TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI TP. HỒ CHÍ MINH

KHOA CÔNG NGHỆ THÔNG TIN

AN TOÀN THÔNG TIN-
INFORMATION SECURITY

CHƯƠNG 2
CÁC LỖ HỔNG TRONG BẢO MẬT VÀ CÁC ĐIỂM YẾU CỦA MẠNG

Giảng viên: TS. Trần Thế Vinh

I. KIẾN THỨC CHUNG VỀ LỖ HỔNG BẢO MẬT

Lỗ hổng trong bảo mật máy tính là gì?:

Lỗ hổng hệ thống máy tính là một lỗ hổng, điểm yếu trong hệ thống hoặc mạng có thể bị khai thác để gây ra thiệt hại hay cho phép kẻ tấn công thao túng hệ thống theo một cách nào đó.

Các loại lỗ hổng bảo mật:

- Lỗ hổng hệ điều hành:

Đây là những lỗ hổng trong một hệ điều hành cụ thể mà tin tặc có thể khai thác để dành quyền truy cập vào nội dung mà hệ điều hành được cài đặt trên đó.

- Lỗ hổng mạng:

Đây là những sự cố xảy ra với phần cứng hoặc phần mềm của mạng khiến mạng có thể bị xâm nhập trái phép từ bên ngoài. Ví dụ: điểm truy cập Wifi không an toàn, tường lửa được cấu hình kém.

- Lỗ hổng quy trình:

Một số lỗ hổng có thể được tao ra bởi các biện pháp kiểm soát quy trình cụ thể. Ví dụ: việc sử dụng mật khẩu yếu

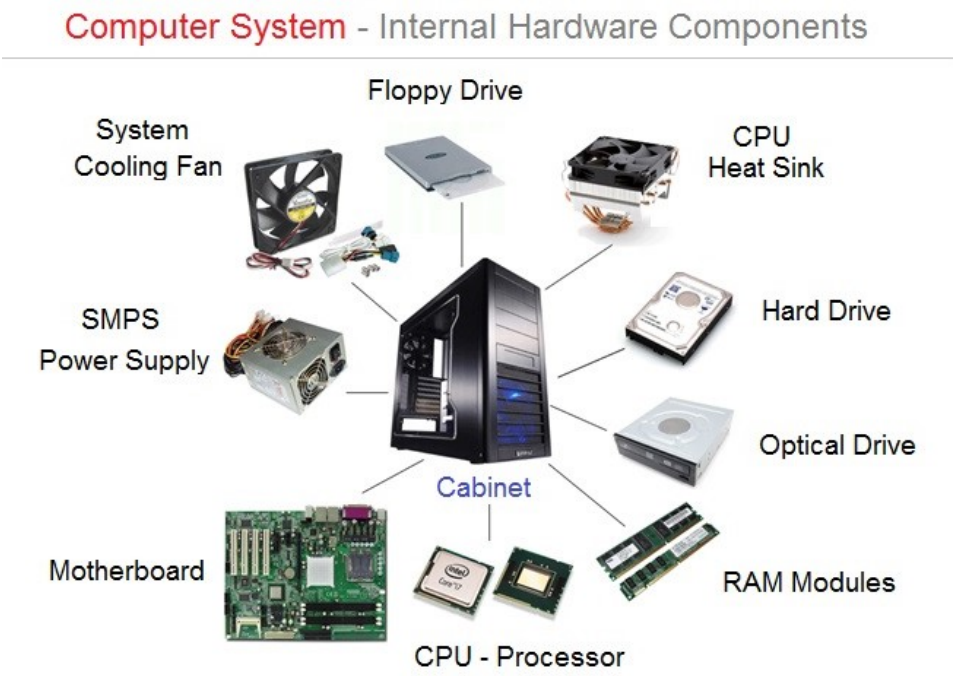
- Lỗ hổng từ người dùng:

Liên kết yếu nhất trong nhiều kiến trúc an ninh mạng là yếu tố con người. Lỗi người dùng có thể dễ dàng làm lộ dữ liệu nhạy cảm, tạo điểm truy cập có thể khai thác cho kẻ tấn công hoặc làm gián đoạn hệ thống

I. KIẾN THỨC CHUNG VỀ LỖ HỔNG BẢO MẬT

Các thành phần của hệ thống máy tính:

Hệ thống phần cứng



- **Hệ thống phần mềm**
  - Hệ điều hành

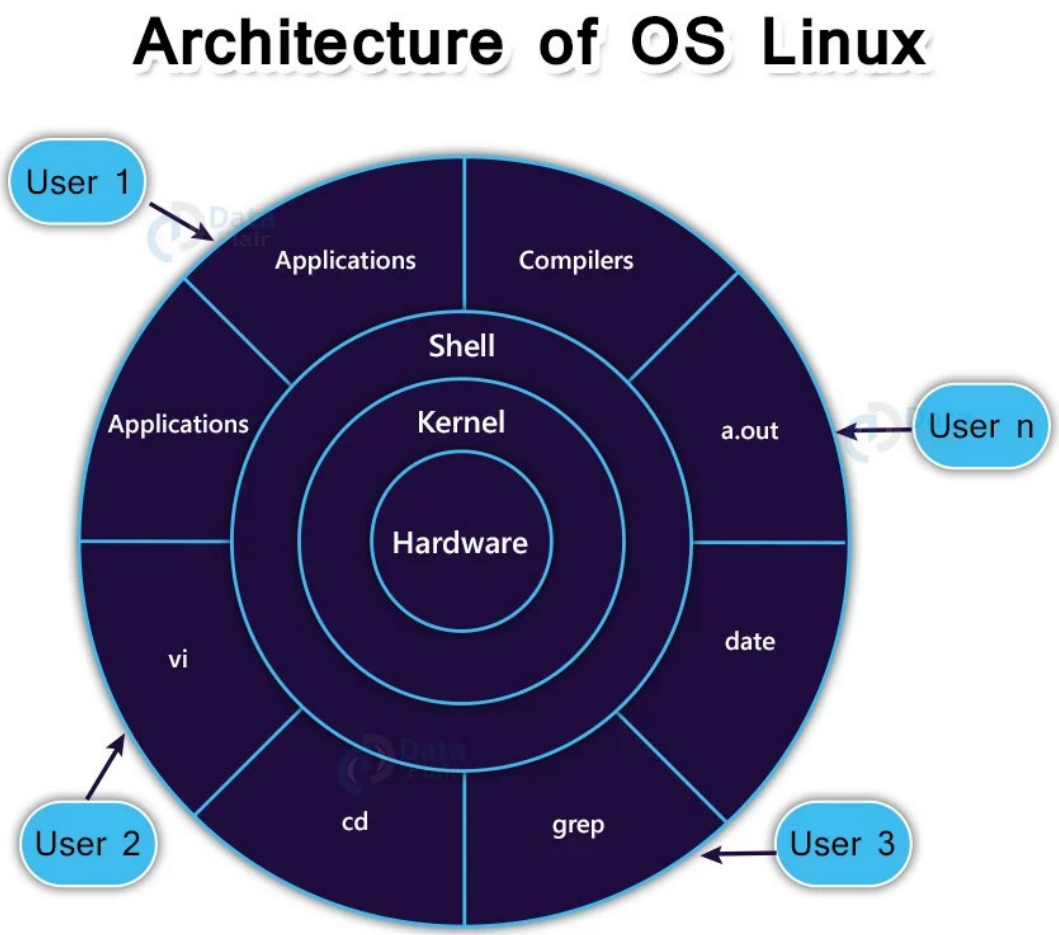
– Nhân hệ điều hành, các trình điều khiển thiết bị
– Các trình cung cấp dịch vụ, tiện ích,…

  - Các phần mềm ứng dụng

– Các dịch vụ (máy chủ web, CSDL, DNS,...)
– Trình duyệt web, các ứng dụng giao tiếp,…
– Các bộ ứng dụng văn phòng, lập trình.

