountain: xác nhận có virus trong thiệp chúc mừng điện tử.
    - "Goodtimes" hoax (1994): Trò lừa bịp trên diện rộng đầu tiên

Worm

- Thay đổi các truyền nhiễm của malware: bây giờ ít "disk swapping" hơn, nhưng kết nối mạng nhiều hơn
- Không hoàn toàn khác với virus: có thể lây nhiễm các tập tin thực thi sau khi sử dụng mạng để phát tán. Nhưng thường chỉ được cài đặt trên hệ thống dưới dạng các chương trình bổ sung, hoàn chỉnh
- Sự làn truyền có thể tự động và yêu cầu người dùng làm
  - Thường cố lừa người dùng mở một tệp đính kèm đang hoạt động

- Tự động lây lan qua e-mail (ví dụ: khai thác lỗi Outlook) hoặc các dịch vụ mạng có lỗ hổng (MS IIS, SQL Server,…)
- The Internet Worm - Sự cố Internet nghiêm trọng trên diện rộng đầu tiên (02/11/1988)

=> CERT (computer emergency response team) được tạo ra để ứng phó với các sự cố

- Phân loại:
  - **Infamous Worms – Code Red**
    - Lây lan qua lỗ hổng bảo mật buffer overflow MS IIS
    - Được phát hiện vào mùa hè năm 2001
    - Ước tính có khoảng 750.000 máy chủ bị nhiễm
    - Có thể có động cơ chính trị: thông điệp "Hacked by Chinese" được để lại trên máy, bao gồm các cuộc tấn công DOS timebomb trên www.whitehouse.gov
    - Hai giai đoạn chính: quét / lây nhiễm và tấn công (dựa trên ngày tháng)
  - **Infamous Worms – Slammer**
    - Còn được gọi là "Sapphire" hoặc "SQL Slammer"
    - Lây lan qua lỗ hổng buffer overflow trong MS SQL Server
    - Được phát hiện vào đầu năm 2003
    - Khả năng lan truyền cực kỳ nhanh chóng!
      - Máy chủ bị nhiễm nhân đôi sau mỗi 8,5 giây
      - Lây nhiễm trên 90% máy chủ có lỗ hổng trong 10 phút
      - Bị nhiễm bởi dịch vụ UDP (không phải TCP)
    - Mạng quá tải, vô hiệu hóa các dịch vụ khác (Ví dụ: Nhiều máy ATM của Ngân hàng Mỹ ngừng hoạt động)
  - **Infamous Worms – Stuxnet**
    - Một trong những loại worm phức tạp nhất từng được phát hiện (được tìm thấy vào năm 2010)
    - Khai thác nhiều lần (ít nhất 4) 0-day được khai thác lan rộng
    - Có thể lây lan qua USB cũng như mạng
    - Bao gồm cả rootkit để ẩn mình
    - Payload độc hại chỉ được gọi trong một số tình huống nhất định (bom logic), cấu hình máy mục tiêu trùng khớp với máy ly tâm hạt nhân của Iran

Trojan horses

- **Trojan horses** là một chương trình hoặc tiện ích hữu ích, hoặc có vẻ hữu ích, chứa mã ẩn điều khiển, và khi được gọi ra, thực hiện một chức năng không mong muốn hoặc có hại.
- Các loại Trojan:
  - Remote Access Trojan (RAT)
  - Backdoor Trojans
  - Rootkit trojan
  - Proxy server Trojan
  - Mobile Trojan, IoT Trojan,
- **Wrappers** là những lớp bao kết nối trojan với một ứng dụng có vẻ chính hãng (trò chơi, văn phòng, phần mềm diệt virus,... hoặc ứng dụng đã được crack đầy đủ).
- **Các tạo Trojan:**
  - Bước 1: Tạo một gói Trojan mới bằng cách sử dụng Bộ Trojan Horse Construction Kit.
  - Bước 2: Tạo một "dropper," đó là một phần của gói bị nhiễm Trojan mà cài đặt mã độc hại lên hệ thống mục tiêu.
  - Bước 3: Tạo một "wrapper" bằng cách sử dụng các công cụ bao gói để cài đặt Trojan trên máy tính của nạn nhân.
  - Bước 4: Lan truyền Trojan.
  - Bước 5: Thực hiện dropper.

    - Bước 6: Thực hiện quy trình gây hại.
- **Ransomware:** mã hóa dữ liệu của người dùng và yêu cầu thanh toán để truy cập vào khóa cần thiết để khôi phục thông tin
- **Zombie và bots:**
    - **Bot** (hay còn gọi là. Robot, zombie, drone): PC bị xâm nhập, máy chủ, thiết bị nhúng như bộ định tuyến hoặc camera giám sát được sử dụng để khởi động các cuộc tấn công vào các máy khác
    - **Botnet**: một mạng (tập hợp) các bot
    - Chức năng của bot:
        - Các cuộc tấn công DDoS phân tán
        - Gửi thư rác
        - Sniffing traffic
        - Phát tán phần mềm độc hại mới
        - Cài đặt add-ons quảng cáo
        - Tấn công mạng lưới trò chuyện IRC
- Công thức tính sự lây lan của malware: Số máy chủ bị nhiễm trong thời gian (t + 1) = Số máy chủ bị nhiễm trong thời gian t + tỷ lệ lây nhiễm * Số máy chủ bị nhiễm trong thời gian t * Số máy chủ còn lại

Cách phòng tránh

- **Signature-Based: Recognizes "known-bad" code**
    - Nhà cung cấp có nhóm phân tích malware và cập nhật Signature
    - Thường có thể tránh bị phát hiện với những sửa đổi nhỏ
    - Người dùng phải cập nhật cơ sở dữ liệu virus đã biết!
    - Thận trọng: Nhiều chương trình anti-virus trên máy tính mới có "thời gian dùng thử miễn phí" giới hạn cho các bản cập nhật virus - sau đó dừng lại!
    - Có thể quét toàn bộ hệ thống tập tin hoặc giám sát động - cả hai đều tốt!
⇨ Tốt: Đáng tin cậy với mức dương tính giả thấp false-positives (độ sai lệch thấp)
⇨ Xấu: Phải biết malware, 0-days không tránh khỏi
- **Anomaly Detection: Detects unusual activity**
    - Đọc / ghi một số lượng lớn tập tin
    - Phát hiện mã được gắn với trình xử lý sự kiện (keyboard loggers)
⇨ Tốt: Có thể phát hiện ngay cả malware không xác định / 0-days
⇨ Xấu: Có xu hướng có nhiều kết quả dương tính giả false positives
- Một số kỹ thuật trốn tránh để phản hồi với AV tốt hơn:
    - **Polymorphic or encrypted viruses (Đa hình hoặc mã hoá)**
        - Code lõi được trình bày khác nhau trong các phiên bản khác nhau
        - Thường thì một phần chính (trình giải mã hoặc trình biến hình virus) có thể được nhận dạng
    - **Metamorphic viruses (siêu đa hình)**
        - Toàn bộ code thay đổi thông qua các phép biến đổi bảo toàn chức năng
        - Có thể xáo trộn các thanh ghi đã sử dụng, thêm mã vô dụng, sử dụng các hoạt động tương đương...
        - Khó phát hiện hơn nhiều!

