

Lab 2: Information Disclosure	
Name	Đặng Hoàng Nguyên
Student ID	SE171946

**Giới Thiệu:** Trong bài lab này ta sẽ tìm hiểu về lỗ hổng information disclosure

## I. Source code disclosure via backup files:

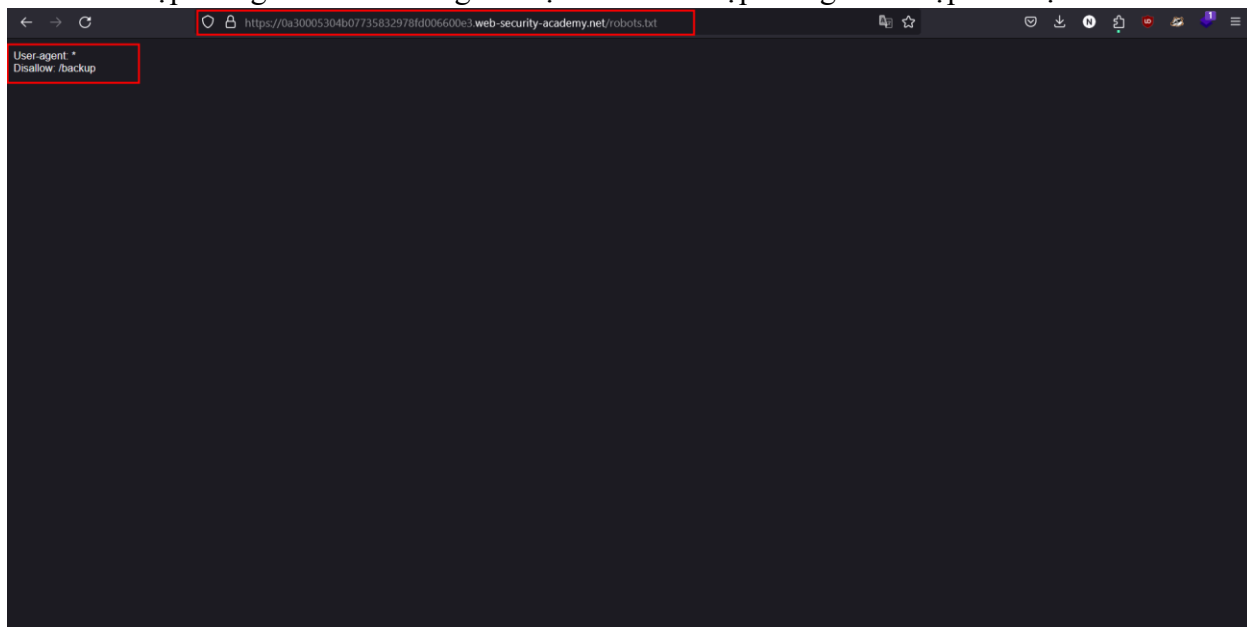
The screenshot shows the PortSwigger Web Security Academy interface. The top navigation bar includes links for Products, Solutions, Research, Academy, and Support. The main content area displays the lab title 'Lab: Source code disclosure via backup files' with an 'APPRENTICE' difficulty level. A description states: 'This lab leaks its source code via backup files in a hidden directory. To solve the lab, identify and submit the database password, which is hard-coded in the leaked source code.' Below the description is an 'ACCESS THE LAB' button. On the left, a sidebar menu lists various topics related to information disclosure. At the bottom, there are sections for 'Solution' and 'Community solutions'.

⇒ Mục tiêu của bài lab này là tìm được **password** của database để submit ở phần **Submit solution**.

⇒ Ta bấm vào **ACCESS THE LAB** để được dẫn đến trang làm bài

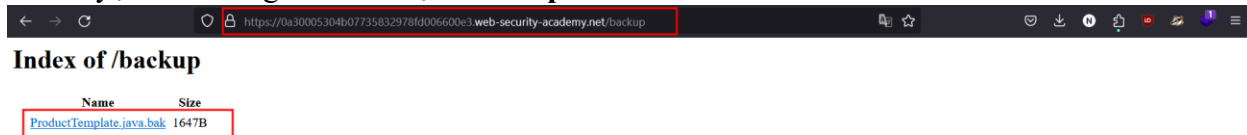
The top part of the screenshot shows the submission page for the 'Source code disclosure via backup files' lab. It includes a 'Submit solution' button and a 'Back to lab description' link. The lab status is 'LAB Not solved'. Below this is a promotional banner for 'WE LIKE TO SHOP' featuring four products: 'Sprout More Brain Power' (\$82.13), 'Caution Sign' (\$38.01), 'Snow Delivered To Your Door' (\$30.38), and 'Hologram Stand In' (\$13.62). Each product has a star rating and a 'View details' button.