I. KIẾN THỨC CHUNG VỀ LỖ HỔNG BẢO MẬT

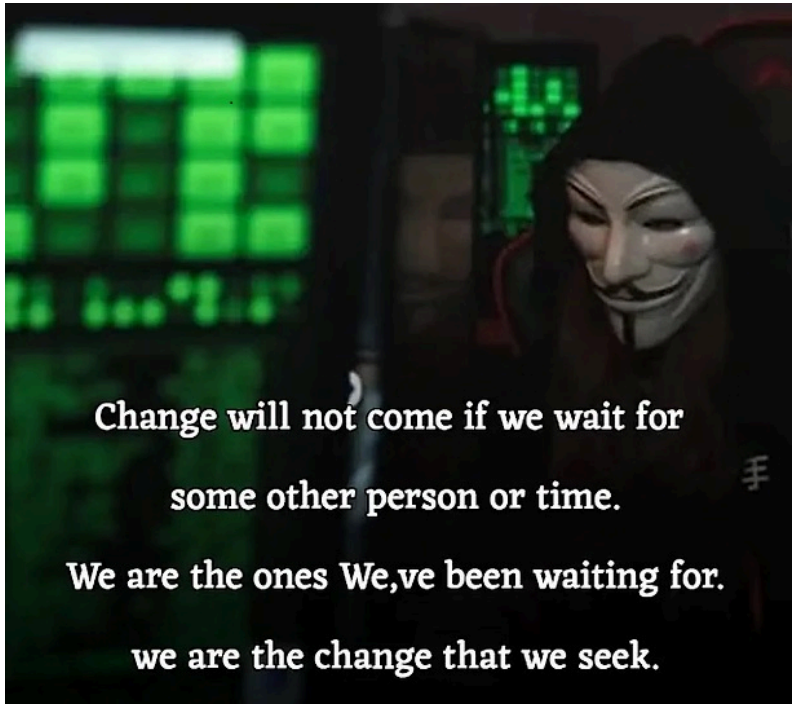
Mô hình hệ điều hành Unix/Linux,



I. KIẾN THỨC CHUNG VỀ LỖ HỔNG BẢO MẬT

Điểm yếu của hệ thống máy tính:
Là các lỗi hay các khiếm khuyết(thiết kế, cài đặt, phần cứng, phần mềm) tồn tại trong hệ thống

- DDoS attack
- Pharming
- Effects:
  - Hủy dữ liệu (Data destruction)
  - Thao tác dữ liệu (Data manipulation)
  - Sửa đổi dữ liệu (Data modification)
  - Lấy cắp dữ liệu (Data theft)
  - Lấy cắp danh tính (Indentity theft)
- Lỗ hổng (Vulnerabilities):
  - Lỗ hổng vật lý
  - Mật khẩu yếu
  - Mô hình không an toàn
  - Phần mềm không an toàn
  - Thảm họa thiên nhiên



I. KIẾN THỨC CHUNG VỀ LỖ HỔNG BẢO MẬT

Lỗ hổng bảo mật là một điểm yếu tồn tại trong hệ thống cho phép kẻ tấn công khai thác gây tổn hại đến:

- Tính bảo mật (confidentiality)
- Tính toàn vẹn (integrity)
- Tính sẵn sàng (availability).

Phụ thuộc vào khả năng bị khai thác, các lỗ hổng bảo mật có mức độ nghiêm trọng khác nhau.

I. KIẾN THỨC CHUNG VỀ LỖ HỔNG BẢO MẬT

Tính bảo mật (confidentiality)

• Chỉ những người có thẩm được phép truy nhập đến thông tin/hệ thống;
• Hacker có thể lợi dụng lỗi an ninh trong hệ thống để đột nhập trái phép vào hệ thống;

• Ví dụ:
– Một lỗ hổng trên hệ thống cho phép kẻ tấn công lấy được thông tin mật của tổ chức, công ty

I. KIẾN THỨC CHUNG VỀ LỖ HỔNG BẢO MẬT

Tính toàn vẹn (integrity):

- Hệ thống chỉ có thể được sửa đổi bởi những người dùng có thẩm quyền.
- Tính toàn vẹn liên quan đến tính hợp lệ (validity) và chính xác (accuracy) của hệ thống.
- Hacker có thể lợi dụng lỗ hổng trong hệ thống để đột nhập sửa đổi thông tin hệ thống;

Ví dụ:

- Trong hệ thống kiểm soát truy cập, người quan trị mới có quyền thay đổi quyền truy cập đến mọi thông tin/hệ thống.

- Một lỗ hổng trong hệ thống cho phép hacker tấn công và chiếm quyền kiểm soát truy cập.

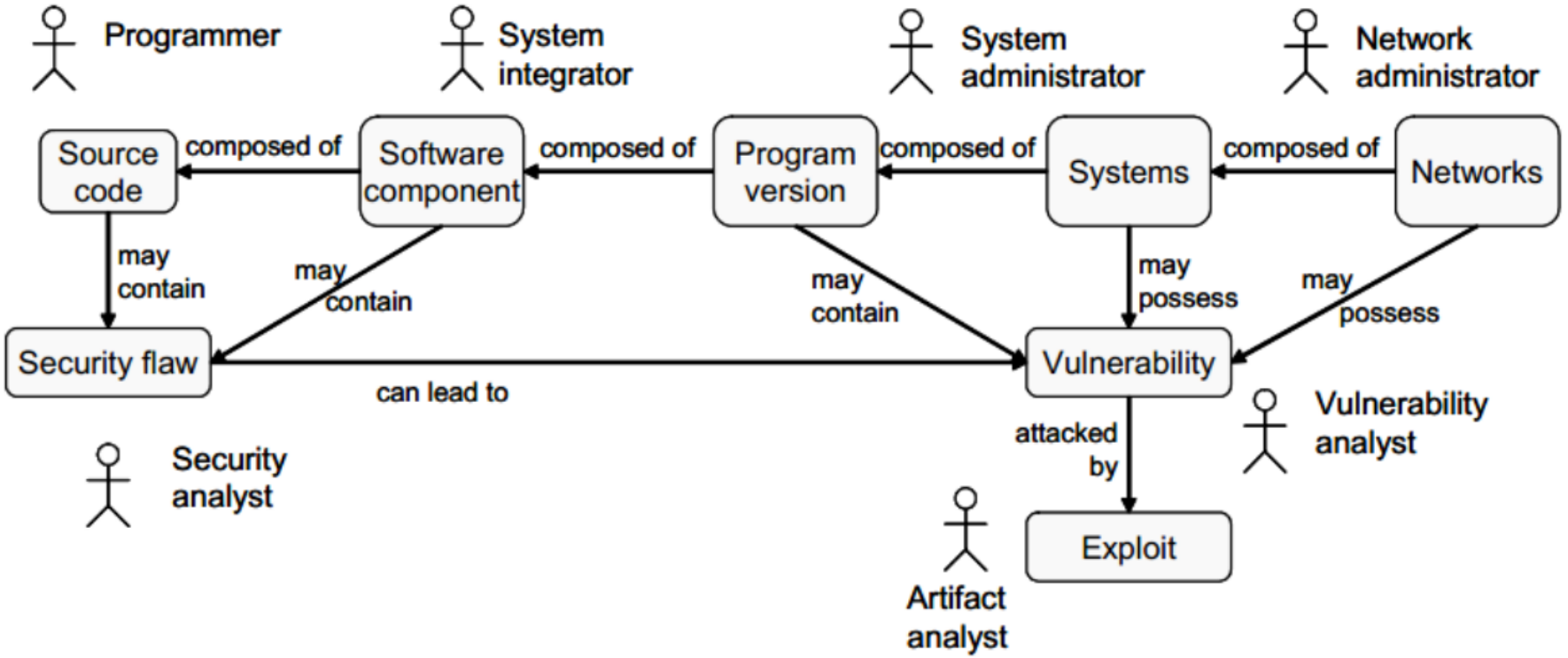
I. KIẾN THỨC CHUNG VỀ LỖ HỔNG BẢO MẬT

Tính Sẵn sàng (availability):

• Đảm bảo khả năng truy nhập đến thông tin/hệ thống cho người dụng hợp pháp;
• Hacker có thể lợi dụng lỗ hổng để ngăn cản hoặc gây khó khăn cho người dùng truy cập vào hệ thống.
• Ví dụ:
– Hacker sử dụng tấn công từ chối dịch vụ để ngăn người dùng truy cập vào hệ thống.

