BÁO CÁO THỰC HÀNH

**Môn học: Bảo mật Web và Ứng dụng**

**Lab 6: Lab Tổng Hợp**

*GVHD: Ngô Khánh Khoa*

**Nhóm: 6**

1. **THÔNG TIN CHUNG:**

Lớp: NT213.P11.ANTT.2

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Họ và tên | MSSV | Email |
| 1 | Lại Quan Thiên | 22521385 | 22521385@gm.uit.edu.vn |
| 2 | Mai Nguyễn Nam Phương | 22521164 | 22521164@gm.uit.edu.vn |
| 3 | Hồ Diệp Huy | 22520541 | 22520541@gm.uit.edu.vn |
| 4 | Nguyễn Phúc Nhi | 22521041 | 22521041@gm.uit.edu.vn |

1. **NỘI DUNG THỰC HIỆN:[[1]](#footnote-2)**

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Nội dung | Tình Trạng | Thực hiện |
| 1 | Bài 1 | 100% | Nhóm 6 |
| 2 | Bài 2 | 0% | Nhóm 6 |
| 3 | Bài 3 | 80% | Nhóm 6 |
| 4 | Bài 4,5 | 50% | Nhóm 6 |

**Phần bên dưới của báo cáo này là tài liệu báo cáo chi tiết của nhóm thực hiện.**

BÁO CÁO CHI TIẾT

Bài tập 1:(2đ): Báo cáo lỗ hổng tìm thấy của lab1: review-lab.

**Tiêu đề**: Directory Indexing được bật

- **Mô tả lỗ hổng**:

  + **Tóm tắt**: Máy chủ cho phép duyệt thư mục /images/, điều này có thể làm lộ dữ liệu nhạy cảm.

+ **Các bước thực hiện và minh chứng:**

1. Truy cập URL: <http://127.0.0.1:8999/images/>.

2. Duyệt được danh sách các tệp trong thư mục, bao gồm cả các tệp có thể chứa thông tin nhạy cảm.

A screenshot of a computer

Description automatically generated

- **Mức độ ảnh hưởng của lỗ hổng:** Cho phép truy cập trái phép vào các tệp hoặc thư mục nhạy cảm, dẫn đến việc lộ thông tin và có thể bị khai thác bởi tin tặc.

- **Khuyến cáo khắc phục:**

+ Tắt Directory Indexing trong tệp cấu hình Apache:

*<Directory /path/to/images>*

*Options -Indexes*

*</Directory>*

+ Sau đó khởi động lại Apache:

*sudo systemctl restart apache2*

**Tiêu đề**: Tệp nhạy cảm được tìm thấy

- **Mô tả lỗ hổng**:

  + **Tóm tắt**: Một số tệp nhạy cảm được tìm thấy trên máy chủ web, có thể tiết lộ thông tin về hệ thống:

+ /LICENSE.txt: File này tiết lộ thông tin phần mềm hoặc framework đang được sử dụng.

+ /icons/README: Đây là tệp mặc định của Apache, có thể tiết lộ cấu hình máy chủ.

+ **Các bước thực hiện và minh chứng:**

1. Truy cập các URL trực tiếp:

+ <http://127.0.0.1:8999/LICENSE.txt>

A screenshot of a computer

Description automatically generated

+ <http://127.0.0.1:8999/icons/README>

A screenshot of a computer

Description automatically generated

2. Nội dung tệp có thể đọc được, cung cấp thông tin hệ thống nhạy cảm.

- **Mức độ ảnh hưởng của lỗ hổng:** Thông tin tiết lộ có thể hỗ trợ tin tặc xác định phiên bản phần mềm hoặc cấu hình máy chủ, từ đó tiến hành các cuộc tấn công có chủ đích.

- **Khuyến cáo khắc phục:**

+ Xóa các tệp không cần thiết:

*sudo rm /var/www/html/LICENSE.txt*

*sudo rm /var/www/html/icons/README*

+ Hạn chế quyền truy cập vào các tệp hoặc thư mục không cần thiết bằng cách sử dụng cấu hình máy chủ web.