1. Đầu tiên khi phân tích một Website ta có thể làm mà công cần scan là thử đọc file **robots.txt** – file văn bản nằm trong thư mục gốc của trang web và cung cấp hướng dẫn cho các công cụ tìm kiếm thu thập thông tin về các trang mà họ có thể thu thập thông tin để lập chỉ mục.



⇒ Có thể thấy ở trên thì có một đường dẫn đến thư mục có tên **/backup**, giờ ta sẽ thử duyệt đến đường dẫn **/backup**

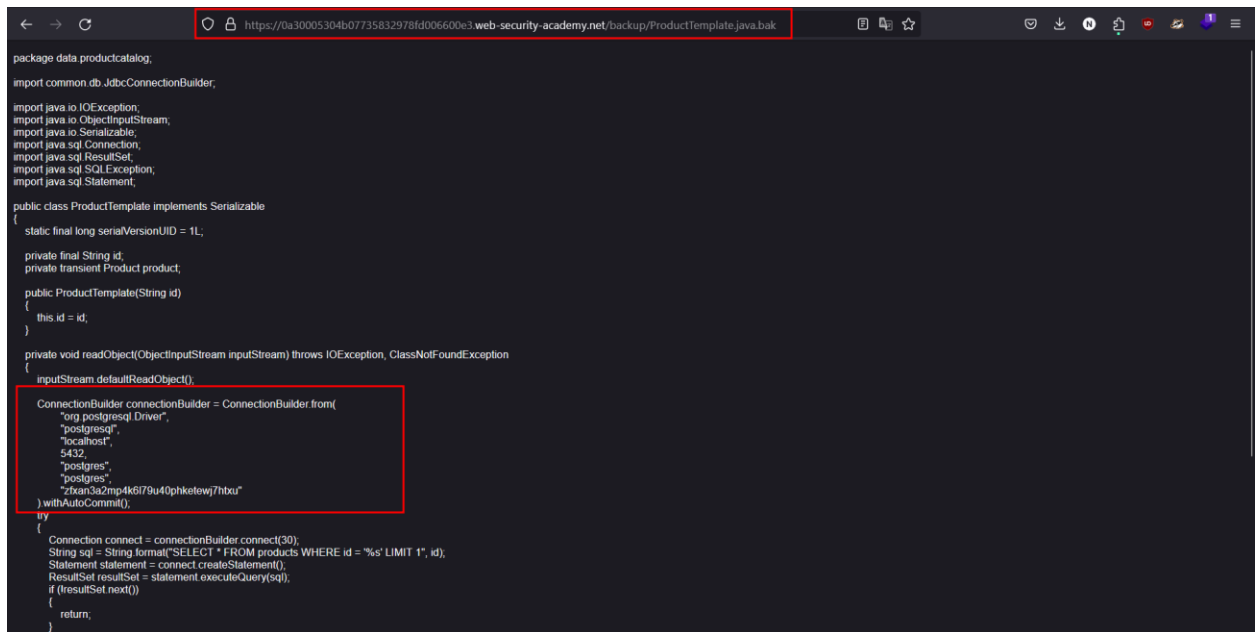
2. Sau khi duyệt đến đường dẫn thư mục **/backup**



⇒ Ta có thể thấy thì tại đây có một file có kích thước **1647 Bytes** có tên là **“ProductTemplate.java.bak”**