Chương 4

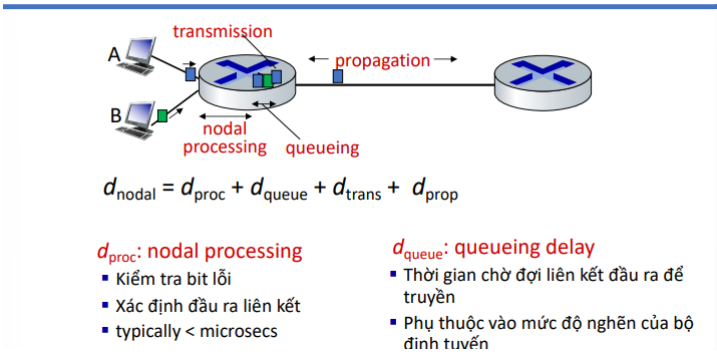
- Góc nhìn về "nuts and bolts":
    - Internet: network of networks
    - Protocols ở khắp mọi nơi, điều khiển gửi và nhận tin nhắn
    - Tiêu chuẩn Internet:
        - **RFC:** Request for Comments
        - **IETF:** Internet Engineering Task Force
- Góc nhìn về "services"

- Cơ sở hạ tầng (infrastructure) cung cấp dịch vụ cho các ứng dụng: web, streaming video, email,…
- Cung cấp giao diện lập trình (programming interface) cho các ứng dụng phân tán
  - "hooks" cho phép gửi/nhận ứng dụng để connect, sử dụng dịch vụ truyền tải internet
  - Cung cấp các tùy chọn dịch vụ, tương tự như dịch vụ bưu chính

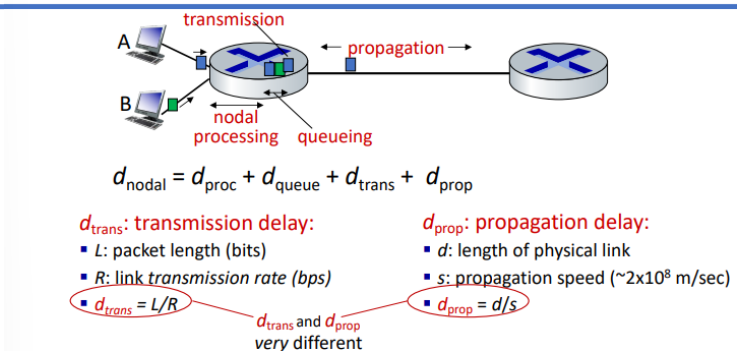
- Cấu trúc Internet:
  - **Network edge:** host gồm: client và server, máy chủ thường ở trung tâm dữ liệu.
  - **Access networks, physical media** (truy cập mạng, phương tiện vật lý): wired (dây) và wireless (không dây) communication links
  - **Network core:**
    - Bộ định tuyến được kết nối với nhau (routers)
    - Mạng lưới (network of networks)

- Hai chức năng lõi của mạng:
  - **Forwarding (switching)** – hành động local: di chuyển các packet đến từ liên kết đầu vào của bộ định tuyến đến liên kết đầu ra bộ định tuyến thích hợp.
  - **Routing** – hành động global: xác định đường dẫn nguồn – đích được thực hiện bởi các packet, các giải thuật định tuyến
- Trễ và mất gói tin:
  - Packet xếp vào hàng đợi trong bộ đệm định tuyến, chờ đến lượt truyền.
  - Độ dài hàng đợi tăng lên khi tỷ lệ đến để liên kết (tạm thời) vượt quá khả năng liên kết đầu ra
  - Mất packet xảy ra khi bộ nhớ để giữ các packet được xếp hàng chờ đầy

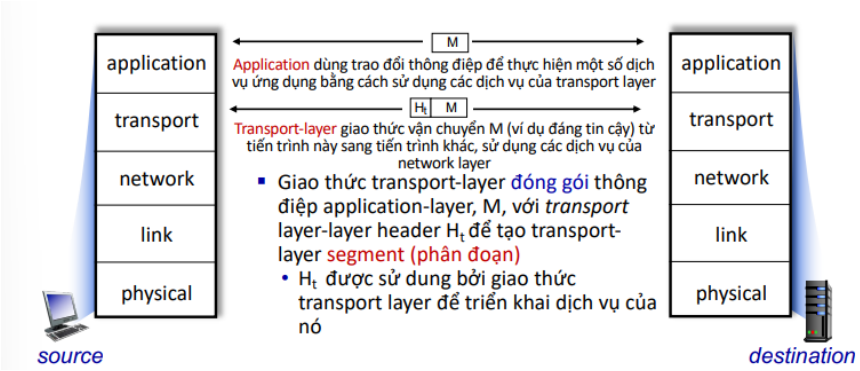
Packet delay: four sources



Packet delay: four sources

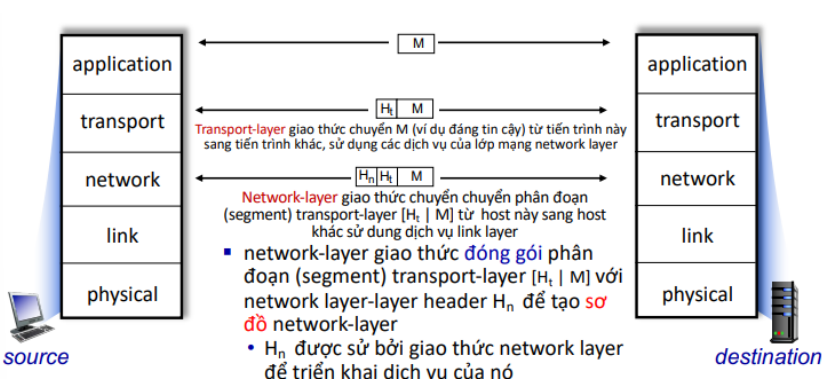


- **Phân lớp:** phương pháp tiếp cận để thiết kế / thảo luận các hệ thống phức tạp
  - Cấu trúc rõ ràng cho phép xác định, mối quan hệ của các phần của hệ thống.
  - Mô đun hóa giúp dễ dàng bảo trì, cập nhật hệ thống
- **Các tầng mạng:**
  - Application: hỗ trợ các ứng dụng mạng
    HTTP, IMAP, SMTP, DNS
  - Transport: quá trình xử lý chuyển dữ liệu
    TCP, UDP
  - Network: định tuyến các diagram dữ liệu từ nguồn đến đích
    IP, Routing protocol