Bài tập 2:(2đ): Báo cáo lỗ hổng tìm thấy của lab2.

**Tiêu đề**:

- **Mô tả lỗ hổng**:

  + **Tóm tắt**:

**+ Các bước thực hiện và minh chứng:**

- **Mức độ ảnh hưởng của lỗ hổng:**

- **Khuyến cáo khắc phục:**

Bài tập 3:(2đ): Tìm subdomain thuộc tên miền chính ở trang web sau: arrow.com.

- Dùng <https://crt.sh/> để tìm subdomain của arrow.com và lưu vào file

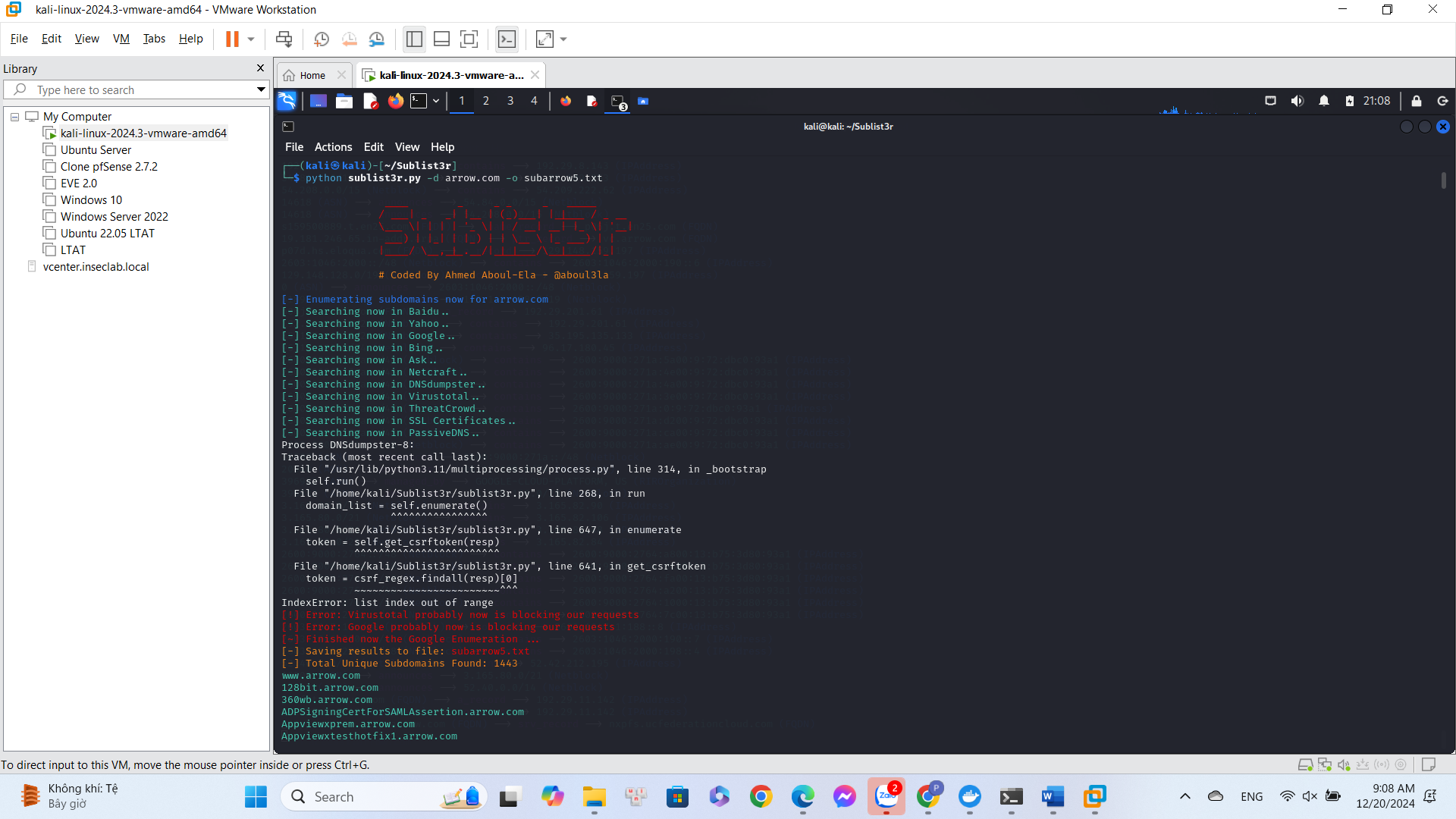
A screenshot of a computer

Description automatically generated

A screen shot of a computer

Description automatically generated