⇒ Tiếp theo thì ta sẽ thử mở file này để xem bên trong có gì

3. Click vào file **“ProductTemplate.java.bak”** để mở file



```
package data.productcatalog;

import common.db.JdbcConnectionBuilder;

import java.io.IOException;
import java.io.ObjectInputStream;
import java.io.Serializable;
import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;

public class ProductTemplate implements Serializable
{
    static final long serialVersionUID = 1L;

    private final String id;
    private transient Product product;

    public ProductTemplate(String id)
    {
        this.id = id;
    }

    private void readObject(ObjectInputStream inputStream) throws IOException, ClassNotFoundException
    {
        inputStream.defaultReadObject();

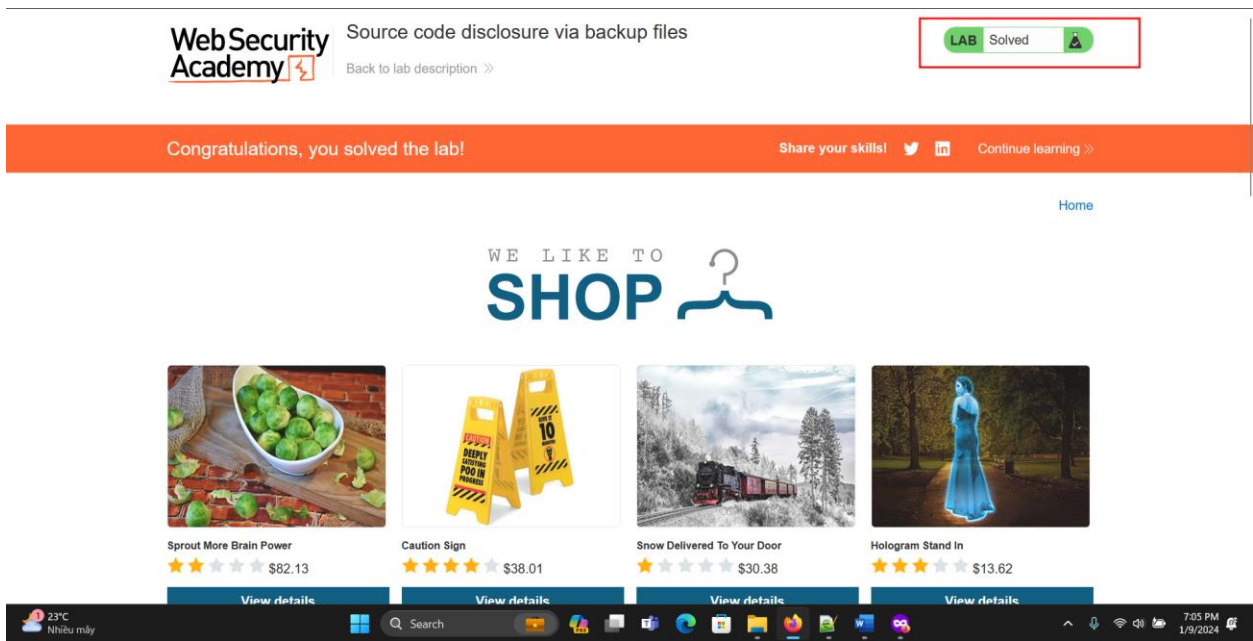
        ConnectionBuilder connectionBuilder = ConnectionBuilder.from(
            "org.postgresql.Driver",
            "postgresql",
            "localhost",
            5432,
            "postgres",
            "postgres",
            "zfxan3a2mp4k6l79u40phketewj7htxu"
        ).withAutoCommit();

        try
        {
            Connection connect = connectionBuilder.connect(30);
            String sql = String.format("SELECT * FROM products WHERE id = '%s' LIMIT 1", id);
            Statement statement = connect.createStatement();
            ResultSet resultSet = statement.executeQuery(sql);
            if (resultSet.next())
            {
                return;
            }
        }
    }
}
```

- ⇒ Đây là một đoạn code mà developer đã backup lại, tuy nhiên họ không lưu trữ ở ở khác mà lưu trực tiếp tại Web Server.
- ⇒ Có thể thấy được đây mà một đoạn code Java xử lý việc kết nối với database thông qua thư viện **java.sql.Connection** của Java và từ đó lấy được thông tin của sản phẩm qua **ID**.
- ⇒ Và từ hình trên thì ta có thông tin như sau:

<b>Driver:</b> org.postgresql.Driver
<b>Database Type:</b> PostgreSQL
<b>Host:</b> localhost
<b>Port:</b> 5432
<b>Username:</b> postgres
<b>Password:</b> zfxan3a2mp4k6l79u40phketewj7htxu
<b>AutoCommit:</b> Enabled (withAutoCommit())

- ⇒ Vậy ta đã biết được thông tin password của database, nên ta sẽ submit password trên



⇒ Vậy là ra đã solve được bài lab.

## II. Information disclosure in version control history:

Products Solutions Research Academy Support

Dashboard Learning paths Latest topics All content Hall of Fame Get started Get certified

Web Security Academy > Information disclosure > Exploiting > Lab

### Lab: Information disclosure in version control history

**PRACTITIONER** LAB Not solved

This lab discloses sensitive information via its version control history. To solve the lab, obtain the password for the `administrator` user then log in and delete the user `carlos`.