- Link: dữ liệu giữa các thành phần mạng lân cận
- Physical: bits

- OSI:
  - presentation: cho phép các ứng dụng diễn giải ý nghĩa của dữ liệu, ví dụ: mã hóa, nén, các quy ước dành riêng cho máy
  - session: đồng bộ hóa, kiểm tra, phục hồi trao đổi dữ liệu



- **Bảo mật mạng:**
  - **confidentiality**: chỉ người gửi và người nhận mới "understand" nội dung thông điệp
    - Người gửi mã hoá thông điệp
    - Người nhận giải mã thông điệp
  - **authentication:** người gửi, người nhận muốn xác nhận danh tính của nhau
  - **message integrity**: người gửi, người nhận muốn đảm bảo tin nhắn không bị thay đổi (đang chuyển tiếp hoặc sau đó) mà không bị phát hiện
  - **access and availability**: dịch vụ phải có thể truy cập và khả dụng cho người dùng

- **Hijacking:** chiếm quyền điều khiển - "tiếp quản" kết nối đang diễn ra bằng cách loại bỏ người gửi hoặc người nhận, tự chèn vị trí vào
- **Denial of service (DoS):** từ chối dịch vụ - ngăn không cho người khác sử dụng dịch vụ (ví dụ: quá tải tài nguyên). Kẻ tấn công làm cho tài nguyên (máy chủ, băng thông) không có sẵn sàng cho lưu lượng truy cập hợp pháp bằng cách áp đảo tài nguyên với lưu lượng không có thật

  1. Lựa chọn mục tiêu
  2. Đột nhập vào các máy chủ trên mạng (xem botnet)
  3. Gửi các gói đến mục tiêu từ các máy chủ bị xâm phạm

- **IP spoofing:** đưa gói tin có địa chỉ nguồn sai
- **packet "sniffing":** • broadcast media (shared Ethernet, wireless) • giao diện mạng hỗn tạp đọc / ghi lại tất cả các gói (ví dụ: bao gồm cả mật khẩu!) đi qua
- **Các tuyến phòng thủ:**
  - **authentication:** chứng minh bạn là con người bạn nói
  - **confidentiality:** thông qua mã hóa
  - **integrity checks:** ký điện tử ngăn chặn / phát hiện giả mạo
  - **access restrictions:** VPN được bảo vệ bằng mật khẩu
  - **Firewalls** - "middleboxes" chuyên biệt trong truy cập và mạng lõi:
    - off-by-default: lọc các gói đến để hạn chế người gửi, người nhận, ứng dụng
    - phát hiện / phản ứng với các cuộc tấn công DOS

Chương 5

- **Recall Socket:** quá trình gửi/nhận thông điệp đến từ socket
- Socket tương tự như "door"
  - Quá trình "đẩy" thông điệp ra ngoài
  - Quá trình gửi dựa vào hạ tầng transport ở một phía khác của "door" để gửi thông điệp đến socket tại quá trình nhận
  - Two sockets involved: one on each side
- **NIC (Network Interface Card)** là thiết bị vật lý hoặc logical link giữa máy và mạng
  - Mỗi NIC có một địa chỉ MAC
  - Mọi NIC trên mạng sẽ lắng nghe tất cả các frame trên dây

    - NIC kiểm tra địa chỉ cho mọi gói tin, nếu địa chỉ khớp với địa chỉ card MAC, nó sẽ tiếp tục được sao chép vào buffer của kernel
- **Packet Sniffing** mô tả quá trình thu thập dữ liệu trực tiếp khi chúng truyền qua mạng.
- **Promiscuous Mode:**
    - Khi hệ thống trong chế độ promicuous, NIC sẽ chuyển mọi frame nhận được từ mạng đến kernel.
    - Nếu một chương trình được đăng ký sniffer với kernel, nó sẽ có thể thấy tất cả các gói tin.
    - Trong wifi, nó được gọi là Monitor Mode.
- Lọc ra các gói tin không mong muốn – **BSD Packer Filter (BPF)**
    - BPF cho phép user-progran đính kèm một filter vào socker, filter này ra lệnh cho kernel loại bỏ các gói không mong muốn.
    - BPF pseudo-code đã biên dịch có thể được đính kèm vào socket thông qua setsockopt()
      setsockopt(sock, SOL_SOCKET, SO_ATTACH_FILTER, &bpf, sizeof(bpf))
    - Khi một gói tin được nhận bởi kernel, BPF sẽ được gọi.
    - Gói được chấp nhận sẽ được đẩy lên ngăn xếp giao thức (protocol stack)
- **Raw socket approach** là một cách tiếp cận cho phép ứng dụng truy cập và gửi dữ liệu qua mạng ở mức thấp hơn so với cách tiếp cận thông thường. Thư **viện PCAP – Packet Capture API:**
    - Vẫn sử dụng các raw socket bên trong, nhưng API là tiêu chuẩn cho tất cả nền tảng. Thông tin về OS bị ẩn bởi triển khai API.
    - Được viết bằng C. Trình wrappers triển khai trên ngôn ngữ khác. Hỗ trợ trên nhiều nền tảng: Linux – libcap, Windows – Wincap và Npcap
    - Nền tảng cho nhiều công cụ: wireshark, tcpdump, scapy, nmap, snort
- **Sniffing using Scapy**: Scapy là mô-đun (hoặc chương trình) mạnh mẽ để thao tác packet
    - Packet parsing
    - Packet sending and receiving
    - Packet sniffing and spoofing
    - Adding new protocols
    - Many more applications
- **Packet spoofing**: giả mạo gói tin. Có 2 bước chính trong quá trình giả mạo gói tin: xây dựng gói tin và gửi gói tin ra ngoài.

```
#!/usr/bin/python3
from scapy.all import *

print("SENDING SPOOFED ICMP PACKET.........")
ip = IP(src="1.2.3.4", dst="93.184.216.34")  ①
icmp = ICMP()                                ②
pkt = ip/icmp                                ③
pkt.show()
send(pkt,verbose=0)                          ④
```

ICMP Spoofing

```
#!/usr/bin/python3
from scapy.all import *

print("SENDING SPOOFED UDP PACKET.........")
ip = IP(src="1.2.3.4", dst="10.0.2.69") # IP Layer
udp = UDP(sport=8888, dport=9090)       # UDP Layer
data = "Hello UDP!\n"                   # Payload
pkt = ip/udp/data      # Construct the complete packet
pkt.show()
send(pkt,verbose=0)
```

UDP Spoofing

- **Sniffing and then Spoofing:**
    - Sử dụng PCAP API đẻ capture các gói quan tâm
    - Tạo một bản sao từ gói đã capture
    - Thay thế trường data UDP bằng một thông điệp mới và hoán đổi trường nguồn và trường đích.
    - Gửi phản hồi giả mạo