Dùng Sublist3r để tìm subdomain của arrow.com và lưu vào file



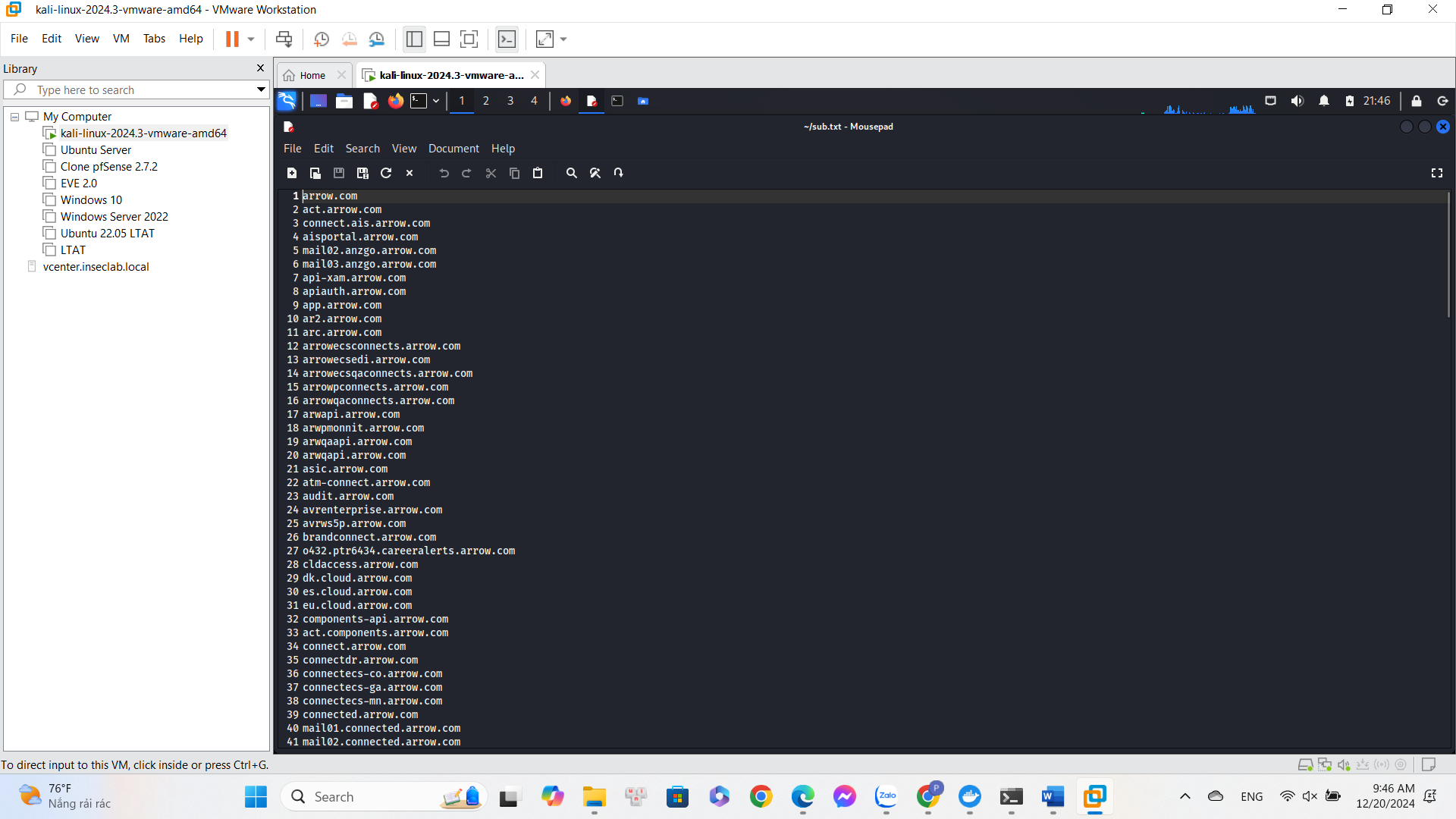
A computer screen with a black screen

Description automatically generated

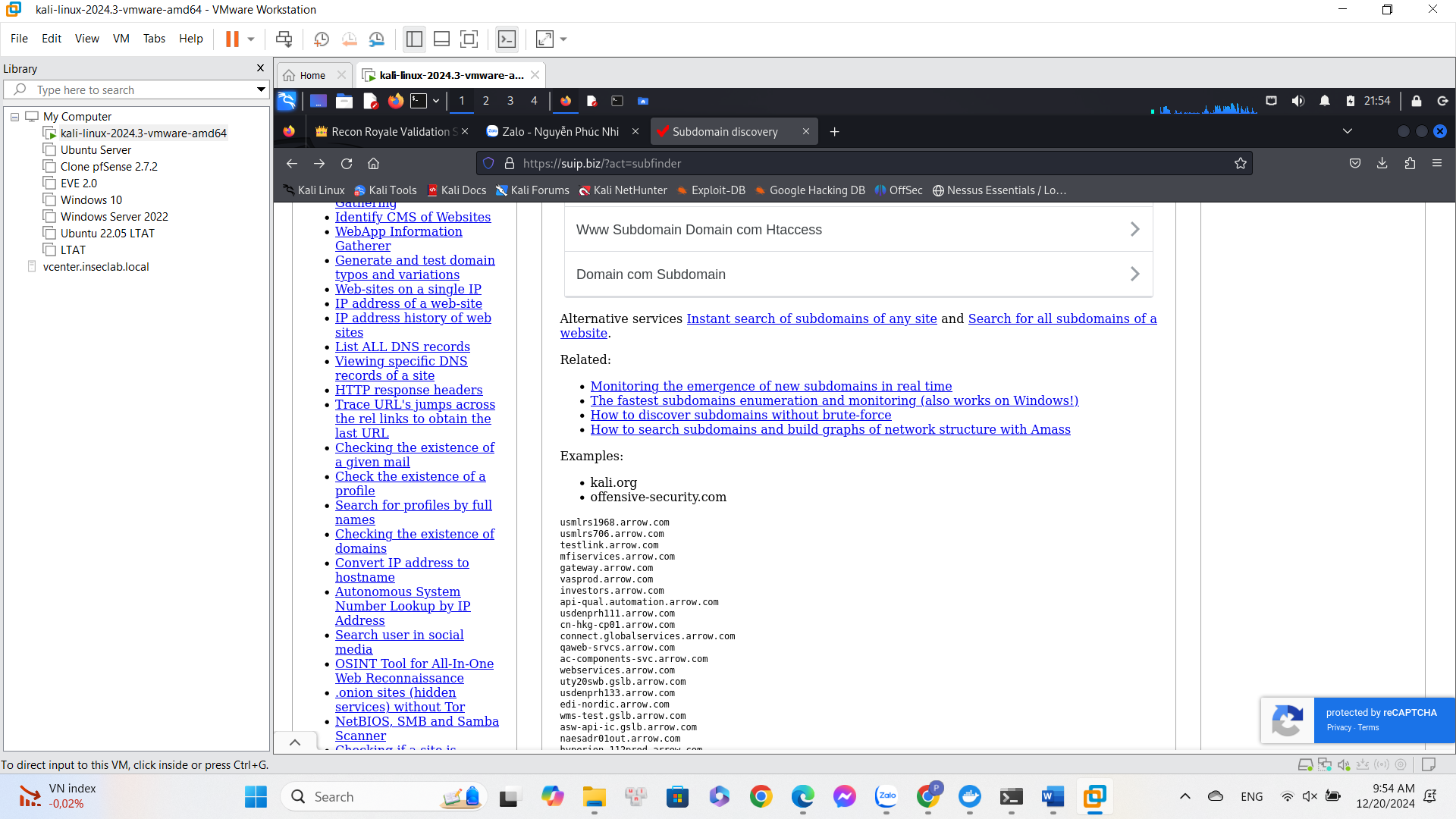
Dùng https://hackertarget.com/ để tìm subdomain của arrow.com và lưu vào file