I. KIẾN THỨC CHUNG VỀ LỖ HỔNG BẢO MẬT

Mô hình các quan hệ giữa các đối tượng và vai trò trong hệ thống:

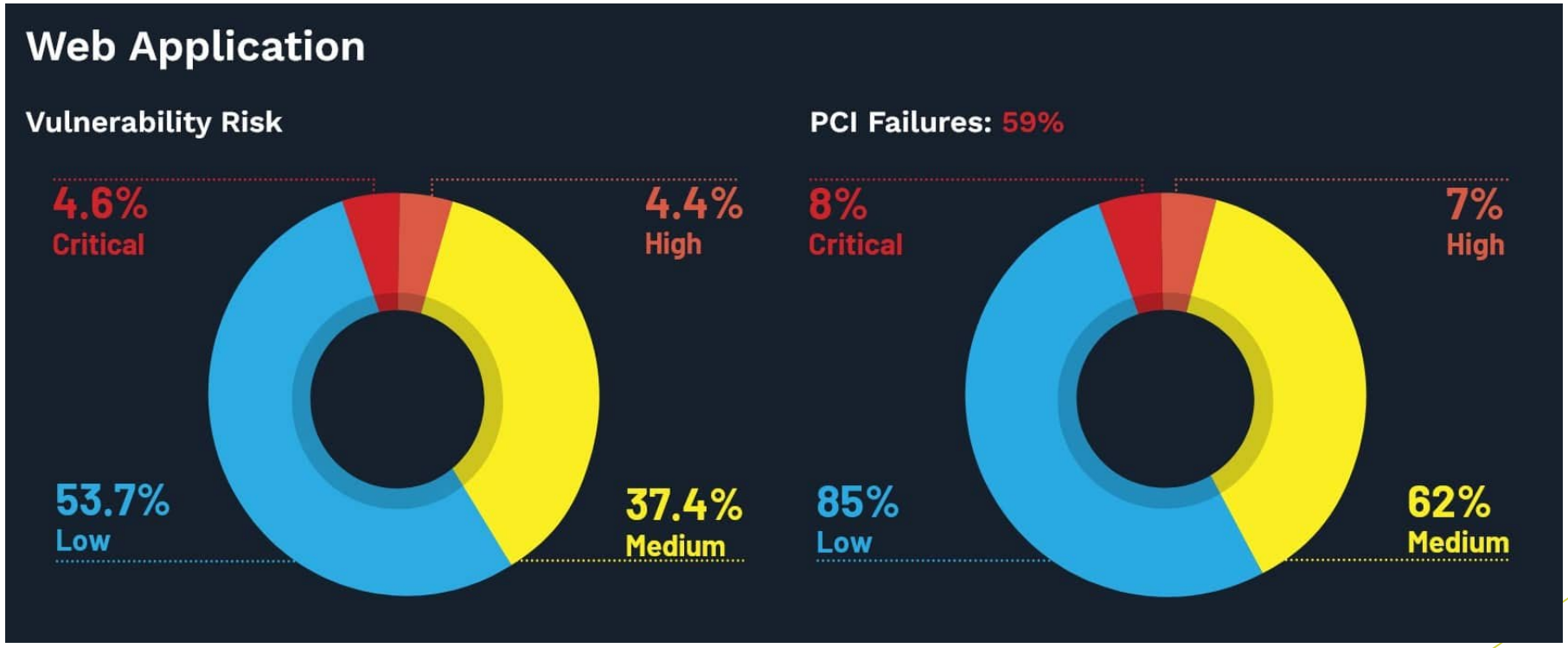


- Programmer: lập trình viên
- System integrator: nhân viên tích hợp hệ thống
- System administrator: nhân viên quản trị hệ thống
- Network administrator: nhân viên quản trị mạng
- Security analyst: nhân viên phân tích an ninh
- Vulnerability analyst: nhân viên phân tích lỗ hổng an ninh
- Artifact analyst: nhân viên phân tích .

- Source code: mã nguồn
- Software component: thành phần phần mềm
- Program version: phiên bản chương trình
- Systems: các hệ thống
- Networks: các mạng
- Security flaw: khiếm khuyết an ninh
- Vulnerability: lỗ hổng an ninh
- Exploit: khai thác lỗ hổng an ninh

I. KIẾN THỨC CHUNG VỀ LỖ HỔNG BẢO MẬT

Báo cáo thống kê về lỗ hổng bảo mật năm 2022 của Edgescan đã phân tích mức độ nghiêm trọng của các lỗ hổng ứng dụng web. Nó phát hiện ra rằng gần 1/10 lỗ hổng trong các ứng dụng sử dụng internet được coi là nguy cơ cao hoặc nguy hiểm. Con số này tăng lên 15 phần trăm nếu mục tiêu thường xử lý các khoản thanh toán trực tuyến.



I. KIẾN THỨC CHUNG VỀ LỖ HỔNG BẢO MẬT

Hơn 11% lỗ hổng bảo mật có điểm số nghiêm trọng.

Theo CVE (Common Vulnerabilities and Exposures) Details, trong số khoảng 176.000 lỗ hổng bảo mật, hơn 19.000 lỗ hổng có điểm CVSS từ 9,0–10,0.

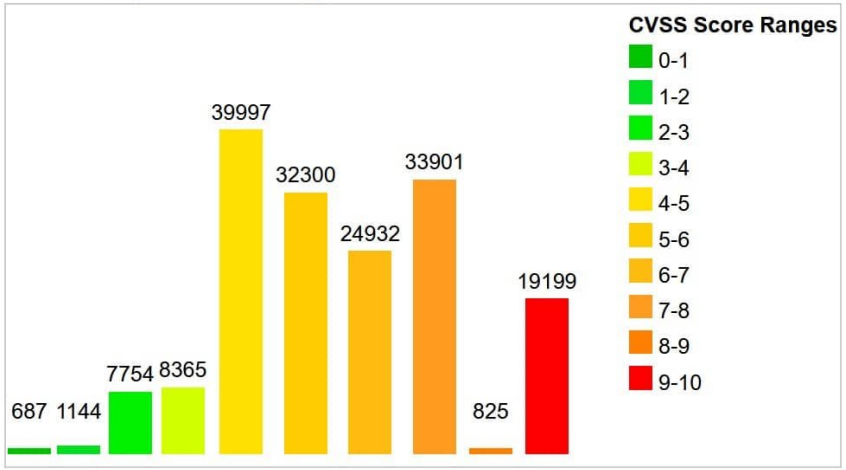
Current CVSS Score Distribution For All Vulnerabilities

Distribution of all vulnerabilities by CVSS Scores

CVSS Score	Number Of Vulnerabilities	Percentage
0-1	687	0.40
1-2	1144	0.70
2-3	7754	4.60
3-4	8365	4.90
4-5	39997	23.70
5-6	32300	19.10
6-7	24932	14.70
7-8	33901	20.00
8-9	825	0.50
9-10	19199	11.40
Total	169104	