- **Python + Scapy**
  - Pros: xây dựng các gói ton rất đơn giản
  - Cons: chậm hơn nhiều so với mã C
- **C Program (using raw socket)**
  - Pros: nhanh hơn nhiều
  - Cons: xây dựng các gói tin rất phức tạp
- **Hybrid Approach**
  - Sử dụng Scapy để xây dựng gói tin
  - Sử dụng C để sửa đổi một chút các gói tin và sau đó gửi các gói tin
  - Example: Kaminsky DNS attack (will be discussed
- **Byte order:** Endianness là một thuật ngữ đề cập đến thứ tự mà một mục dữ liệu multi-byte nhất định được lưu trữ trong bộ nhớ.
  - Little Endian: lưu trữ byte dữ liệu quan trong nhất ở địa chỉ cao nhất.
  - Big Endian: lưu trữ byte dữ liệu quan trọng nhất ở địa chỉ thấp nhất.
  - Các máy tính có byte order khác nhấy sẽ misunderstand lẫn nhau (chọn 1 order chung – network order)
  - Tất cả các máy tính cần chuyển đổi dữ liệu giữa host order và network order

Macro	Description
`htons()`	Convert unsigned short integer from host order to network order.
`htonl()`	Convert unsigned integer from host order to network order.
`ntohs()`	Convert unsigned short integer from network order to host order.
`ntohl()`	Convert unsigned integer from network order to host order.

Chương 6

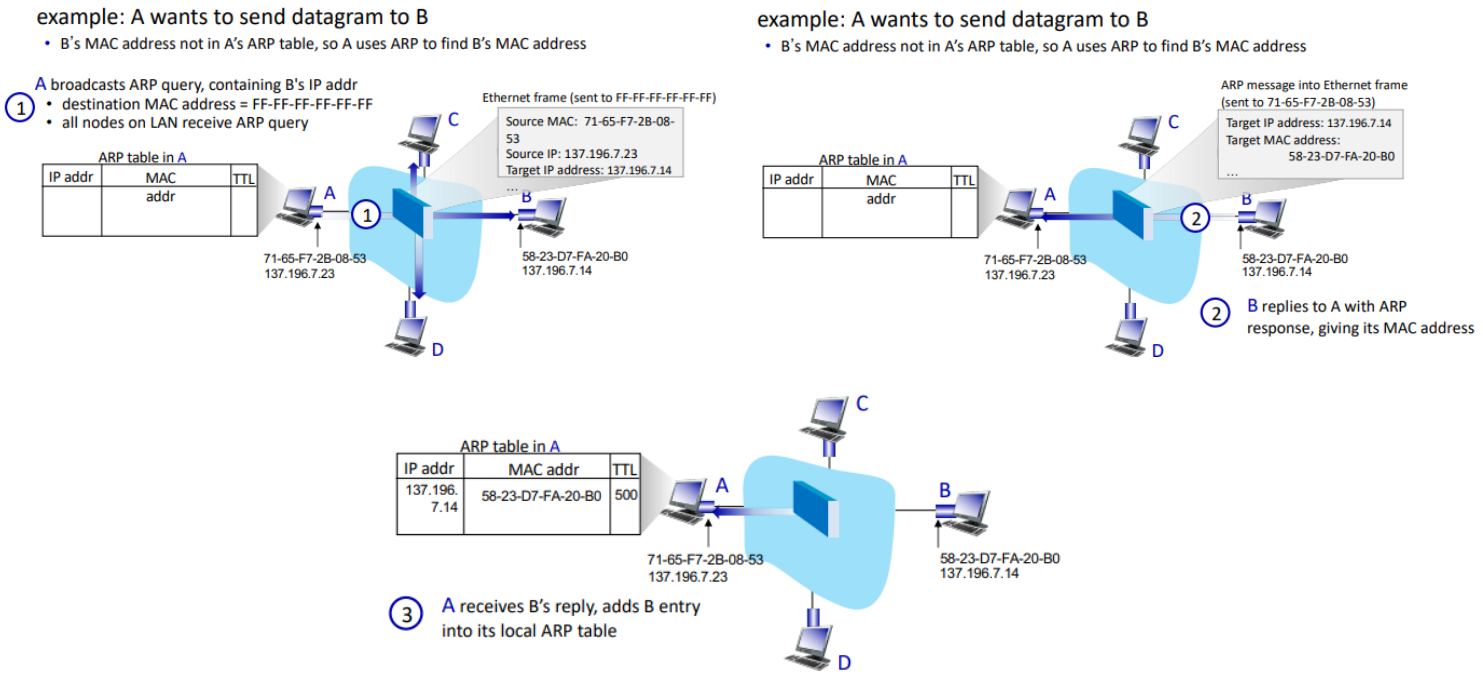
- Kết nối Internet:
  - Khi máy tính xách tay kết nối cần có địa chỉ IP riêng, địa chỉ của router đầu tiên và địa chỉ của máy chủ DNS: sử dụng DHCP
  - Yêu cầu DHCP được đóng gói trong UDP, được đóng gói trong IP, được đóng gói trong Ethernet 802.3
  - Gói Ethernet được broadcast (đích: FFFFFFFFFFFF) trên mạng LAN, được nhận tại router chạy máy chủ DHCP
  - Khung Ethernet được demuxed thành IP, UDP được demuxed thành DHCP
  - Máy chủ DHCP tạo ra DHCP ACK chứa địa chỉ IP của máy khách, địa chỉ IP của router đầu tiên cho máy khách, tên và địa chỉ IP của máy chủ DNS.
  - Quá trình đóng gói DHCP tại máy chủ DHCP, khung được chuyển tiếp (switch learning) qua LAN, giải nén tại máy khách.
  - Máy khách hiện đã có địa chỉ IP, biết tên và địa chỉ IP của máy chủ DNS, địa chỉ IP của router đầu tiên của nó.
  - Máy khách DHCP nhận phản hồi DHCP ACK.
- ARP (before DNS, before HTTP)
  - Trước khi gửi yêu cầu HTTP đến www.google.com, cần có địa chỉ IP của trang web đó thông qua quá trình DNS (Domain Name System).
  - DNS (Hệ thống Tên Miền): Trước hết, máy khách tạo một truy vấn DNS, được đóng gói trong giao thức UDP, sau đó được đóng gói trong giao thức IP, và tiếp theo là đóng gói trong giao thức Ethernet.
  - Để gửi gói tin đến router, cần phải có địa chỉ MAC của giao diện router, và điều này được thực hiện thông qua ARP (Giao thức Địa chỉ Định danh).
  - ARP (Giao thức Địa chỉ Định danh): Một truy vấn ARP được phát sóng, được router nhận và trả lời bằng một ARP reply cung cấp địa chỉ MAC của giao diện router.
  - Bây giờ, máy khách biết đến địa chỉ MAC của router, vì vậy nó có thể gửi gói tin chứa truy vấn DNS đến router.