A screenshot of a computer

Description automatically generated



Dùng https://suip.biz/?act=subfinder để tìm subdomain của arrow.com và lưu vào file

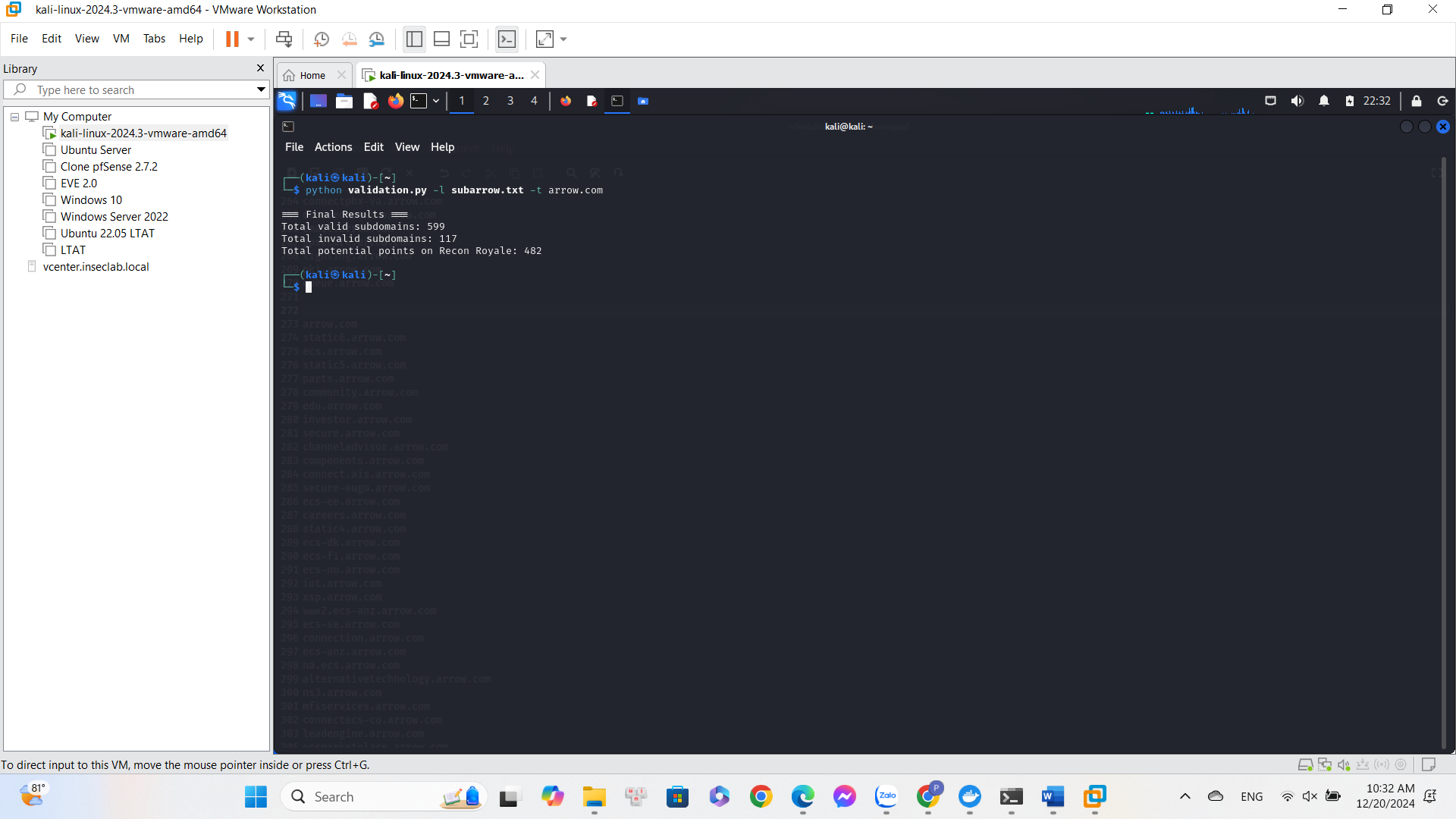


Dùng <https://securitytrails.com/> để tìm subdomain của arrow.com và lưu vào file

