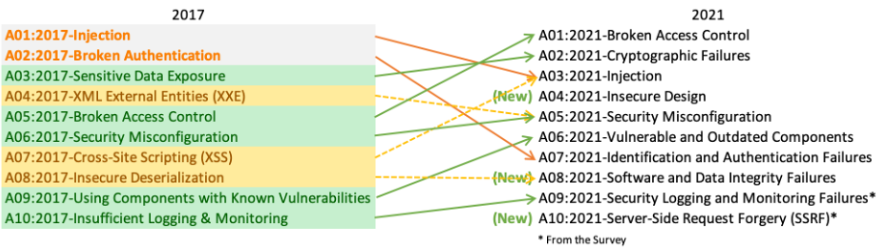
OWASP ( Open Web Application Security Project ) là 1 tổ chức phi lợi nhuận nhằm cải thiện tính bảo mật phần mềm , trang web . Các cấp độ nguy hiểm về nguy cơ tàn phá các trang web . Top 10 OWASP tập trung vào 10 rủi ro , lỗ hổng quan trọng nhất được cập nhập thường xuyên 4 – 5 năm 1 lần . Báo cáo luôn được tín nhiệm bởi 1 nhóm chuyên gia bảo mật từ các nơi trên thế giới .



Theo như cập nhập gần đây nhất ( 2021 ) :

A01 : Broken Access Control (Kiểm soát lỗi truy cập) :

* Lý do : Thường thì sẽ có 1 số chính sách nhất định cho phép người dùng được truy cập ở mức độ cho phép . Thế nhưng , attacker có thể lợi dụng một vài lỗ hổng từ phía server để truy cập trái phép , vượt mức mà người dùng có thể
* Một vài cách điều khiển truy cập vượt mức như :

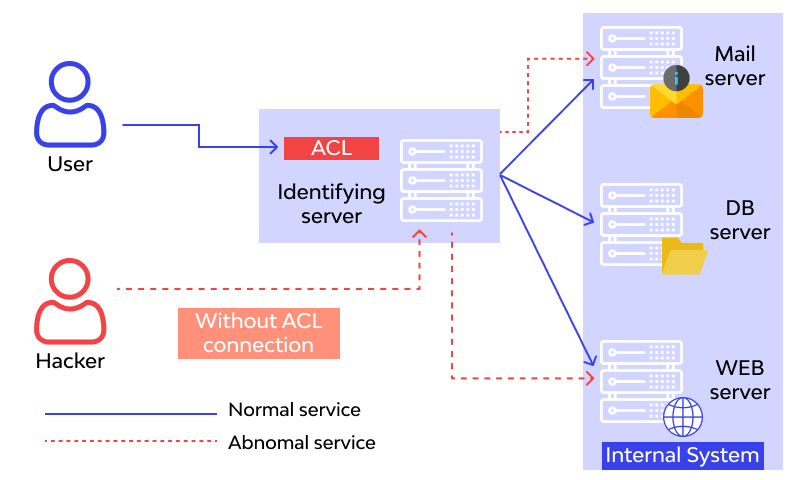
+ ) Sửa đổi URL , trạng thái nội bộ , trang HTML , hoặc sử dụng các công cụ tấn công yêu cầu chỉnh sửa API

+ ) Truy cập API với những kiểm soát truy cập cho POST , PUT và DELETE

+ ) Giả truy cập với tư cách là admin

+ ) .....

* Cách thức hoạt động :



Đầu tiên User sẽ đăng nhập vào ACL để xác định danh tính của mình sau đó nhập các thông tin , dữ liệu cá nhân mình lên ( đương nhiên tất cả hành động trên đều hoàn toàn được cho phép và hợp lệ ) . Tiếp theo hacker sẽ bắt đầu đăng nhập 1 cách không hợp lệ , bỏ qua việc kết nối ACL để truy cập thẳng vào server và lấy cắp mọi thông tin

* Cách ngăn chặn :

+ ) that đổi cơ chế kiểm soát truy cập từ server

+ ) Cảnh báo cho admin về những lần access không thành công

* Ví dụ minh họa :