- DNS
  - Gói tin IP chứa truy vấn DNS được chuyển tiếp qua switch mạng LAN từ máy khách đến router ở điểm đầu tiên.
  - Gói tin IP được chuyển tiếp từ mạng trường học vào mạng Comcast, được định tuyến (bảng định tuyến được tạo ra bởi các giao thức định tuyến như RIP, OSPF, IS-IS và/hoặc BGP) đến máy chủ DNS.
  - Gói tin được giải đã đến máy chủ DNS.
  - DNS trả lời lại máy khách với địa chỉ IP của www.google.com.
- TCP connection carrying HTTP
  - Để gửi yêu cầu HTTP, máy khách trước tiên mở một ổ cắm TCP tới máy chủ web.
  - Đoạn mã TCP SYN (bước 1 trong bắt tay 3 bước của TCP) được định tuyến qua các miền đến máy chủ web.
  - Máy chủ web đáp lại bằng TCP SYNACK (bước 2 trong bắt tay 3 bước của TCP).
  - Kết nối TCP được thiết lập!
- HTTP request/reply
  - Yêu cầu HTTP được gửi vào TCP socket.
  - Gói tin IP chứa yêu cầu HTTP được định tuyến đến www.google.com.
  - Máy chủ web đáp lại bằng phản hồi HTTP (chứa trang web).
  - Gói tin IP chứa phản hồi HTTP được định tuyến trở lại máy khách.
- The Data link layer – Interface communication
  - **Bên gửi:**
    - Đóng gói gói tin vào khung (frame).
    - Thêm các bit kiểm tra lỗi, truyền dữ liệu đáng tin cậy, kiểm soát luồng, v.v.
  - **Bên nhận:**
    - Kiểm tra lỗi, truyền dữ liệu đáng tin cậy, kiểm soát luồng, v.v.
    - Trích xuất datagram và chuyển đến tầng trên ở bên nhận.
- **Địa chỉ MAC** (hoặc địa chỉ mạng LAN, vật lý, Ethernet):
  - Chức năng: sử dụng "cục bộ" để lấy khung từ một giao diện đến một giao diện khác được kết nối vật lý (cùng mạng con, theo ý nghĩa của địa chỉ IP).
  - Địa chỉ MAC 48-bit (đối với hầu hết các LAN) được ghi vào ROM của NIC, cũng đôi khi có thể được cài đặt bằng phần mềm.
  - Phân bổ địa chỉ MAC được quản lý bởi IEEE.
  - Địa chỉ MAC là duy nhất
  - Địa chỉ MAC là địa chỉ phẳng: có thể di chuyển giao diện từ một Mạng LAN sang Mạng LAN khác.
- **Địa chỉ IP 32-bit:**
  - Địa chỉ tầng mạng cho interface.
  - Sử dụng cho việc chuyển tiếp tầng 3 (tầng mạng).
  - Lưu ý rằng IP không di động: phụ thuộc vào mạng con IP mà nút được kết nối.
- 0x0806: ARP
- 0x0800: IPv4

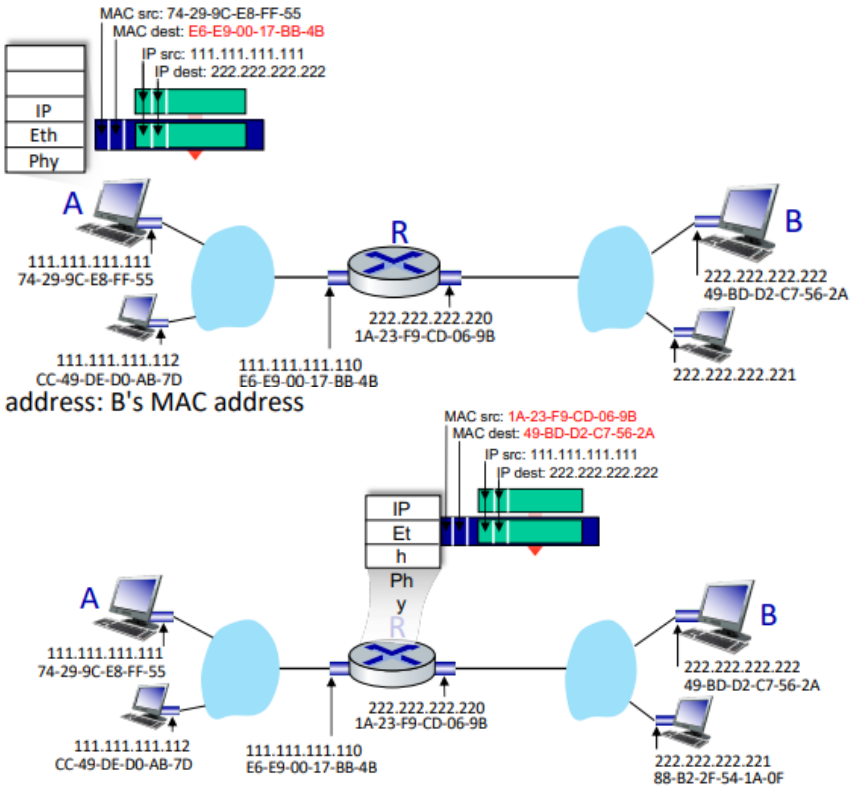
Câu hỏi: Làm thế nào để xác định địa chỉ MAC của interface khi biết địa chỉ IP của nó?

=> Bảng ARP: mỗi node IP (h, router) trên Mạng LAN có một bảng địa chỉ IP/MAC < Địa chỉ IP; Địa chỉ MAC; TTL>

- ARP có một bộ nhớ đệm, nghĩa là nó không cần phải hỏi địa chỉ MAC mỗi lần.
- ARP là giao thức không lưu trạng thái.
- Trên hệ thống dựa trên Linux:
  - arp –n: hiển thị bộ nhớ đệm ARP.
  - arp –d: xóa một mục ARP.



- Định tuyến đến một mạng con khác:
  - A tạo ra gói tin IP với địa chỉ nguồn là A, địa chỉ đích là B.
  - A tạo ra link layer frame chứa gói tin IP từ A đến B với địa chỉ MAC của R là đích đến của frame.
  - Frame được gửi từ A đến R.
  - Frame được nhận tại R, datagram được loại bỏ và chuyển lên tầng IP.
  - R xác định outgoing interface, chuyển gói tin với địa chỉ nguồn IP là A, địa chỉ đích là B lên link layer.
  - R tạo ra link layer frame chứa gói tin IP từ A đến B. Địa chỉ đích của frame: Địa chỉ MAC của B.
  - Truyền link layer frame
  - B nhận frame, trích xuất datagram, đưa lên ngăn xếp giao thức tới IP.