ACCESS THE LAB

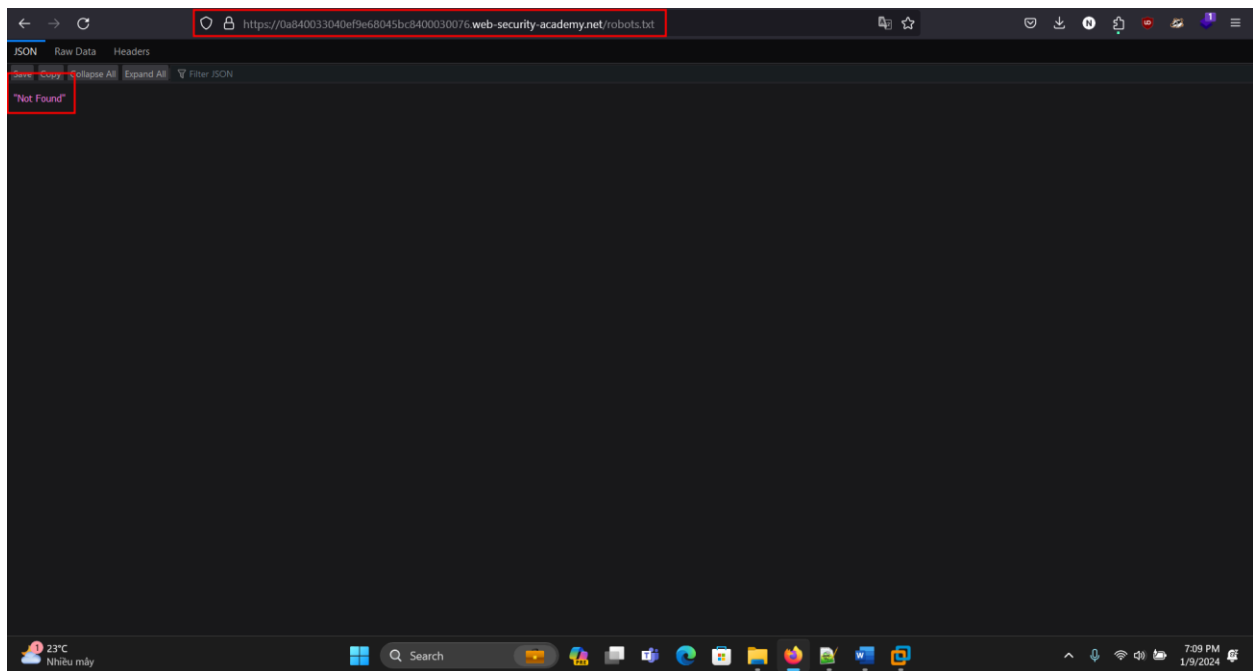
Solution

Community solutions

⇒ Mục tiêu của bài lab này là lấy được password của user **administrator** sau đó login và xóa user **carlos**.