1 ) Attacker nhắm mục tiêu vào các URLs . Thông qua quyền hạn của admin để có thể truy cập vào trang admin :

https://example.com/app/getappInfo

https://example.com/app/admin\_getappInfo

Nếu như người dùng không được xác thực truy cập được trang admin => có lỗ hổng

A02 : Cryptographic failures ( Lỗi bảo mật )

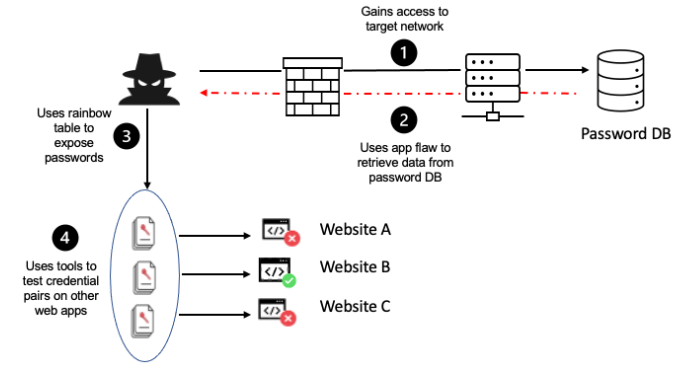
* Lí do : Hacker thường sẽ nhắm tới các dữ liệu nhạy cảm như : password , thẻ ngân hàng , thông tin cá nhân ... khi mà người dùng không bảo vệ chúng cẩn thận . Lỗi bảo mật thường là nguyên do chi các rò rỉ dữ liệu nhạy cảm . Những lí do lỗi bảo mật xảy ra là khi :

1 ) Lưu trữ hay truyền đi dữ liệu dưới dạng clear text ( text chưa được mã hóa )

2 ) Lưu trữ dữ liệu với Encyption cũ và yếu

3 ) .....

* Cách thức :



Ngăn chặn :

1 ) Encrypt toàn bộ dữ liệu khi nó đang ở database hoặc đang đượcc chuyển đi

2 ) Đảm bảo cập nhập và tiêu chuẩn thuật toán , giao thức , key ở mức độ mạnh . Dùng key ban quản lí hợp lí

3 ) Vì đây là lỗi bảo mật nên cần lưu ý về mật khẩu , mã hóa , ...

A03 : Injection

* Cách thức :

Khi gửi dữ liệu không đáng tin cậy đén trình thông mã ( code interpreter ) thông qua việc điền vào các form hoặc 1 số dữ liệu khác đến ứng dụng web . Lợi dụng những lỗ hổng về câu truy vấn của các ứng dụng . Được thực hiện bằng cách chèn 1 đoạn SQL để làm sai lệch đi câu truy vấn ban đầu .

* Ngăn chặn :

1 ) Kiểm tra ký các trường nhập dữ liệu và cần ràng buộc kỹ dữ liệu input

2 ) Dùng Regular Expression để loại bỏ các ký tự lạ hoặc các ký tự không phải là số

3 ) Dùng thêm hàm mysql\_real\_escape\_string để chuyển đổi 1 chuỗi thành một query an toàn

4 ) Dùng các Framework và hạn chế dùng code thuần tối đa có thể

* Ví dụ :

Giả sử , một web sử dụng PHP cho server-side , để lấy giá trị tham số ‘name’ từ đường dẫn làm input cho truy vấn SQL , sử dụng đoạn code sau :

$sql = "SELECT \* FROM Users WHERE lname='$\_GET["name"]'";

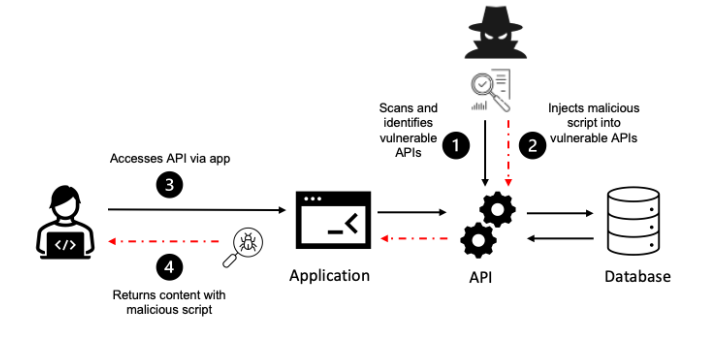
Input : ‘ OR 1=1;-- => truy vấn tất cả dữ liệu trong bảng users do điều kiện 1 = 1 là đúng