- **Đầu độc ARP cache**:
  Tưởng tượng bạn là Alice (A) và bạn muốn làm cho máy tính của Bob (B) đặt thông tin của nạn nhân Charlie (C) vào bộ nhớ đệm ARP của mình. Trong tình huống này:
  1. Bạn gửi một yêu cầu ping giả mạo đến máy tính của Bob, nhưng giả vờ là máy tính của Charlie.
  2. Bob nhận được yêu cầu ping và tưởng rằng nó đến từ máy tính của Charlie.
  3. Bob trả lời yêu cầu ping, nhưng đối với máy tính của Bob, địa chỉ MAC của Charlie cần được biết để gửi trả lời.
  4. Do đó, Bob đặt thông tin về máy tính của Charlie vào bộ nhớ đệm ARP của mình.
  5. Kết quả là máy tính của Bob bây giờ có một mục mới trong bộ nhớ đệm ARP với thông tin về máy tính của Charlie, mặc dù thực tế là yêu cầu ping đã được gửi từ máy tính của Alice. Điều này làm cho Bob tin rằng địa chỉ MAC của Charlie là địa chỉ MAC của máy tính của Alice.
- Cách thức hoạt động của Telnet khác với Netcat.
  - **Netcat:** Bất kỳ điều gì bạn nhập, trước khi nhấn Enter, tất cả trong cùng một dòng sẽ được gửi trong một gói tin TCP.
  - **Telnet:** Mỗi khi bạn nhập một ký tự, ký tự đó sẽ được gửi đi, thường là trong một gói tin TCP duy nhất, và máy chủ sẽ phản hồi lại. Đó là cách ký tự bạn nhập được hiển thị trên phía máy khách. Nó không hiển thị ngay lập tức mỗi khi bạn nhập; thực sự nó mất một chuyến đi và hiển thị.

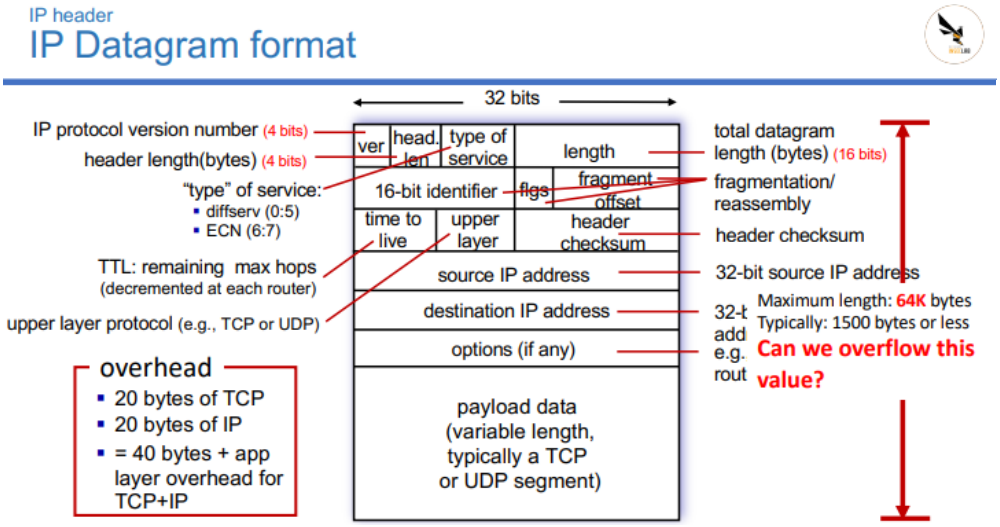
Chương 7

- Router:
  - Kiểm tra các trường header trong tất cả các gói tin IP đi qua nó.
  - Di chuyển gói tin từ input port đến output port để chuyển gói tin từ điểm này đến điểm kia.
- Gửi segment từ host này đến host khác:
  - Người gửi: đóng gói các segment thành các gói tin và chuyển đến tầng liên kết.
  - Người nhận: chuyển giao tất cả các segment đến transport layer protocol.
- Hai chức năng chính của tầng mạng (Network layer)
  - **Chuyển tiếp – forwarding**: di chuyển gói tin từ đầu vào của một router đến đầu ra của một router tương ứng.
  - **Định tuyến – routing:** xác định tuyến đường mà các gói tin sẽ đi từ nguồn đến đích
- **Data plane – mặt dữ liệu:**
  - Chức năng cục bộ, từng router.
  - Xác định cách gói tin đi từ router input port đến router output port.
- **Control plane – mặt kiểm soát:**
  - Network – wide logic
  - Xác định cách gói tin được định tuyến qua các router dọc theo đường dẫn từ nguồn đến đích.
  - Hai phương pháp kiểm soát control – plane:
    - Thuật toán định tuyến truyền thống – **traditional routing algorithms**: được triển khai trong các router.
    - Mạng định nghĩa phần mềm – **software-defined networking (SDN):** được triển khai trên các server từ xa.
  - Control – plane tại mỗi router: các thành phần thuật toán định tuyến riêng lẻ trong từng (in each and every) router tương tác trên control – plane.
  - Control – plane của SDN: bộ điều khiển từ xa tính toán và cài đặt bảng chuyển tiếp trong các router.
- Các liên kết mạng có **MTU** – kích thước truyền tối đa: là kích thước frame lớn nhất có thể truyền => phân mảnh.
- **Ping of Death (PoD) attack**: tạo ra một IP packet lớn hơn 65 536 bytes (64KB)

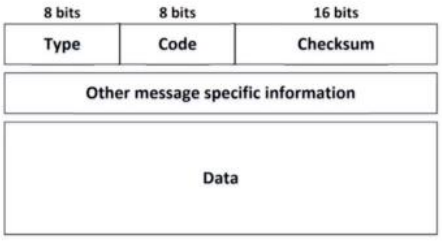
Câu hỏi: Có thể sử dụng một lượng băng thông nhỏ để làm gián đoạn một lượng tài nguyên đáng kể trên máy mục tiêu không?

=> Sử dụng DoS

- **Smuft Attack** là một loại tấn công từ chối dịch vụ DoS, kẻ tấn công gửi một số lượng lớn yêu cầu ICMP Echo Request (ping) đến một địa chỉ đích, thường là địa chỉ broadcast, sử dụng địa chỉ nguồn bị giả mạo.