A screenshot of a computer

Description automatically generated

Gộp kết quả và lưu vào file subarrow.txt và dùng tool được cung cấp để check số lượng invalid subdomain, ta thấy có 599 invalid subdomain



Bài tập 4, 5:(4đ): Báo cáo lỗ hổng tìm thấy của netsight.apk.

**Lỗ hỏng 1:**

**- Tiêu đề**: Lỗ hổng bảo mật ở LaunchActivity của ứng dụng video\_player.apk

- **Mô tả lỗ hổng**:

  + **Tóm tắt**: Lỗ hổng cho phép ứng dụng video\_player.apk khởi chạy LaunchActivity từ bên ngoài, dẫn đến việc truy cập không mong muốn và yêu cầu quyền đối với album media của người dùng.

**+ Các bước thực hiện và minh chứng:**

1. Khi khởi động ứng dụng thông thường, chỉ hiển thị một video và không cho phép người dùng tương tác.

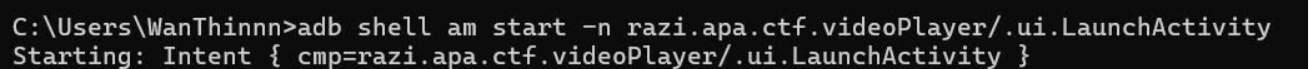
A person in a suit

Description automatically generated

2. Chạy lệnh như hình và quan sát thấy ứng dụng có thể được mở từ một nguồn bên ngoài.

1. A screenshot of a computer program

Description automatically generated



3. Khi chạy lệnh trên, ứng dụng tự động chuyển hướng đến giao diện media và yêu cầu quyền truy cập vào Album Media của người dùng.

A screen shot of a phone

Description automatically generated

- **Mức độ ảnh hưởng của lỗ hổng:** Lỗ hổng này có thể bị lợi dụng để truy cập trái phép vào giao diện media của ứng dụng, dẫn đến rủi ro về quyền riêng tư hoặc xâm phạm dữ liệu nhạy cảm của người dùng.

- **Khuyến cáo khắc phục:**

+ Cấu hình AndroidManifest.xml để hạn chế truy cập từ bên ngoài:

*xml*

*Copy code*

*<activity*

*android:name=".ui.LaunchActivity"*

*android:exported="false">*

*</activity>*

+ Kiểm tra và bảo vệ các quyền nhạy cảm trong ứng dụng, đảm bảo rằng chỉ được yêu cầu khi cần thiết và thông qua sự đồng ý của người dùng.

**Lỗ hỏng 2:**

**- Tiêu đề: WebView Cho Phép Điều Hướng Đến URL Không An Toàn Trong WebviewActivity**

**- Mô tả lỗ hổng**

+ Lỗ hổng này cho phép kẻ tấn công điều hướng ứng dụng đến các URL không an toàn thông qua WebView trong WebviewActivity.

+ Do cấu hình của ứng dụng (android:exported="true") và việc không kiểm tra URL truyền vào, kẻ tấn công có thể sử dụng adb để chuyển hướng ứng dụng đến một trang web độc hại, mở ra nguy cơ đánh cắp dữ liệu hoặc thực hiện các hành động trái phép.

**- Tóm tắt**

**+ Lỗ hổng**: WebviewActivity được định nghĩa trong AndroidManifest.xml với thuộc tính

<activity android:exported="true" android:name="com.tcpip.netsight.WebviewActivity"/>

=> Điều này cho phép các ứng dụng khác hoặc kẻ tấn công sử dụng Intent để khởi chạy activity này. URL được truyền vào không được kiểm tra hoặc xác minh trước khi tải trong WebView.