Weighted Average CVSS Score: **6.5**

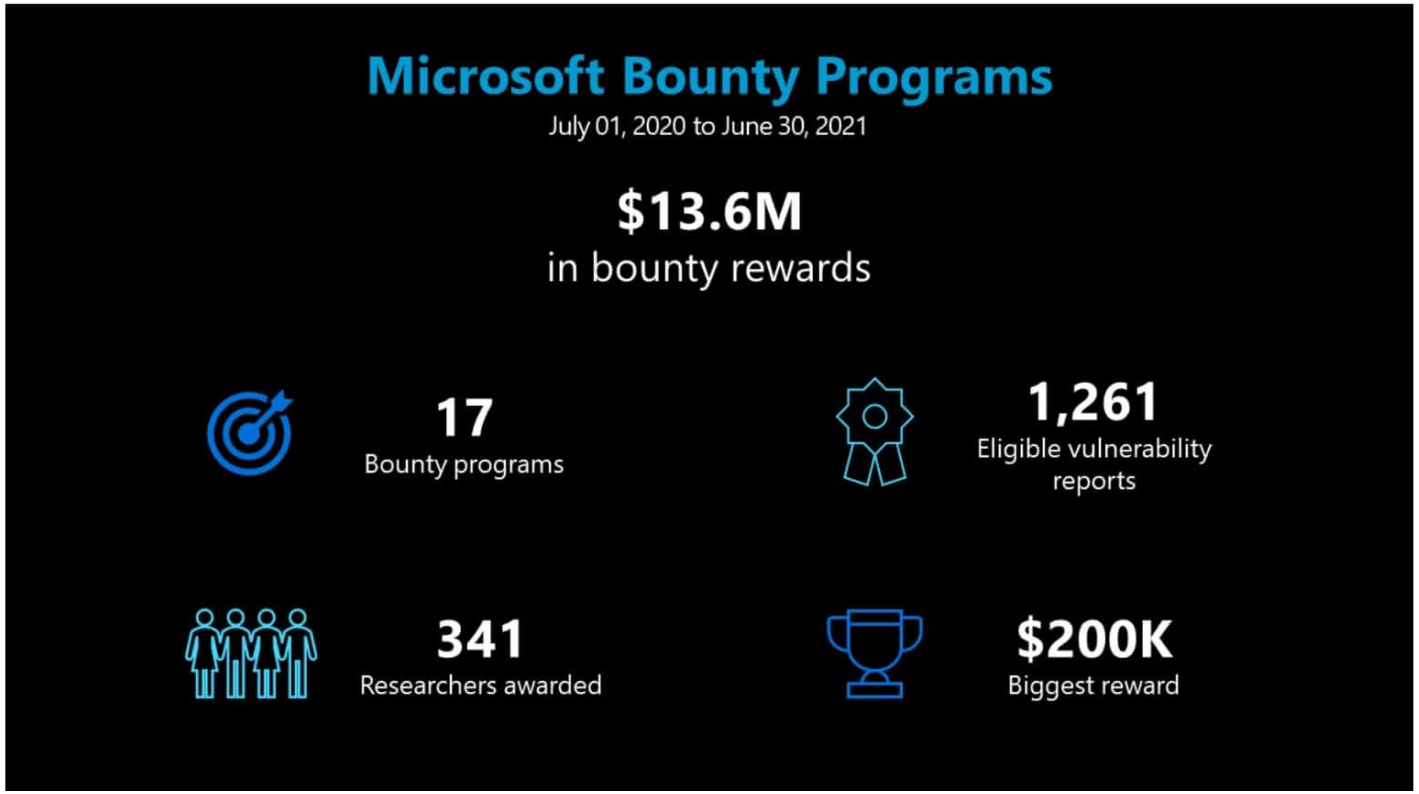
Vulnerability Distribution By CVSS Scores



I. KIẾN THỨC CHUNG VỀ LỖ HỔNG BẢO MẬT

Microsoft đã trả gần 14 triệu đô la tiền thưởng cho lỗi trong một năm.

Theo cách tương tự, Microsoft thưởng cho các nhà nghiên cứu phát hiện và báo cáo lỗi trong phần mềm của mình. Trong một đánh giá vào tháng 7 năm 2021, công ty báo cáo rằng họ đã trả 13,6 triệu đô la tiền thưởng lỗi trong 12 tháng qua. Con số này cao hơn gấp đôi số tiền Google đã trả vào năm 2019



II. LỖ HỔNG HỆ ĐIỀU HÀNH

Các dạng lỗ hổng bảo mật thường gặp trong hệ điều hành và các phần mềm ứng dụng:

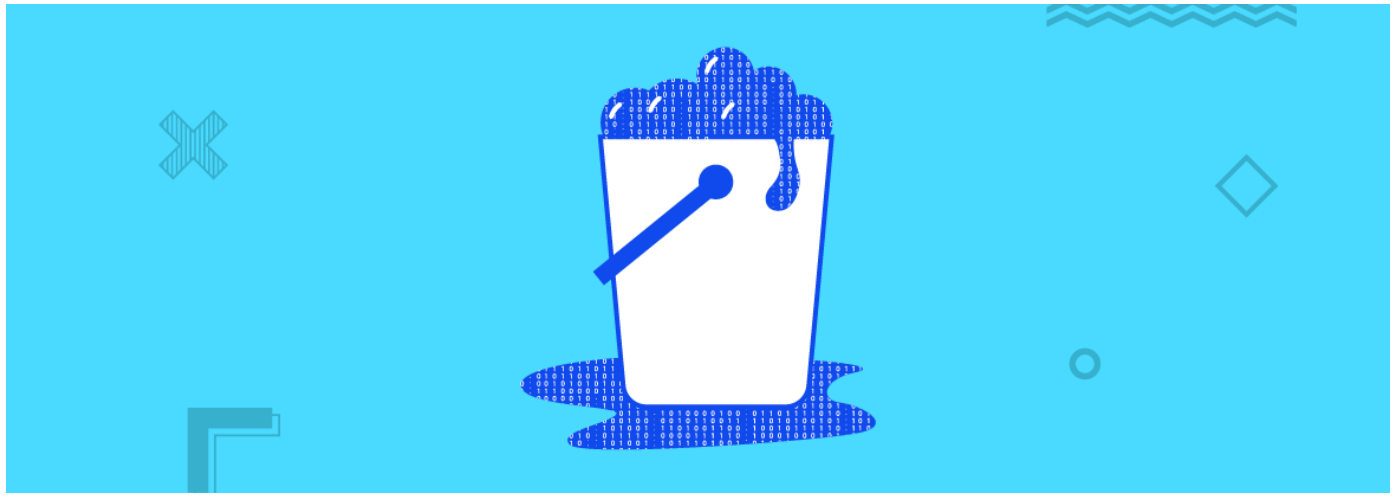
- Vi phạm an toàn bộ nhớ (Memory safety violations)
- Lỗi xác thực đầu vào (Input validation errors)
- Các vấn đề với điều khiển truy cập (access-control problems)
- Các điểm yếu trong xác thực, trao quyền hoặc các hệ mật mã (weaknesses in authentication, authorization, or cryptographic practices)
- Các lỗ hổng bảo mật khác.

II. LỖ HỔNG HỆ ĐIỀU HÀNH

VI PHẠM AN TOÀN BỘ NHỚ

Lỗi tràn bộ đệm (buffer overflow):

Lỗi tràn bộ đệm xảy ra khi một ứng dụng cố gắng ghi dữ liệu vượt khỏi phạm vi bộ đệm (giới hạn cuối hoặc cả giới hạn đầu của bộ đệm);



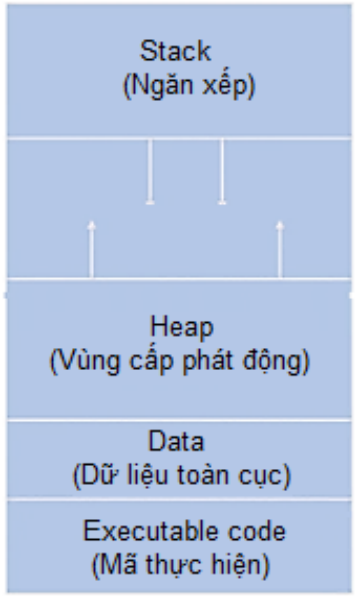
- Lỗi tràn bộ đệm có thể khiến ứng dụng ngừng hoạt động, gây mất dữ liệu hoặc thậm chí giúp kẻ tấn công kiểm soát hệ thống;
- Lỗi tràn bộ đệm là lỗi trong khâu lập trình phần mềm và nó chiếm một tỷ lệ lớn cho số các lỗi gây lỗ hổng bảo mật;
- Không phải tất cả các lỗi tràn bộ đệm có thể bị khai thác bởi kẻ tấn công.