- Routing Tables Configured:
  - For Router:
    - Routing protocols (OSPF)
    - Attacks on routing protocols (BGP)
  - For hosts (tiny routing table)
    - DHCP
    - Default routers
    - Manual configuration (static route)
    - ICMP redirect messages
- **Symmetric routing**: khii R nhận một gói tin từ cổng A, nó sẽ thực hiện một quá trình reverse lookup, nếu đường trả về qua cùng một cổng => cho phép.
- **Asymmetric routing**: ngược lại => drop
- **ICMP – Internet Control Message Protocol**: được sử dụng với các host và router để trao đổi thông tin cấp mạng. Các thông điệp ICMP được chuyển trong các gói tin IP và mỗi thông điệp ICMP bao gồm loại, mã, cùng với 8 byte đầu tiên của gói tin IP gây ra lỗi. Mục đích chính của ICMP:
  - Error Reporting: khi các host, network, port, protocol, time không thể đạt được.
  - Control Message:
    - Echo request / reply (used py ping)
    - Redirect
    - Timestamp request / reply
    - Router advertisement / solicitation



Type	Code	description
0	0	echo reply (ping)
3	0	dest. network unreachable
3	1	dest host unreachable
3	2	dest protocol unreachable
3	3	dest port unreachable
3	6	dest network unknown
3	7	dest host unknown
4	0	source quench (congestion control - not used)
8	0	echo request (ping)
9	0	route advertisement
10	0	router discovery
11	0	TTL expired
12	0	bad IP header

Câu hỏi: Có thể thực hiện lệnh chuyển hướng ICMP từ một máy tính từ xa không?

=> Không. Reverse Path Filtering (RPF) tại router sẽ loại bỏ chúng.

Câu hỏi: Có thể sử dụng các cuộc tấn công chuyển hướng ICMP để chuyển hướng đến một máy tính từ xa không?

=> Không. Khi máy A nhận được lệnh chuyển hướng ICMP, nó sẽ kiểm tra xem cổng điều hướng có ở trên cùng một mạng không. Nếu không phải thì bỏ qua.

Chương 8

- Transport service and protocol (UDP, TCP):
  - Cung cấp giao tiếp logic giữa các tiến trình ứng dụng đang chạy trên các host khác nhau.
  - Các hành động của các transport protocol trong các end system:
    - Người gửi: chia nhỏ các thông điệp ứng dụng thành các segment và chuyển đến tầng mạng.
    - Người nhận: tập hợp các segment thành các thông điệp và chuyển đến tầng ứng dụng.
- Port number:
  - ftp (20, 21), ssh (22), telnet (23), smtp (25), DNS (53), http (80), https (443)
  - openVPN (1194), Microsoft SQL server (1433), Docker (2375-2377)
  - Cổng riêng tư (Private ports): 49152 – 65535 (Source ports)

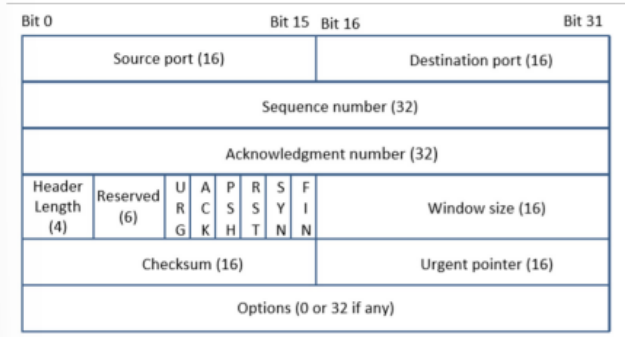
	TCP	UDP
Kết nối	Connection – oriented Là một mô hình giao tiếp trong đó có sự thiết lập trước một kết nối trước khi truyền dữ liệu.	Connection – less Là một mô hình giao tiếp trong đó không có sự thiết lập trước kỳ vọng trước khi truyền dữ liệu.
Ranh giới giữa các gói tin – Packet Boundary	Stream – based (dựa trên luồng) truyền dữ liệu như một luồng liên tục mà không có sự phân biệt giữa các gói tin hoặc biên giới cụ thể.	Maintain boundary (duy trì biên giới) Dữ liệu được chia thành các gói tin riêng lẻ và mỗi gói tin có ranh giới riêng biệt và rõ ràng.
Độ tin cậy	Có	Không
Duy trì thứ tự	Có	Không
Tốc độ	Chậm hơn	Nhanh hơn
broadcast	Không	Có
	Providing a Virtual Connection Maintaining Order Reliability Flow Control	DNS Protocol Video/Audio Streaming, Skype, Zoom Note: Netflix and YouTube use TCP Real-Time Applications VPN Tunnel (OpenVPN)

- **UDP – User Datagram Protocol:**
  - UDP Ping-pong effect: khi một dịch vụ hoặc ứng dụng phát ra một UDP reply mà không quan tâm đến gói tin đầu vào là gì, việc thiết lập cổng nguồn và đích của gói tin UDP để tạo ra hiệu ứng ping-pong. Điều này làm cho các gói tin ping (yêu cầu) và pong (trả lời) liên tục được gửi giữa nguồn và đích mà không có điểm dừng, tạo ra một chuỗi tương tác không kết thúc giữa chúng.
  - UDP Amplification Attack – tấn công gia tăng: các ứng dụng phản hồi bằng các gói tin lớn đối với các yêu cầu nhỏ. Các máy chủ có thể bị tấn công bằng cách sử dụng các ứng dụng này như bộ khuếch đại, với việc làm giả IP nguồn.
  - UDP attack – bottom line: càng phức tạp một giao thức, bạn có thể tạo ra những cuộc tấn công phức tạp hơn.
- **TCP – Transmission Control Protocol:** là một giao thức cốt lõi trong bộ giao thức Internet
  - Đặt ở tầng IP, tầng vận chuyển
  - **Tạo một TCP server program:**
    - **Bước 1:** Tạo một socket. Tương tự như Chương trình Client.
    - **Bước 2:** Liên kết với một số cổng. Một ứng dụng giao tiếp với các ứng dụng khác qua mạng cần đăng ký một số cổng trên máy tính của mình. Khi gói tin đến, hệ điều hành biết được ứng dụng nào là người nhận dựa trên số cổng. Server cần thông báo cho hệ điều hành về số cổng mà nó đang sử dụng. Điều này được thực hiện thông qua lệnh hệ thống bind().
    - **Bước 3:** Nghe kết nối.
      Sau khi socket được thiết lập, các chương trình TCP gọi listen() để đợi kết nối.

Nó thông báo cho hệ thống rằng nó sẵn sàng nhận yêu cầu kết nối.
Khi một yêu cầu kết nối được nhận, hệ điều hành sẽ thực hiện bước 3-way handshake để thiết lập kết nối.
Kết nối đã thiết lập được đặt trong hàng đợi, chờ ứng dụng xử lý nó.
Đối số thứ hai (5) chỉ định số kết nối có thể được lưu trữ trong hàng đợi.