**+ Hệ quả bảo mật**:

* Kẻ tấn công có thể chuyển hướng ứng dụng đến các trang web giả mạo hoặc độc hại.
* Nguy cơ xảy ra tấn công lừa đảo (phishing), đánh cắp thông tin hoặc thực thi mã JavaScript độc hại.

**- Các bước để thực hiện lại và bằng chứng:**

**+ Bước 1: Kiểm tra thuộc tính android:exported:** Mở file AndroidManifest.xml và xác nhận cấu hình:

<activity android:exported="true" android:name="com.tcpip.netsight.WebviewActivity"/>

**+ Bước 2: Khởi chạy WebviewActivity bằng lệnh ADB**

Sử dụng lệnh sau để điều hướng ứng dụng đến URL không an toàn:

adb shell am start -n com.tcpip.netsight/.WebviewActivity -e url "http://example.com"

**+ Bước 3: Quan sát hành vi**

* Ứng dụng sẽ mở http://example.com trong WebView mà không kiểm tra hoặc cảnh báo.
* Logcat hiển thị như sau: START u0 {dat=http://example.com/... flg=0x10000000 cmp=com.tcpip.netsight/.WebviewActivity} from uid 0

**- Mức độ ảnh hưởng của lỗ hổng**

1. **Tác động bảo mật**:
   * **Tấn công phishing**: Kẻ tấn công có thể chuyển hướng ứng dụng đến trang giả mạo, lừa người dùng nhập thông tin nhạy cảm.
   * **Thực thi JavaScript độc hại**: WebView với setJavaScriptEnabled(true) có thể chạy mã JavaScript nguy hiểm từ trang web độc hại.
   * **Nguy cơ rò rỉ thông tin**: Nếu ứng dụng tải URL không an toàn qua HTTP, dữ liệu có thể bị chặn hoặc thay đổi trong quá trình truyền.
2. **Nguy cơ xảy ra tấn công**: Cao, đặc biệt nếu ứng dụng được cài đặt trên thiết bị của nhiều người dùng mà không kiểm tra đầu vào URL.

**- Khuyến cáo khắc phục**

1. **Kiểm tra và xác minh URL đầu vào**:
   * Thêm kiểm tra trong mã nguồn WebviewActivity để chỉ cho phép các URL đáng tin cậy:
2. **Giới hạn quyền truy cập**: Đặt android:exported="false" trong AndroidManifest.xml nếu không cần cho phép các ứng dụng khác khởi chạy activity này.
3. **Sử dụng HTTPS**: Chỉ tải các URL bắt đầu bằng https:// để tránh bị tấn công MITM.