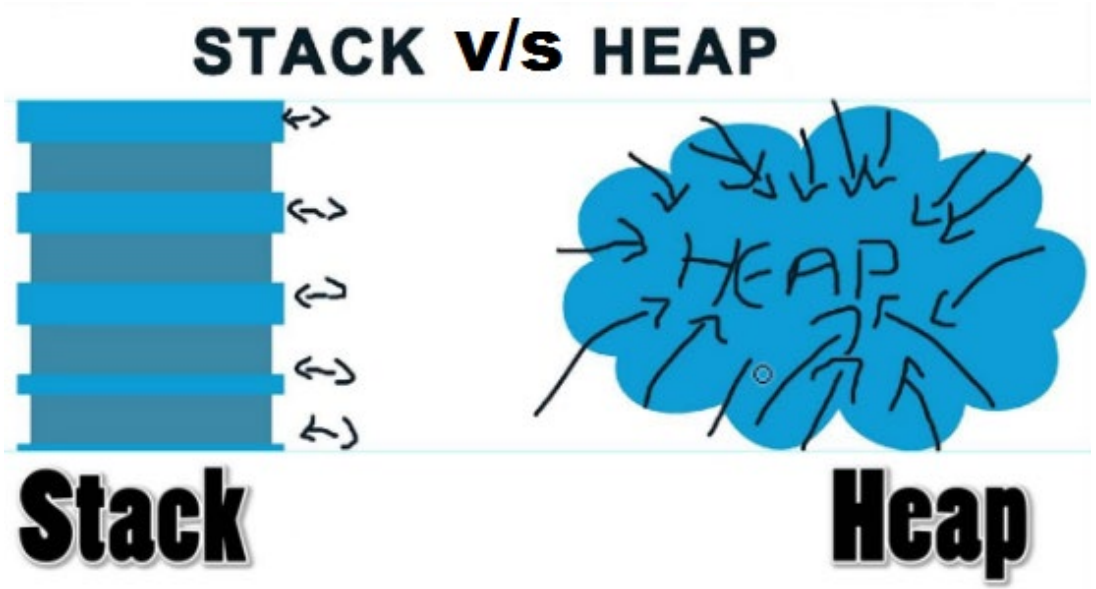
II. LỖ HỔNG HỆ ĐIỀU HÀNH

VI PHẠM AN TOÀN BỘ NHỚ

Các vùng nhớ chứa bộ đệm của ứng dụng:

Ngăn xếp (stack): là một cấu trúc dữ liệu tuyến tính, hoạt động theo cơ chế LIFO (Last In First Out), tạm dịch là "vào sau ra trước". Có nghĩa là phần tử nào được thêm vào sau trong stack thì sẽ được lấy ra trước.





Vùng nhớ được sử dụng để cấp phát cho các tác vụ cấp phát động là vùng nhớ Heap trên RAM. Vùng nhớ này là giới hạn và khi chúng ta sử dụng quá nhiều tác vụ cấp phát động và quên không giải phóng vùng nhớ cấp phát đó thì sẽ sinh ra tràn bộ đệm.

II. LỖ HỔNG HỆ ĐIỀU HÀNH

VI PHẠM AN TOÀN BỘ NHỚ

Các biện pháp phòng chống lỗi tràn bộ đệm:

▪ Lựa chọn ngôn ngữ lập trình. Nhiều ngôn ngữ cung cấp việc kiểm tra tại thời gian chạy, gửi cảnh báo.
▪ Sử dụng các thư viện an toàn. Cài đặt các thư viện an toàn như The Better String Library, Arri Buffer API, Vstr.
▪ Bảo vệ không gian thực thi. Kỹ thuật này ngăn chặn thực hiện mã trong Stack (DEP – Data Execution Prevention);
▪ Sử dụng các cơ chế bảo vệ Stack:
- Thêm một số ngẫu nhiên (canary) phía trước địa chỉ trở về;
- Kiểm tra số ngẫu nhiên này trước khi trở về chương trình gọi để xác định khả năng bị thay đổi địa chỉ trở về.

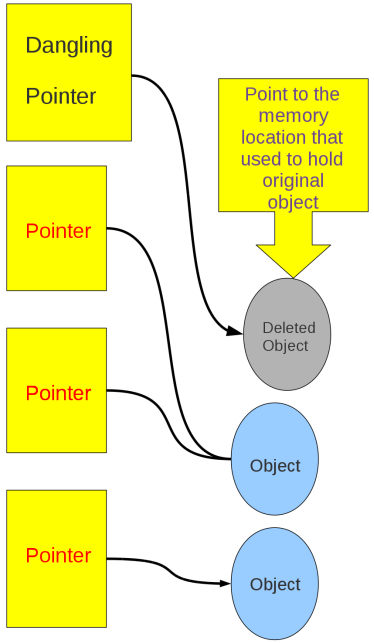
II. LỖ HỔNG HỆ ĐIỀU HÀNH

VI PHẠM AN TOÀN BỘ NHỚ

Con trỏ lơ lửng:

Là con trỏ không trỏ đến một đối tượng hợp lệ thuộc loại tương ứng. Đây là trường hợp đặc biệt của vi phạm bảo mật bộ nhớ.

Con trỏ lơ lửng xảy ra khi một đối tượng bị xóa hoặc di chuyển mà không thay đổi giá trị của con trỏ thành null, do đó con trỏ vẫn trỏ đến vị trí bộ nhớ nơi dữ liệu được lưu trữ trước đó. Vì hệ thống có thể phân bổ lại bộ nhớ đã giải phóng trước đó (bao gồm cả cho tiến trình khác), một con trỏ bị hỏng có thể dẫn đến hành vi không thể đoán được trước của chương trình. Khi một chương trình ghi dữ liệu vào bộ nhớ bằng cách sử dụng một con trỏ như vậy, thì dữ liệu có thể bị hỏng một cách âm thầm dẫn đến các lỗi rất khó phát hiện.



Loại này rất nguy hiểm, và cùng với rò rỉ bộ nhớ, nó xảy ra rất thường xuyên

Các biện pháp phòng chống con trỏ lơ lửng:

- Một số ngôn ngữ lập trình làm giảm khá năng co trỏ bị treo (Java, Python, Go, Javascript..)
- Các phương pháp để cải thiện bảo mật truy cập bộ nhớ

II. LỖ HỔNG HỆ ĐIỀU HÀNH

KHÔNG KIỂM TRA ĐẦU VÀO

Các dữ liệu đầu vào (input data) cần được kiểm tra để đảm bảo đạt các yêu cầu về định dạng và kích thước;

Các dạng dữ liệu nhập điển hình cần kiểm tra:

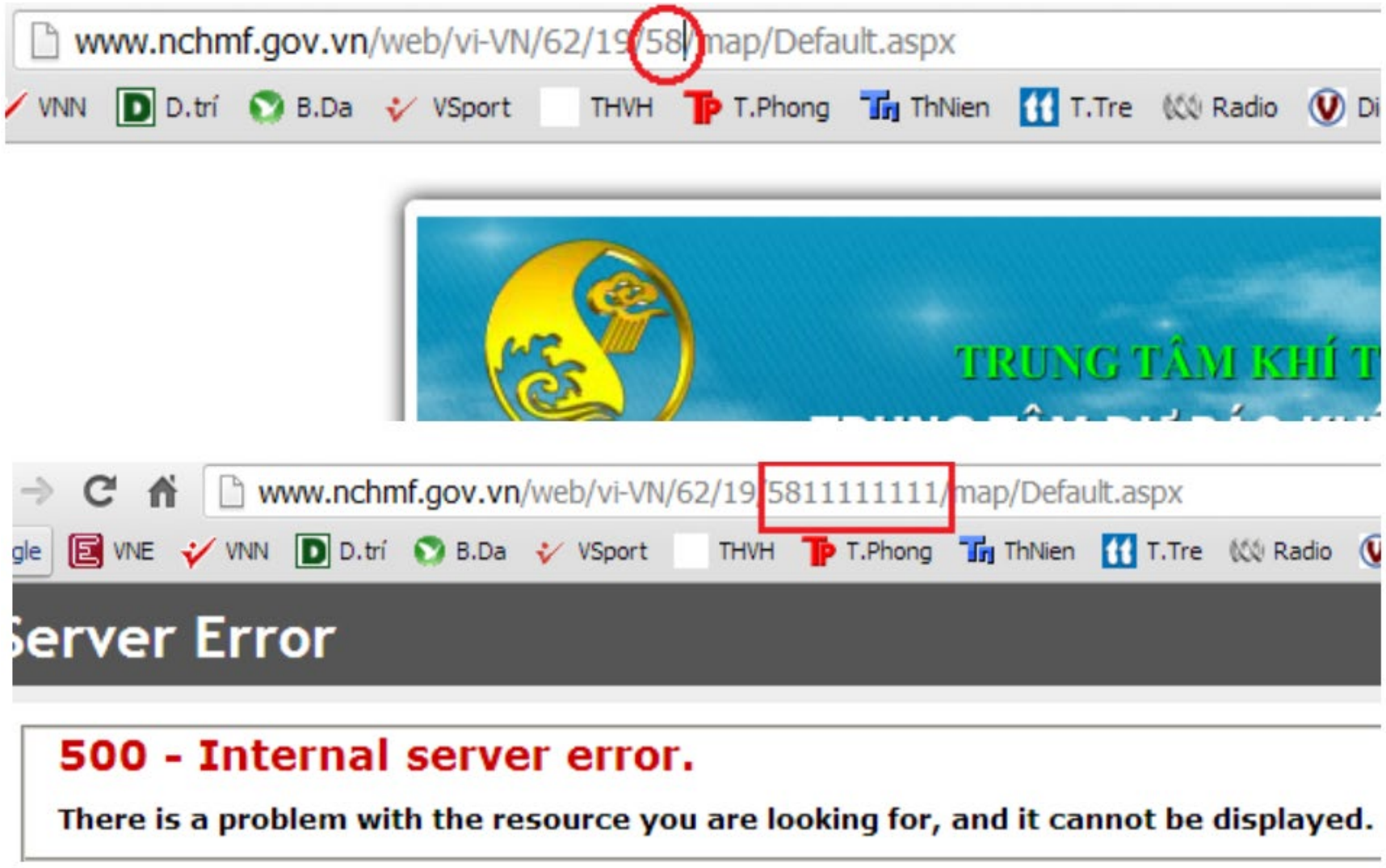
- Các trường dữ liệu text
- Các lệnh được truyền qua URL để kích hoạt chương trình
- Các file âm thanh, hình ảnh, hoặc đồ họa do người dùng hoặc các tiến trình khác cung cấp
- Các đối số đầu vào trong dòng lệnh
- Các dữ liệu từ mạng hoặc các nguồn không tin cậy

Hacker có thể kiểm tra các dữ liệu đầu vào và thử tất cả các khả năng có thể để khai thác.

II. LỖ HỔNG HỆ ĐIỀU HÀNH

KHÔNG KIỂM TRA ĐẦU VÀO

Tấn công khai thác



II. LỖ HỔNG HỆ ĐIỀU HÀNH

KHÔNG KIỂM TRA ĐẦU VÀO

Chèn mã độc SQL vào trường text:

‘Mã asp + SQL server
Dim searchString, sqlString

searchString = “iPhone 14"
sqlString = "SELECT * FROM tbl_products WHERE product_name = '" & searchString & “’”
==> SELECT * FROM tbl_products WHERE product_name = 'iPhone 14’

searchString = "iPhone 14';DELETE FROM tbl_products;--"
sqlString = "SELECT * FROM tbl_products WHERE product_name = '" & searchString & “’”
==> SELECT * FROM tbl_products WHERE product_name = 'iPhone 14'; DELETE FROM tbl_products; --'

II. LỖ HỔNG HỆ ĐIỀU HÀNH

KHÔNG KIỂM TRA ĐẦU VÀO

Chèn mã SQL để đăng nhập mà không cần tài khoản và mật khẩu:

‘Mã asp + SQL server
Dim username, password, sqlString

username = “Antoanthongtin"
password = “lohonghethong"
sqlString = "SELECT * FROM tbl_users WHERE username = '" & username & "' AND password = '" & password & “’”
==> SELECT * FROM tbl_users WHERE username = ' Antoanthongtin ' AND password = ' lohonghethong’

username = “aaaa’ OR 1=1 --"
password = "aaaa"
sqlString = "SELECT * FROM tbl_users WHERE username = '" & username & "' AND password = '" & password & “’”
==> SELECT * FROM tbl_users WHERE username = ‘aaaa’ OR 1=1 --' AND password = 'aaaa'

II. LỖ HỔNG HỆ ĐIỀU HÀNH

KHÔNG KIỂM TRA ĐẦU VÀO

Các biện pháp phòng chống:

▪ Kiểm tra tất cả các dữ liệu đầu vào, đặc biệt dữ liệu nhập từ người dùng và từ các nguồn không tin cậy;

▪ Kiểm tra kích thước và định dạng dữ liệu đầu vào;

▪ Kiểm tra sự hợp lý của nội dung dữ liệu;

▪ Tạo các bộ lọc để lọc bỏ các ký tự đặc biệt và các từ khóa của các ngôn ngữ trong các trường hợp cần thiết mà kẻ tấn công có thể sử dụng:

- Các ký tự đặc biệt: *, ‘, =, --
- Các từ khóa: SELECT, INSERT, UPDATE, DELETE, DROP,....

II. LỖ HỔNG HỆ ĐIỀU HÀNH

CÁC VẤN ĐỀ VỚI ĐIỀU KHIỂN TRUY NHẬP

Điều khiển truy nhập (Access control) liên quan đến việc điều khiển ai (chủ thể) được truy cập đến cái gì (đối tượng)?
Điều khiển truy nhập có thể được thiết lập bởi hệ điều hành hoặc mỗi ứng dụng, thường gồm 2 bước:

- Xác thực (Authentication): xác thực thông tin nhận dạng của chủ thể;
- Trao quyền (Authorization): cấp quyền truy nhập cho chủ thể sau khi thông tin nhận dạng được xác thực.