A04 : Insecure Design ( Thiết kế không an toàn )

Lí do xuất hiện thiết kế không an toàn là nó sẽ được tập trung đến những rủi ro liên quan đến lỗ hổng trong thiết kế . Các lỗ hổng trong Insecure Design không phải là những lỗi có thể sửa bằng cách thực hiện . Bên cạnh đó , Insecure design và Security Implementation là 2 khái niệm khác nhau được phân tích như sau : An insecure design cannot be fixed by a perfect implementation as by definition, needed security controls were never created to defend against specific attacks ( dịch thô : Thiết kế không an toàn không thể sửa bằng cách thực hiện theo định nghĩa , các điều khiển an toàn không bao giờ được đặt ra để chống lại các cuộc tấn công cụ thể )

Để khai thác lỗi Insecure Design , attackers có thể tấn công các mô hình quy trình làm việc trong phần mềm để tiết lộ một phạm vi lớn về sự thiếu sót , dễ bị tấn công

Cách thức :



Attacker sẽ khai thác một thiết kế API chưa được bảo mật tốt và chưa có một bộ lọc input hoàn chỉnh

Ngăn chặn :

1 ) Giới hạn tiêu thụ tài nguyên của người dùng

2 ) Viết các bài kiểm tra tích hợp để xác thực rằng tất cả các luồng quan trọng đều kháng với mô hình mối đe dọa.

3 ) ....

A05 : Security Misconfiguration ( Sai sót cấu hình )

Lí do : Cấu hình an ninh lỏng lẻo tại các tầng kiến trúc của web hay ứng dụng như nền tảng , framework , máy chủ , CSDL và mã tùy chỉnh , ....

Cách thức :

1 ) Chạy website khi chế độ debug được bật , dẫn đến các mật khẩu , cáu trúc folder bị lộ .

2 ) Thiếu cấu hình htaccess dẫn đến bị Directory listing

3 ) Sử dụng thư viện cũ

4 ) Thêm vào các chức năng không cần thiết trong web

5 ) Không thay đổi default key hoặc mật khẩu mặc định trong các thư viện sử dụng

Ví dụ :

1 ) Cấu hình cho biến WP\_DEBUG của wordpress set là true ( để gỡ lỗi ) rồi quên tắt

2 ) Cấu hình php trên host mà cho display All Error

3 ) ....

Ngăn chặn :

1 ) Một nền tảng tối thiểu , không có bất kì những tính năng không cần thiết nào .

2 ) Một quy trình tự động để xác minh tính hiệu quả của cấu hình và cài đặt trên mọi môi trường

A06 : Vulnerable and Outdated components ( Các thành phần dễ bị xâm phạm và lỗi thời )

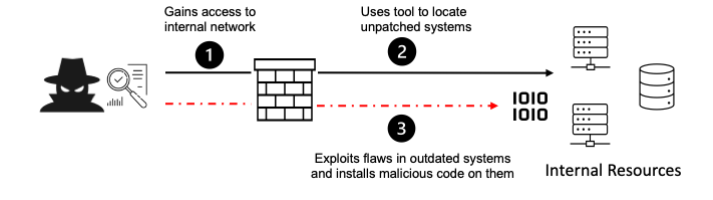
Lí do : 1 ) Không nắm bắt rõ versions của các thành phần đang dùng ( cả 2 phía client và server )

2 ) Phần mềm dễ bị xâm phạm , không an toàn hoặc lỗi thời

3 ) không scan những lỗi dễ bị xâm phạm thường xuyên

4 ) Các software devs không kiểm tra khả năng tương thích , nâng cấp hoặc vá những thư viện