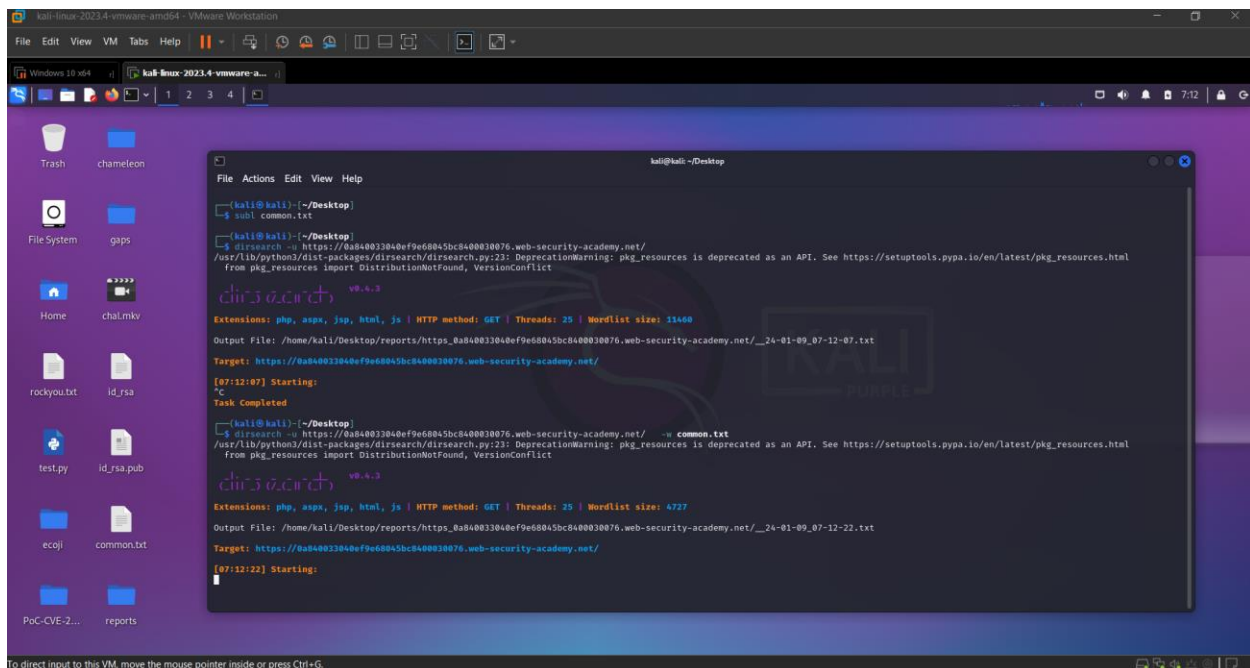


1. Đầu tiên cũng như bài lab ở trên thì ta cũng sẽ kiểm tra thử file **robots.txt**

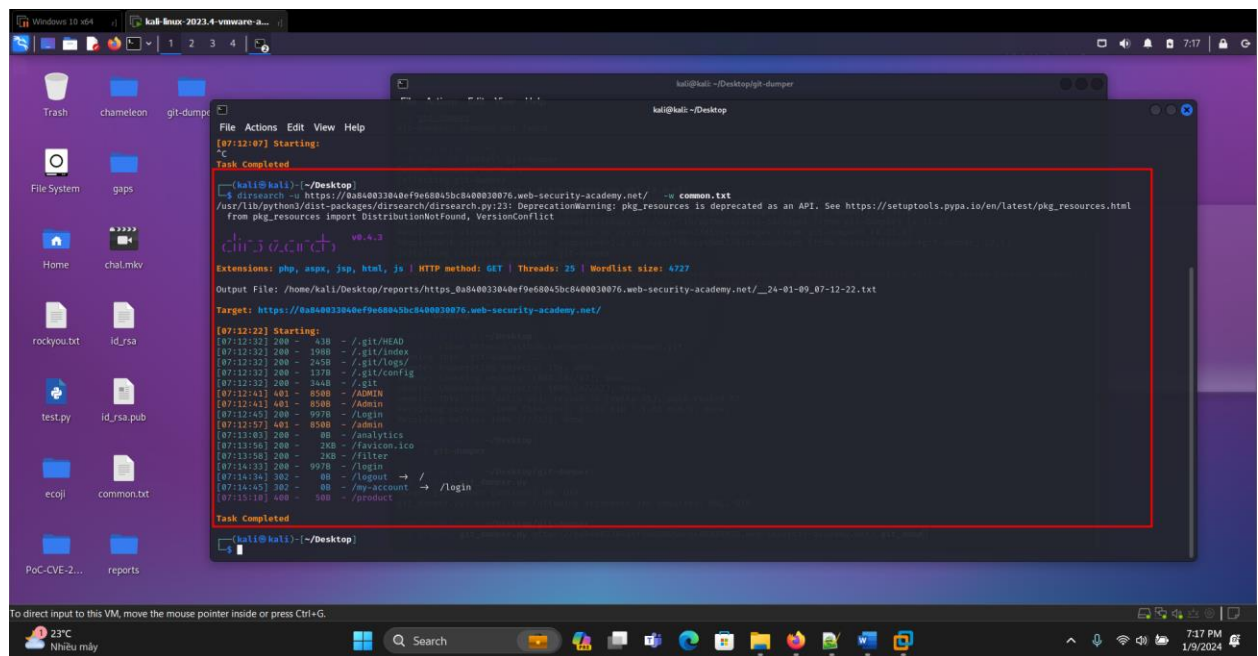


⇒ Tuy nhiên có thể thấy được rằng server trả response về “**Not Found**” => file này không tồn tại

