**Directory Traversal**

**Khái niệm:**

Truyền thư mục là 1 lỗ hổng web cho phép kẻ tấn công đọc tệp tin bất kì trên máy chủ đang chạy ứng dụng. Nó có thể bao gồm đoạn mã và dữ liệu, thông tin đăng nhập trên hệ thống phía sau, tệp tin nhạy cảm của hệ điều hành.

Trong 1 vài trường hợp, kẻ tấn công có thể viết tệp tin bất kì trên máy chủ, cho phép nó điều chỉnh dữ liệu ứng dụng và hành vi, toàn quyền kiểm soát máy chủ.

**Đọc tệp tin bất kì qua truyền thư mục:**

Xem xét một ứng dụng web hiển thị hình ảnh sản phẩm. Những hình ảnh này được tải qua HTML như sau:

<img src="/loadImage?filename=218.png">

Trên url, hàm loadImage sẽ truyền tham số filename=218.png🡪hình ảnh được lưu ở thư mục /var/www/images 🡪Ứng dụng sẽ dùng hệ thống api để truy suất đến thư mục đó và lấy hình ảnh.

/var/www/images/218.png

Nếu ứng dụng không phòng thủ truyền thư mục, kẻ tấn công có thể yêu cầu url sau để truy cập tệp tin bất kì từ hệ thống máy chủ:

https://insecure-website.com/loadImage?filename=../../../etc/passwd

Điều này sẽ làm cho ứng đụng truy suất đến thư mục:

/var/www/images/../../../etc/passwd

Nhưng mà thư mục images làm gì có file /etc/passwd. Chuỗi ../ giúp trả về thư mục cha.

images/../ 🡪 thư mục cha của images là www.

www../🡪 thư mục cha của www là var

var../🡪trả về thư mục gốc (root)

**=> /etc/passwd**

Trong hệ điều hành Unix, file /etc/passwd là tệp tin chứa thông tin người dùng đăng kí trên máy chủ.

Trong Windows, ../ hay ..\ đều giống nhau và cuộc tấn công tương tự để truy cập tệp tin của hệ điều hành là:

https://insecure-website.com/loadImage?filename=..\..\..\windows\win.ini

**Những khó khăn khi khai thác lỗ hổng truyền tệp tin:**

Nếu một ứng dụng bỏ qua hoặc chặn những chuỗi truyền file, vẫn có thể vượt qua được lớp phòng thủ nhờ những cách sau:

* Dùng đường dẫn tuyệt đối từ hệ thống root như filename=/etc/passwd vì một số ứng dụng xử lí tệp tin như tương đối.
* Dùng chuỗi lồng như ….// hay ….\/, sẽ trả về chuỗi gốc khi bị ứng dụng bỏ qua.
* Nếu ứng dụng chặn chuỗi ../, mã hóa URL hoặc double URL như %2e%2e%2f hay %252e%252e%252f
* Thêm thư mục cơ sở như /var/www/images nếu ứng dụng yêu cầu truyền đầy đủ đường dẫn file.

filename=/var/www/images/../../../etc/passwd

* Nếu ứng dụng yêu cầu phần mở rộng của tệp tin như .png, sử dụng null byte để bỏ qua phần mở rộng.

filename=../../../etc/passwd%00.png

**Cách ngăn chặn cuộc tấn công truyền tệp tin:**

Cách hiệu quả nhất để ngăn chặn lỗ hổng này là tránh gửi đầu vào do người dùng cung cấp đến hệ thống file APIs. Rất nhiều ứng dụng làm điều đó có thể viết lại để vận chuyển file theo cách an toàn hơn.

Nếu không thể tránh việc gửi đầu vào đến hệ thống file APIs, phải có 2 lớp phòng thủ:

* Ứng dụng nên xác thực đầu vào. Tốt nhất nên so sánh với danh sách những giá trị được phép như [a-zA-Z0-9]
* Sau khi xác thưc đầu vào, ứng dụng nên đưa đầu vào đến thư mục cơ sở và sử dụng nền tảng tệp tin APIs để chuẩn hóa đường dẫn. Nó nên xác thực xem đường dẫn đã được chuẩn hóa có đúng với thư mục mong muốn không.

Source code Java để chuẩn hóa đường dẫn :

File file = new File(BASE\_DIRECTORY, userInput);

if (file.getCanonicalPath().startsWith(BASE\_DIRECTORY)) {

// process file

}