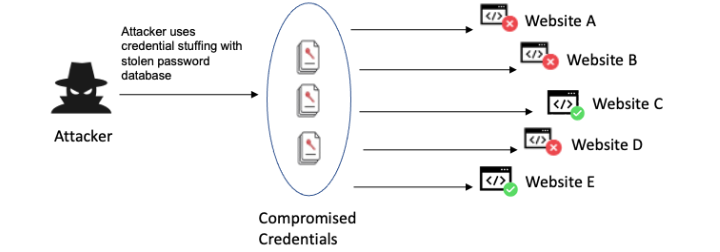
Cách thức :



A07 : Identification and authentication failures ( Lỗi xác thực )

Lí do : Lỗi xuất hiện khi các chức năng liên quan đến nhận dạng người dùng hoặc quản lí phiên không được thực thi chính xác hoặc có thể không được bảo về đầy đủ

Cách thức :

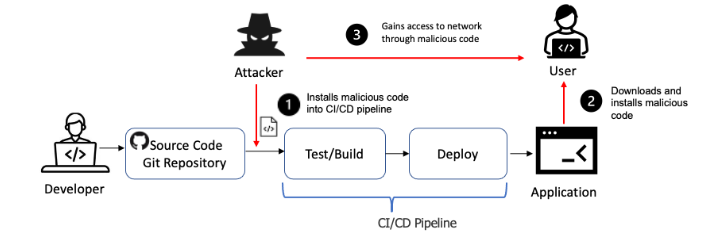


Hacker sẽ kiếm mật khẩu database trên diễn đàn hacker , bởi vì do thuật toán hash được encrypt yếu nên hacker có thể lấy ra được định dạng của người dùng . Sau đó dùng “credential stuffing tools” ( công cụ nhồi thông tin ) để thử xem hợp với website nào

A08 : Software and data integrity failures ( Lỗi toàn vẹn phần mềm và dữ liệu )

Lí do : Lỗi toàn vẹn phần mềm và dữ liệu liên quan đến mã và cơ sở hạ tầng không được bảo về để chống lại tính vi phạm toàn quyền . Điều này xảy ra khi dùng phần mềm từ các nguồn đáng tin cậy

Cách thức :

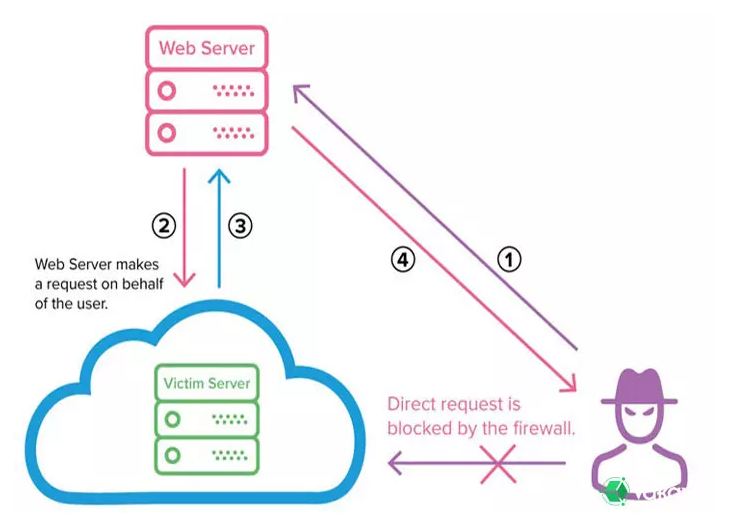


A09 : Security Logging and Monitoring Failures

A10 : Server-side Request Forgery (SSRF) ( Yêu cầu giả mạo máy chủ )

Là một lỗ hổng xuất hiện khi một kẻ tấn công có khả năng tạo ra các yêu cầu từ máy chủ dễ bị xâm hại

Cách thức :



Kẻ tấn công không thể gửi yêu cầu thẳng đến máy nạn nhân ( bởi vì chúng đã bị chặn bởi tường lửa ) , để tấn công một mạng nội bộ , kẻ tấn công phải gửi yêu cầu đến máy chủ web dễ bị tấn công vi phạm lỗ hổng SSRF , máy chủ web yêu cầu máy chủ của nạn nhân nằm phía sau tường lửa . Máy chủ nạn nhân phản hồi với dữ liệu . Nếu lỗ hổng cụ thể cho phép nó , dữ liệu được gửi lại cho kẻ tấn công .