2. Giờ thì ta sẽ sử dụng công cụ Dirsearch và Wordlists đã được đề cập ở trên để scan

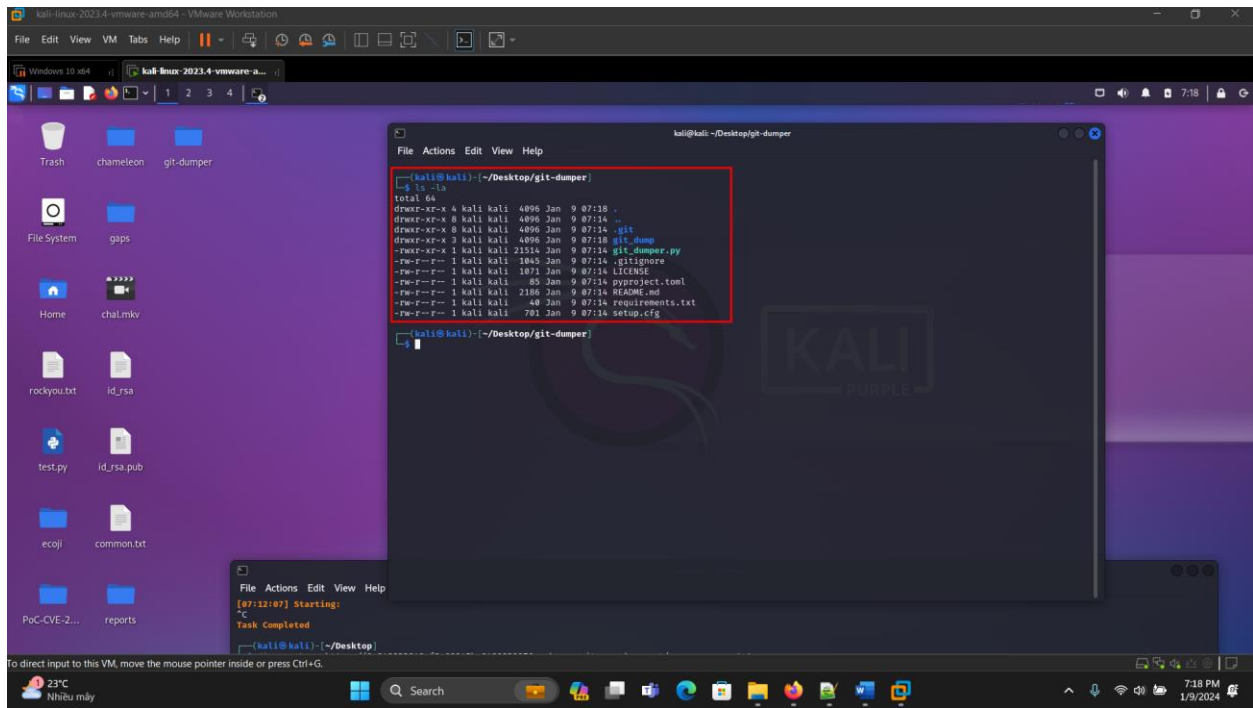
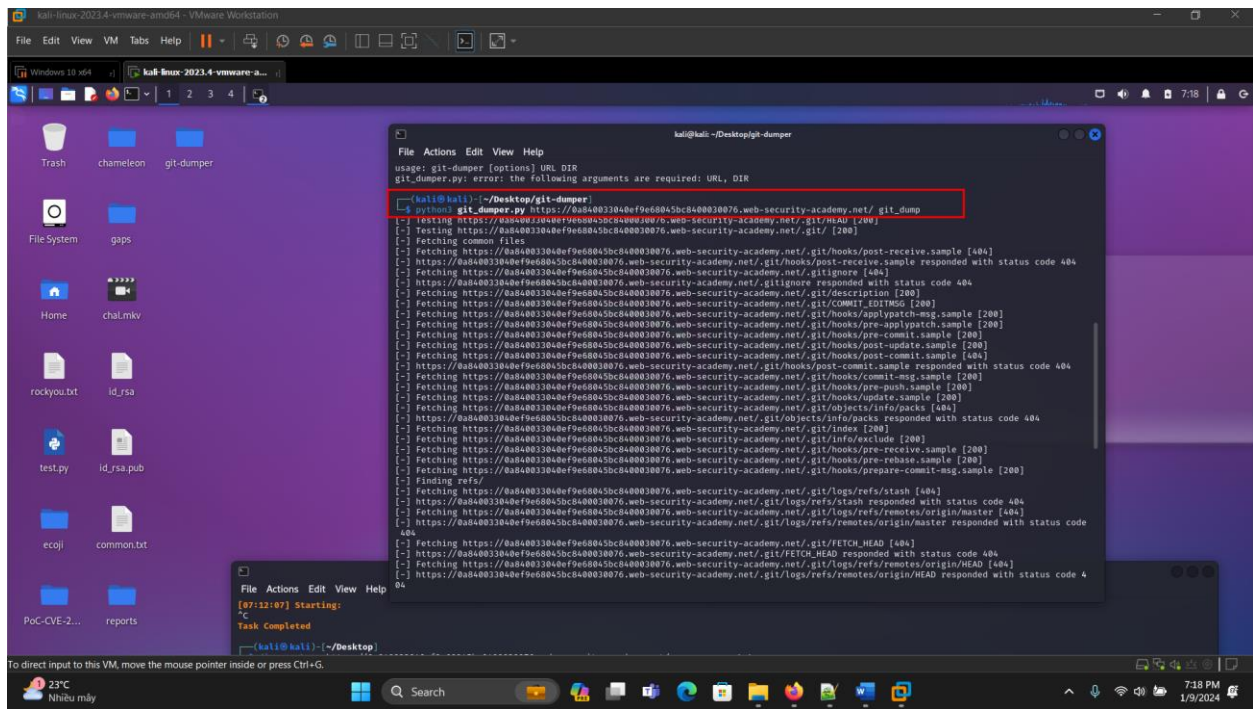