- **Bước 4:** Chấp nhận một yêu cầu kết nối.
  Sau khi kết nối được thiết lập, một ứng dụng cần "chấp nhận" kết nối trước khi có thể truy cập nó. Lệnh hệ thống accept() trích xuất yêu cầu kết nối đầu tiên từ hàng đợi, tạo một socket mới và trả về bộ chỉ mục tập tin liên quan đến socket.
- **Bước 5**: Gửi và Nhận dữ liệu
  Sau khi kết nối được thiết lập và được chấp nhận, cả hai bên có thể gửi và nhận dữ liệu bằng cách sử dụng socket mới này.

- Để thiết lập **multiple connection**:
  - Lệnh hệ thống fork() tạo một tiến trình mới bằng cách nhân đôi tiến trình đang gọi.
  - Khi thành công, ID của tiến trình con được trả về trong tiến trình cha và là 0 trong tiến trình con.
- **TCP data transmission**
  - Sau khi kết nối được thiết lập, hệ điều hành cấp hai bộ đệm ở mỗi đầu, một cho việc gửi dữ liệu (bộ đệm gửi) và bộ đệm nhận (bộ đệm nhận).
  - Khi một ứng dụng cần gửi dữ liệu ra, nó đặt dữ liệu vào bộ đệm gửi TCP.
  - Mỗi octet trong bộ đệm gửi có một trường số thứ tự trong tiêu đề, chỉ số thứ tự của các gói tin. Ở đầu nhận, các số thứ tự này được sử dụng để đặt dữ liệu ở vị trí đúng bên trong bộ đệm nhận.
  - Khi dữ liệu được đặt trong bộ đệm nhận, chúng được hợp nhất thành một dòng dữ liệu đơn.
  - Ứng dụng đọc từ bộ đệm nhận. Nếu không có dữ liệu nào có sẵn, thường sẽ bị chặn. Nó sẽ được mở chặn khi có đủ dữ liệu để đọc.
  - Bên nhận thông báo cho bên gửi về việc nhận dữ liệu bằng cách sử dụng gói xác nhận.
- **TCP header:**
  - Port nguồn và Port đích (16 bit mỗi port)
  - Sequence number (32 bit): Chỉ định số thứ tự của octet đầu tiên trong đoạn TCP.
  - Acknowledgement number (32 bit): Chứa giá trị của số thứ tự tiếp theo được mong đợi.
  - Header length (4 bit)
  - Reverse (6 bit)
  - Code bits (6 bit): Có sáu bit mã, bao gồm SYN, FIN, ACK, RST, PSH và URG.
  - Window (16 bit): Mục đích của trường này là để kiểm soát luồng.
  - Checksum (16 bit): Tổng kiểm tra được tính bằng cách sử dụng một phần của tiêu đề IP, tiêu đề TCP và dữ liệu TCP.
  - Urgent Pointer (16 bit): Nếu bit mã URG được đặt, phần đầu của dữ liệu chứa dữ liệu khẩn cấp. Con trỏ khẩn cấp chỉ định nơi kết thúc dữ liệu khẩn cấp và dữ liệu TCP bình thường bắt đầu. Dữ liệu khẩn cấp được sử dụng cho mục đích ưu tiên vì chúng không đợi trong hàng đợi bộ đệm nhận, và sẽ được gửi ngay lập tức đến ứng dụng.
  - Options (0-320 bit, chia hết cho 32)



- TCP bắt tay 3 bước:
  - Gói SYN: Khách hàng gửi một gói đặc biệt gọi là gói SYN đến máy chủ, sử dụng một số được tạo ngẫu nhiên x làm số thứ tự của nó.
  - Gói SYN-ACK: Khi nhận được nó, máy chủ gửi một gói phản hồi bằng cách sử dụng số được tạo ngẫu nhiên y làm số thứ tự của nó.
  - Gói ACK: Khách hàng gửi gói ACK để kết thúc bước bắt tay.
- TCB – Transmission Control Block:
  - Khi máy chủ nhận được gói SYN ban đầu, nó sử dụng TCB để lưu trữ thông tin về kết nối.
  - Sau khi máy chủ nhận được gói ACK, nó sẽ lấy TCB này ra khỏi hàng đợi và lưu trữ ở một nơi khác.
  - Nếu ACK không đến, máy chủ sẽ gửi lại gói SYN+ACK. TCB sẽ cuối cùng được loại bỏ sau một khoảng thời gian nhất định.
- SYN flooding attack:
  - Ý tưởng: Để làm đầy hàng đợi lưu trữ các kết nối bán mở sao cho không còn chỗ để lưu trữ TCB cho bất kỳ kết nối bán mở mới nào, nói cách khác, máy chủ không thể chấp nhận bất kỳ gói SYN mới nào.
  - Các bước để đạt được điều này: Liên tục gửi rất nhiều gói SYN đến máy chủ. Điều này tiêu thụ không gian trong hàng đợi bằng cách chèn bản ghi TCB.
  - Khi làm tràn máy chủ bằng các gói SYN, chúng ta cần sử dụng địa chỉ IP nguồn ngẫu nhiên; nếu không, các cuộc tấn công có thể bị chặn bởi tường lửa.
  - Các gói SYN+ACK được gửi bởi máy chủ có thể bị loại bỏ vì địa chỉ IP giả mạo có thể không được gán cho bất kỳ máy nào.
  - Nếu nó đến đúng máy tồn tại, một gói RST sẽ được gửi đi và TCB sẽ bị loại khỏi hàng đợi.
  - Vì lựa chọn thứ hai ít có khả năng xảy ra, các bản ghi TCB chủ yếu sẽ ở lại trong hàng đợi. Điều này gây ra Tấn công SYN Flooding.
- SYN cookie
  - Sau khi một máy chủ nhận được gói SYN, nó tính toán một giá trị băm có chìa khóa (H) từ thông tin trong gói sử dụng một khóa bí mật chỉ biết đến máy chủ. Giá trị băm này (H) được gửi đến máy khách như số thứ tự ban đầu từ máy chủ. H được gọi là cookie SYN.
  - Máy chủ sẽ không lưu trữ kết nối bán mở trong hàng đợi của mình. • Nếu máy khách là một kẻ tấn công, H sẽ không đến được kẻ tấn công.
  - Nếu máy khách không phải là kẻ tấn công, nó gửi H+1 trong trường xác nhận.
  - Máy chủ kiểm tra xem số trong trường xác nhận có hợp lệ hay không bằng cách tính toán lại cookie.
- TCP reset Attack:
  - Để ngắt kết nối TCP:
    - A gửi một gói "FIN" đến B.
    - B trả lời bằng một gói "ACK". Điều này đóng kết nối truyền thông từ A đến B.
    - Bây giờ, B gửi một gói "FIN" đến A và A trả lời bằng "ACK".
  - Sử dụng cờ Reset:
    - Một trong các bên gửi gói RST để ngay lập tức ngắt kết nối.
    - Không cần phải đợi ACK.