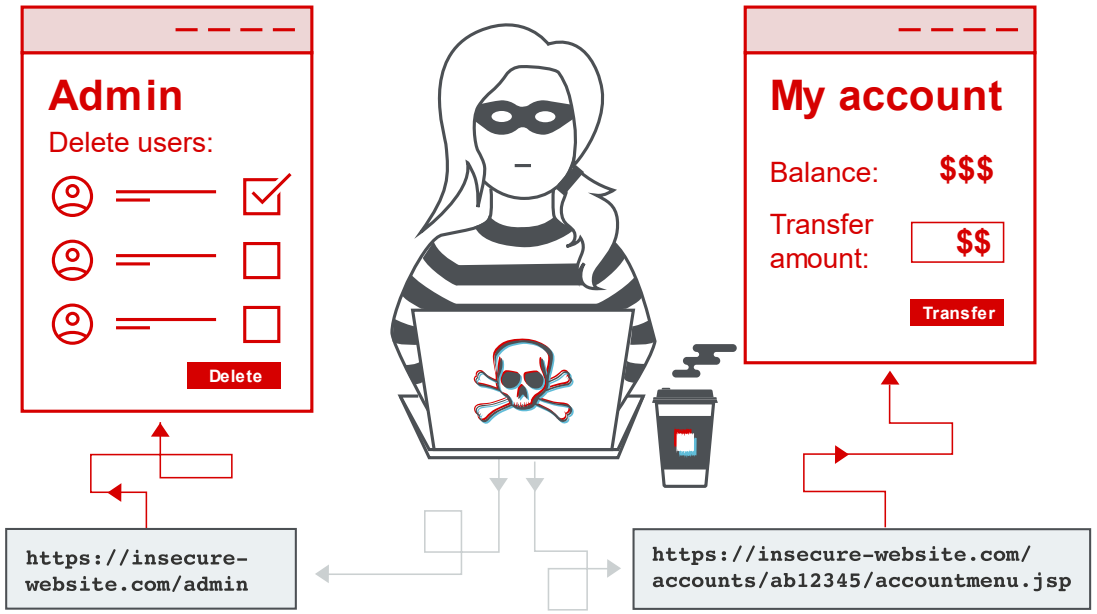
Các chủ thể được cấp quyền truy nhập vào hệ thống theo các cấp độ khác nhau dựa trên chính sách an ninh của tổ chức.

Nếu kiểm soát truy nhập bị lỗi, một người dùng bình thường có thể đoạt quyền của người quản trị và toàn quyền truy nhập vào hệ thống;

Một kẻ tấn công có thể lợi dụng lỗ hổng bảo mật của hệ thống kiểm soát truy nhập để truy nhập vào các file trong hệ thống.

Một ứng dụng chạy trên user quản trị có toàn quyền truy nhập vào hệ thống:

- Nếu một kẻ tấn công chiếm được quyền điều khiển chương trình sẽ có toàn quyền truy nhập vào hệ thống.



II. LỖ HỔNG HỆ ĐIỀU HÀNH

CÁC VẤN ĐỀ VỚI ĐIỀU KHIỂN TRUY NHẬP

Các lỗ hổng kiểm soát truy cập thường có thể được ngăn chặn bằng cách thực hiện phương pháp phòng thủ chuyên sâu và áp dụng các nguyên tắc sau:

- Không bao giờ chỉ dựa vào che giấu để kiểm soát truy cập.
- Trừ khi một tài nguyên được thiết kế để truy cập công khai, còn lại từ chối truy cập theo mặc định.
- Bất cứ khi nào có thể, hãy sử dụng một cơ chế duy nhất trên toàn ứng dụng để thực thi các biện pháp kiểm soát truy cập.
- Ở cấp mã (code), bắt buộc các nhà phát triển phải khai báo quyền truy cập được phép cho từng tài nguyên và từ chối quyền truy cập theo mặc định.
- Kiểm tra kỹ lưỡng và kiểm tra các biện pháp kiểm soát truy cập để đảm bảo chúng hoạt động như thiết kế.

II. LỖ HỔNG HỆ ĐIỀU HÀNH

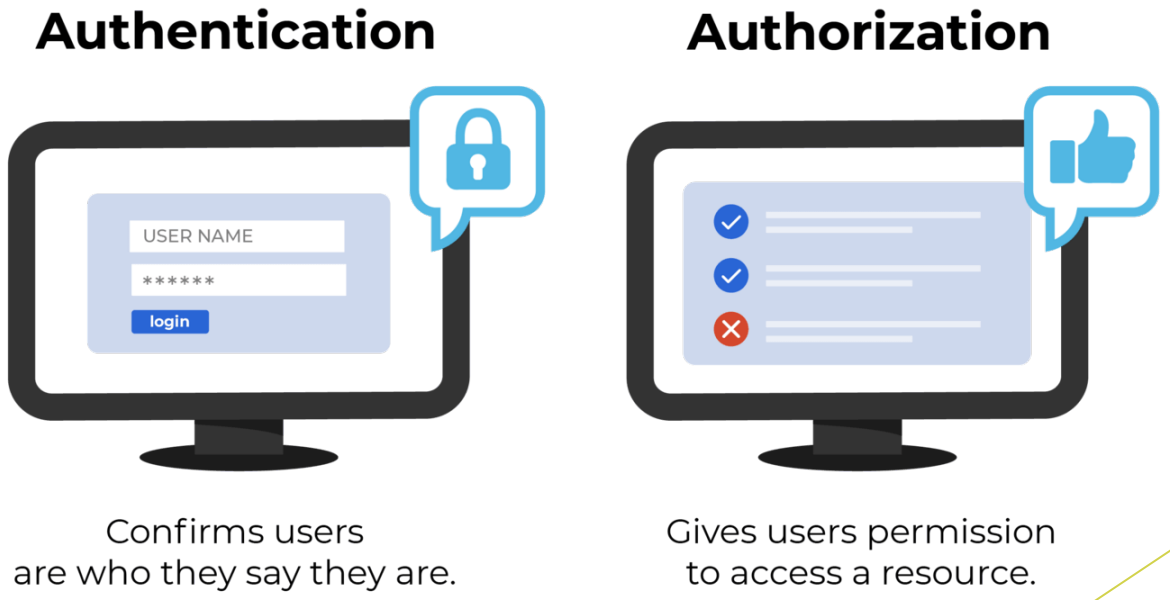
CÁC VẤN ĐỀ VỚI XÁC THỰC, TRAO QUYỀN VÀ MẬT MÃ

Xác thực Authentication:

- Mật khẩu được lưu dưới dạng rõ (plain text) → nguy cơ bị lộ mật khẩu rất cao trong quá truyền thông tin xác thực;
- Sử dụng mật khẩu đơn giản, dễ đoán, hoặc dùng mật khẩu trong thời gian dài;
- Sử dụng cơ chế xác thực không đủ mạnh: ví dụ các cơ chế xác thực của giao thức HTTP.

Trao quyền Authorization:

- Cơ chế thực hiện trao quyền không đủ mạnh, dễ bị vượt qua;

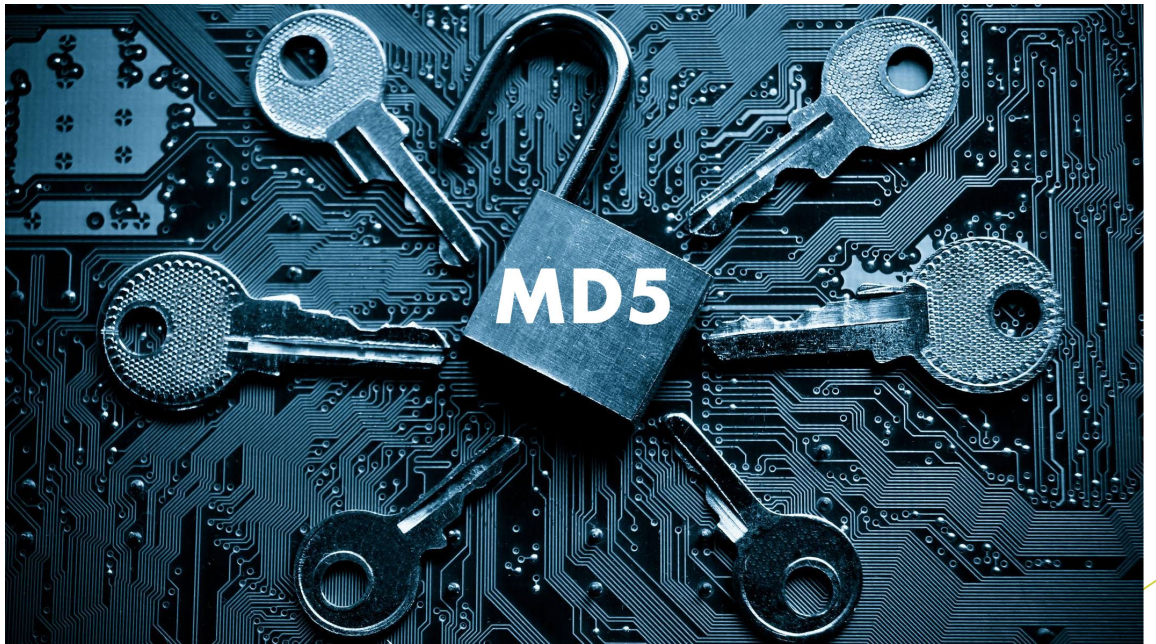


II. LỖ HỔNG HỆ ĐIỀU HÀNH

CÁC VẤN ĐỀ VỚI XÁC THỰC, TRAO QUYỀN VÀ MẬT MÃ

Các vấn đề với các hệ mật mã:

- Sử dụng giải thuật mã hóa/giải mã, hàm băm yếu, lạc hậu, hoặc có lỗ hổng (DES, MD4, MD5,...);
- Sử dụng khóa mã hóa/giải mã yếu;
  - Khóa có chiều dài ngắn;
  - Khóa dễ đoán.
- Các vấn đề trao đổi khóa bí mật;
- Các vấn đề xác thực người gửi/người nhận;
- Chi phí tính toán lớn (đặc biệt đối với các hệ mã hóa khóa công khai).



II. LỖ HỔNG HỆ ĐIỀU HÀNH

CÁC LỖ HỔNG BẢO MẬT KHÁC

Các thao tác không an toàn với files:

▪ Thực hiện đọc/ghi file lưu ở những nơi mà các người dùng khác cũng có thể ghi file đó;
▪ Không kiểm tra chính xác loại file, định danh thiết bị, các links hoặc các thuộc tính khác của file trước khi sử dụng;
▪ Không kiểm tra mã trả về sau mỗi thao tác với file;
▪ Giả thiết một file có đường dẫn cục bộ là file cục bộ và bỏ qua các thủ tục kiểm tra:
• File ở xa có thể được ánh xạ vào hệ thống file cục bộ → có đường dẫn cục bộ

Các điều kiện đua tranh (Race conditions):

▪ Một điều kiện đua tranh tồn tại khi có sự thay đổi trật tự của 2 hay một số sự kiện gây ra sự thay đổi hành vi của hệ thống;
▪ Đây là một dạng lỗi nếu chương trình chỉ có thể thực hiện đúng chức năng nếu các sự kiện phải xảy ra theo đúng trật tự;
▪ Kẻ tấn công có thể lợi dụng khoảng thời gian giữa 2 sự kiện để chèn mã độc, đổi tên file hoặc can thiệp vào quá trình hoạt động bình thường của hệ thống.

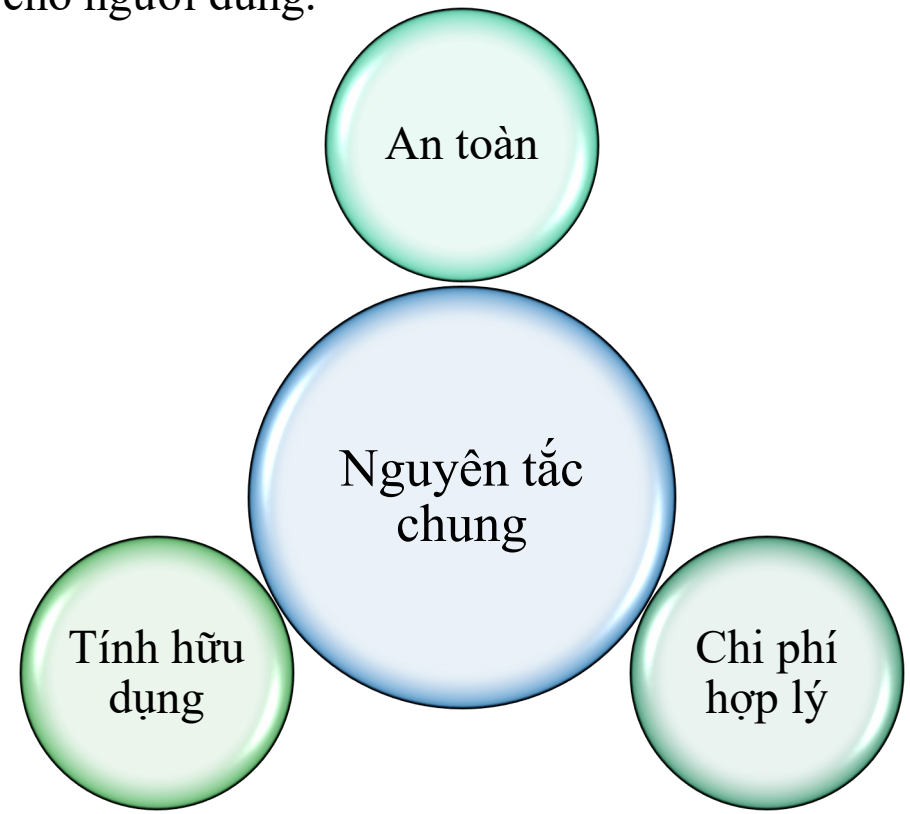
II. LỖ HỔNG HỆ ĐIỀU HÀNH

QUẢN LÝ VÀ KHẮC PHỤC CÁC LỖ HỔNG BẢO MẬT

Nguyên tắc chung

Việc quản lý, khắc phục các lỗ hổng bảo mật và tăng cường khả năng đề kháng cho hệ thống cần được thực hiện theo nguyên tắc chung là cân bằng giữa mức An toàn(Secure), Chi phí hợp lý và tính hữu dụng(usable).

Ý nghĩa cụ thể của nguyên tắc này là đảm bảo an toàn cho hệ thống ở mức phù hợp với chi phí hợp lý và hệ thống vẫn phải hữu dụng, hay có khả năng cung cấp đầy đủ các tính năng hữu ích cho người dùng.



II. LỖ HỔNG HỆ ĐIỀU HÀNH

QUẢN LÝ VÀ KHẮC PHỤC CÁC LỖ HỔNG BẢO MẬT

Một số biện pháp cụ thể

▪ Thường xuyên cập nhật thông tin về các điểm yếu, lỗ hổng bảo mật từ các trang web chính thức:

- http://cve.mitre.org/ (CVE - Common Vulnerabilities and Exposures)
- http://www.cvedetails.com/ (CVE Details)
- http://web.nvd.nist.gov/ (National Vulnerability Database)
- https://owasp.org/www-community/vulnerabilities/

▪ Định kỳ cập nhật các bản vá, nâng cấp hệ điều hành và các phần mềm ứng dụng;

• Sử dụng các hệ thống quản lý các bản vá và tự động cập nhật định kỳ

– Microsoft Windows Updates
– Tiện ích Update trên Linux
– Tính năng tự động Update của các ứng dụng (Như Google Update service)

• Với các lỗ hổng nghiêm trọng, cần cập nhật tức thời.

II. LỖ HỔNG HỆ ĐIỀU HÀNH

QUẢN LÝ VÀ KHẮC PHỤC CÁC LỖ HỔNG BẢO MẬT

Một số biện pháp cụ thể

- Cần có chính sách quản trị người dùng, mật khẩu và quyền truy nhập chặt chẽ ở mức hệ điều hành và mức ứng dụng:
  - Người dùng chỉ được cấp quyền truy nhập vừa đủ để thực hiện công việc được giao.
  - Nếu được cấp nhiều quyền hơn mức cần thiết → lạm dụng.
- Sử dụng các biện pháp phòng vệ ở lớp ngoài như Firewall, proxy:
  - Chặn các dịch vụ/cổng không thực sự cần thiết;
  - Ghi logs các hoạt động truy nhập mạng.
- Sử dụng các phần mềm rà quét lỗ hổng, rà quét các phần mềm độc hại:
  - Có thể giảm thiểu nguy cơ bị lợi dụng, khai thác lỗ hổng bảo mật.