⇒ Từ kết quả scan ra được thì có thể thấy một đường dẫn `/.git`. Hiểu sơ qua đây là một folder chứa chứa các dữ liệu mà ta thao tác với repo mà ta remote



3. Từ thông tin trên, ta sẽ sử dụng một công cụ **git-dumper** ([link](#)) để tải toàn bộ data mà người dùng đã thao tác với git. Sau khi sử dụng công cụ thì ta sẽ được





```
kali-linux-2023.4-vmware-amd64 - VMware Workstation
File Edit View VM Tabs Help
Windows 10 x64 kali-linux-2023.4-vmware-amd64
kali@kali: ~/Desktop/git-dumper
File Actions Edit View Help
[kali@kali]~/Desktop/git-dumper
$ cd .git
[kali@kali]~/Desktop/git-dumper/.git
$ ls
branches config description HEAD hooks index info logs objects packed-refs refs
[kali@kali]~/Desktop/git-dumper/.git
$ cat ..
[kali@kali]~/Desktop/git-dumper
$ git log
commit 32d67a26ca08ee6a26487f98c042072cd3d66 (HEAD -> master, tag: 1.0.0, origin/master, origin/HEAD)
Author: Maxime Arthaud <maxime@arthaud.me>
Date: Fri May 6 21:39:31 2022 -0700

Support more recent version of dulwich (thanks to #33)

commit bfcb39a3258281a79085a967b1f1b77434471
Merge: 2d6bcf5 fee0355
Author: Alexandre Grynnchuk <agrynnchuk@gmail.com>
Date: Fri May 6 21:31:17 2022 -0700

Merge pull request #33 from ph20/dev/replace_deprecated_dulwich_index_iterblobs

Fix unsupported dulwich.Index.iterblobs method

commit fee03558a163348a8750940baa0e2031c1d6cd9
Author: Alexandre Grynnchuk <agrynnchuk@gmail.com>
Date: Thu May 5 01:05:26 2022 -0800

Bump version

commit 21123d8cc162d18d21d63c7209020b1e748eb6
Author: Alexandre Grynnchuk <agrynnchuk@gmail.com>
Date: Thu May 5 00:48:17 2022 -0800

Fix unsupported dulwich.index.Index.iterblobs method

commit 2d6bcf5a2e0f97aac08096ad294bdfab988e1601
Author: Maxime Arthaud <maxime@arthaud.me>
Date: Tue Apr 5 09:44:31 2022 -0700

Use the recommended setup.cfg file to package the project

commit 0e68785556ff9d0f07e42feb34cc33d832879692
Merge: 5036099 36a7a29
Author: Maxime Arthaud <maxime@arthaud.me>
```

⇒ Ta có thể thấy sử dụng câu lệnh git log có thể thấy có khá nhiều commit, chúng ta xem thử từng commit xem như thế nào

4. Git có một tính năng là lưu lại toàn bộ thông tin mà người dùng đã chỉnh sửa trong quá trình remote -> push lên Git. Để kiểm tra thì log thay đổi của file thì ta sẽ dùng lệnh sau:

**git log -p admin.conf**

⇒ Từ hình ảnh trên thì ta có thể thấy được các thông tin đã được thay đổi của file.

```
kali-linux-2023.4-vmware-amd64 - VMware Workstation
File Edit View VM Tabs Help
Windows 10 x64 kali-linux-2023.4-vmware-amd64
kali@kali: ~/Desktop/git-dumper/git_dump
File Actions Edit View Help
+ [options.entry_points]
+ console_scripts =
+ git-dumper = git_dumper:main
[kali@kali]~/Desktop/git-dumper
$ ls
git_dump git_dumper.py LICENSE pyproject.toml README.md requirements.txt setup.cfg
[kali@kali]~/Desktop/git-dumper
$ cd git_dump
[kali@kali]~/Desktop/git-dumper/git_dump
$ ls
admin.conf admin_panel.php
[kali@kali]~/Desktop/git-dumper/git_dump
$ git log -p admin.conf
commit 2ab0c061d8bd3cc7bfc09ca12713b96991b0 (HEAD -> master)
Author: Carlos Montoya <carlos@evil-user.net>
Date: Tue Jun 23 14:05:07 2020 -0800

Remove admin password from config

diff --git a/admin.conf b/admin.conf
index 78f9000..21023f1 100644
--- a/admin.conf
+++ b/admin.conf
@@ -1,2 +1,2 @@
-ADMIN_PASSWORD=dghkpkrg78gyvbowm84
+ADMIN_PASSWORD=env('ADMIN_PASSWORD')

commit f314498e2f6e1985ead5888dbedaddb9c4043e
Author: Carlos Montoya <carlos@evil-user.net>
Date: Mon Jun 22 16:23:42 2020 -0800

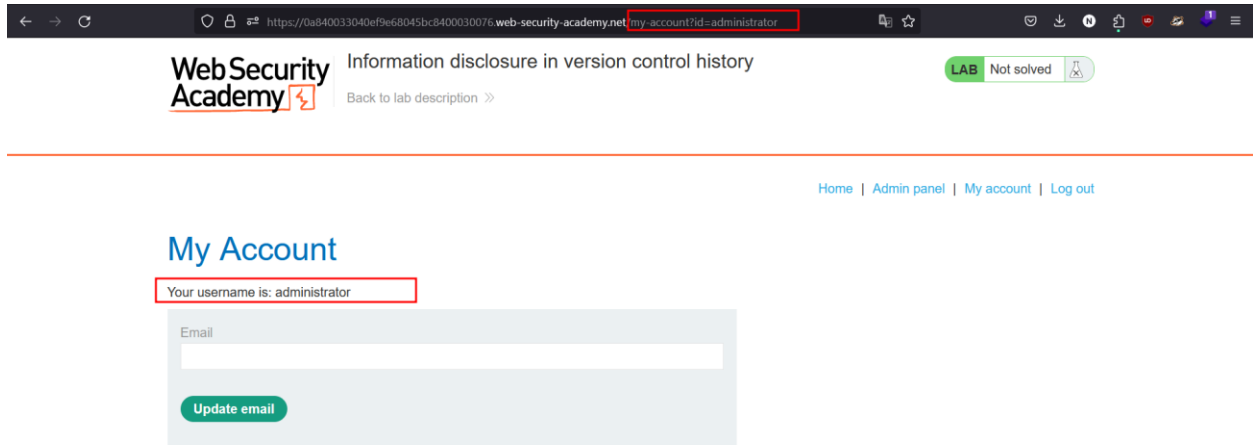
Add skeleton admin panel

diff --git a/admin.conf b/admin.conf
new file mode 100644
index 0000000..78f9000
--- /dev/null
+++ b/admin.conf
@@ -0,0 +1 @@
+ADMIN_PASSWORD=dghkpkrg78gyvbowm84
[kali@kali]~/Desktop/git-dumper/git_dump
```



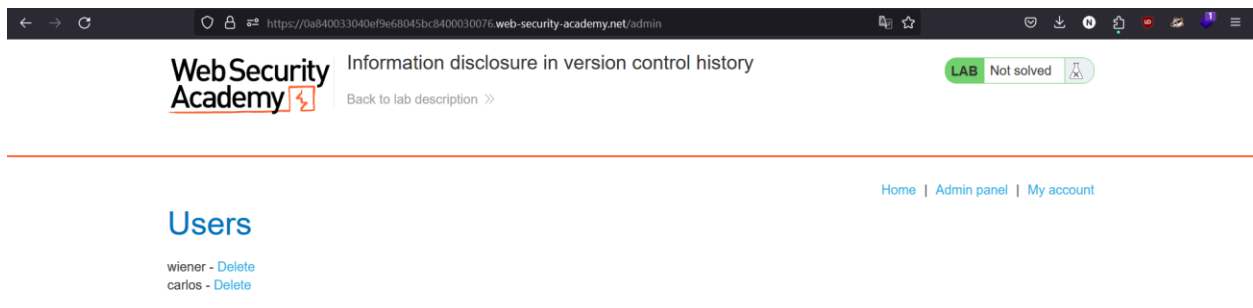
⇒ Từ đây ta cũng biết được password của user **administrator** là **dghkpkg78ggvbowbn84**

5. Giờ ta sẽ thử đăng nhập



⇒ Có thể thấy được là ta đã login thành công vào user **administrator**

6. Tiếp theo ta sẽ thử xóa user Carlos để hoàn thành lab



Web Security Academy Information disclosure in version control history

LAB Solved

Congratulations, you solved the lab!

Share your skills! Continue learning >>

User deleted successfully!

Home | Admin panel | My account

Users

wiener - Delete

23°C Nhiều mây

Search

7:25 PM 1/9/2024

⇒ Vậy là ta đã hoàn thành lab