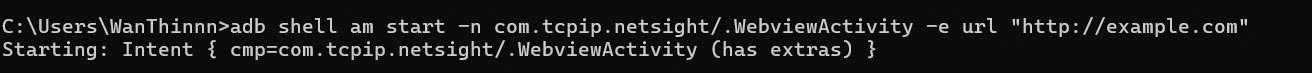
**- Minh chứng:**

A screenshot of a cell phone

Description automatically generated

A computer screen with white text

Description automatically generated



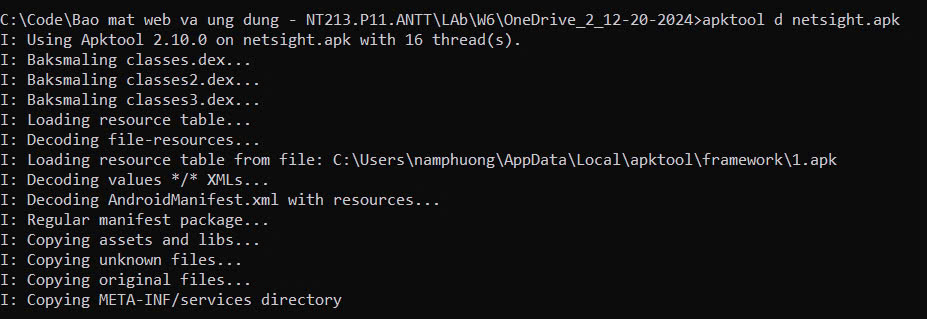
**Lỗ hỏng 3:**

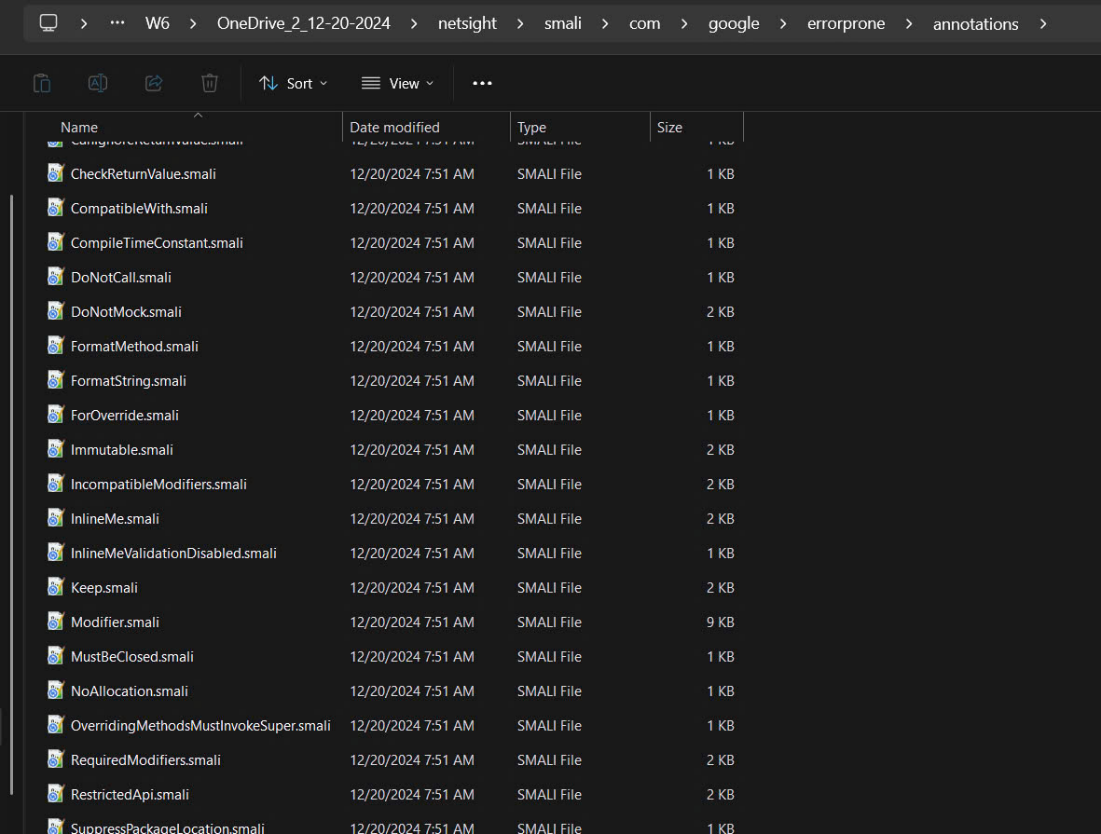
**- Tiêu đề: Lỗ hổng Reverse Engineering ở file netsight.apk**

**- Mô tả lỗ hổng**

+ Có lỗ hổng Reverse Engineering ở file netsight.apk, cụ thể khi sử dụng apktool để decompile file apk này ra và thực hiện truy xuất vào các file smali, thì ta sẽ thấy được rằng file này chưa được làm rối mã, tên các hàm và class để nguyên tên thuận tiện cho việc khai thác

+ Truy cập vào vào các file MainActivity$x, sẽ thấy được các link đường dẫn đến các trang web trong app, từ đó có thể khai thác bằng cách dịch ngược lại nó rồi thay thế nó bằng các đường dẫn độc hại





+ Do cấu hình của ứng dụng (android:exported="true") và việc không kiểm tra URL truyền vào, kẻ tấn công có thể sử dụng adb để chuyển hướng ứng dụng đến một trang web độc hại, mở ra nguy cơ đánh cắp dữ liệu hoặc thực hiện các hành động trái phép.

1. Ghi nội dung công việc, các kịch bản trong bài Thực hành [↑](#footnote-ref-2)