Lab exercise 1

Task description

Connect using netcat to `www.utcluj.ro:80`, send a GET request to fetch data from `/universitatea/educatie/`, and explain the result. Obtain the same content using wget and compare results. Follow the instructions from the page accessed with netcat; what is the result page containing?

Hint: you can use html2text to remove HTML tags and see only the actual text content from the document.

Task solving

- Content using ncat:

```
┌──(kali㉿kali)-[~]
└─$ nc www.utcluj.ro 80
GET /universitatea/educatie/ HTTP/1.1
HOST: www.utcluj.ro

HTTP/1.1 301 Moved Permanently
Date: Sun, 09 Mar 2025 14:27:52 GMT
Server: Apache/2.2.22 (Ubuntu)
Location: https://www.utcluj.ro/universitatea/educatie/
Vary: Accept-Encoding
Content-Length: 332
Content-Type: text/html; charset=iso-8859-1

<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>301 Moved Permanently</title>
</head><body>
<h1>Moved Permanently</h1>
<p>The document has moved <a
href="https://www.utcluj.ro/universitatea/educatie/">here</a>.</p>
<hr>
<address>Apache/2.2.22 (Ubuntu) Server at www.utcluj.ro Port 80</address>
</body></html>
```

- Content using wcat:

```
┌──(kali㉿kali)-[~]
└─$ wget www.utcluj.ro
--2025-03-09 10:39:28--  http://www.utcluj.ro/
Resolving www.utcluj.ro (www.utcluj.ro)... 193.226.5.7, 2001:b30:1802:2::7
Connecting to www.utcluj.ro (www.utcluj.ro)|193.226.5.7|:80... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://www.utcluj.ro/ [following]
--2025-03-09 10:39:28--  https://www.utcluj.ro/
Connecting to www.utcluj.ro (www.utcluj.ro)|193.226.5.7|:443... connected.
```

```
ERROR: The certificate of ‘www.utcluj.ro’ is not trusted.
ERROR: The certificate of ‘www.utcluj.ro’ doesn't have a known issuer.
```

Lab exercise 2

Task description

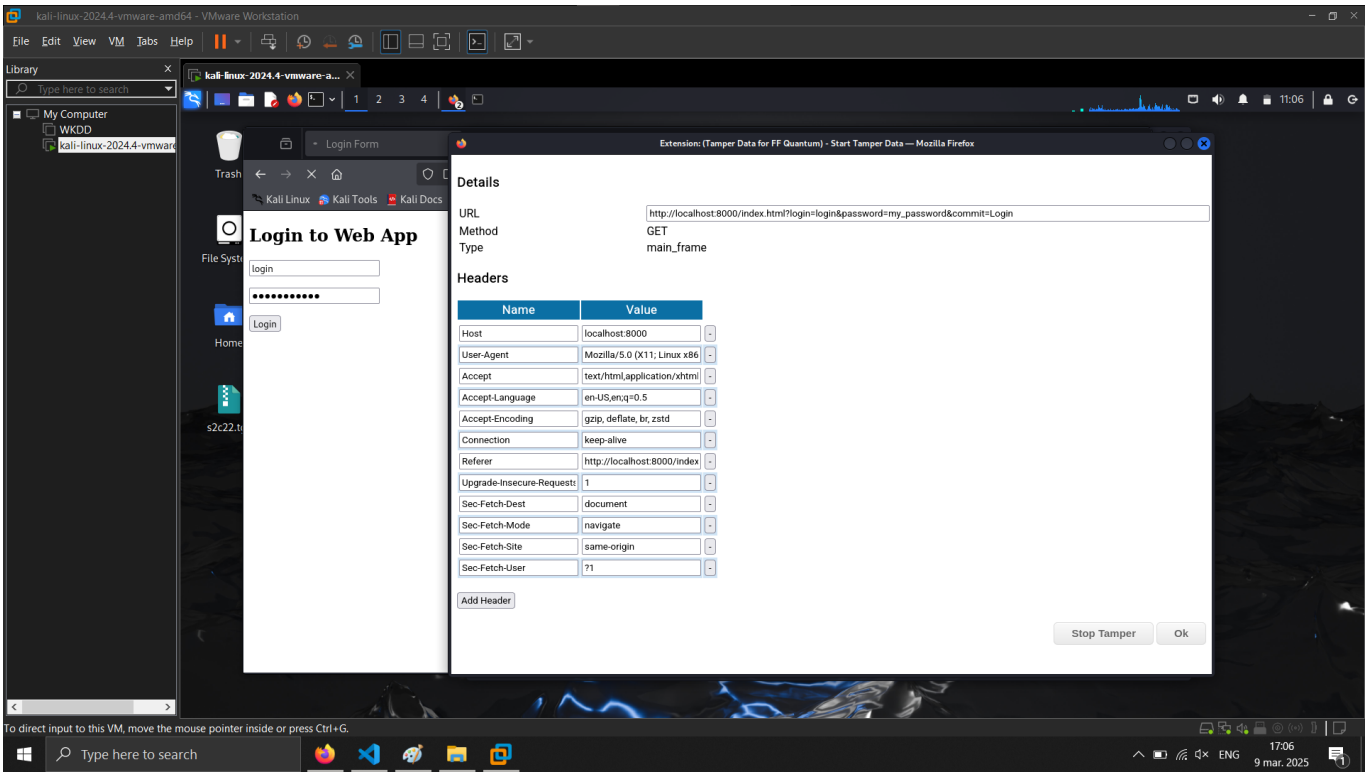
Write the following HTML code, save it in /var/www/html as test.html, and open it in a browser:

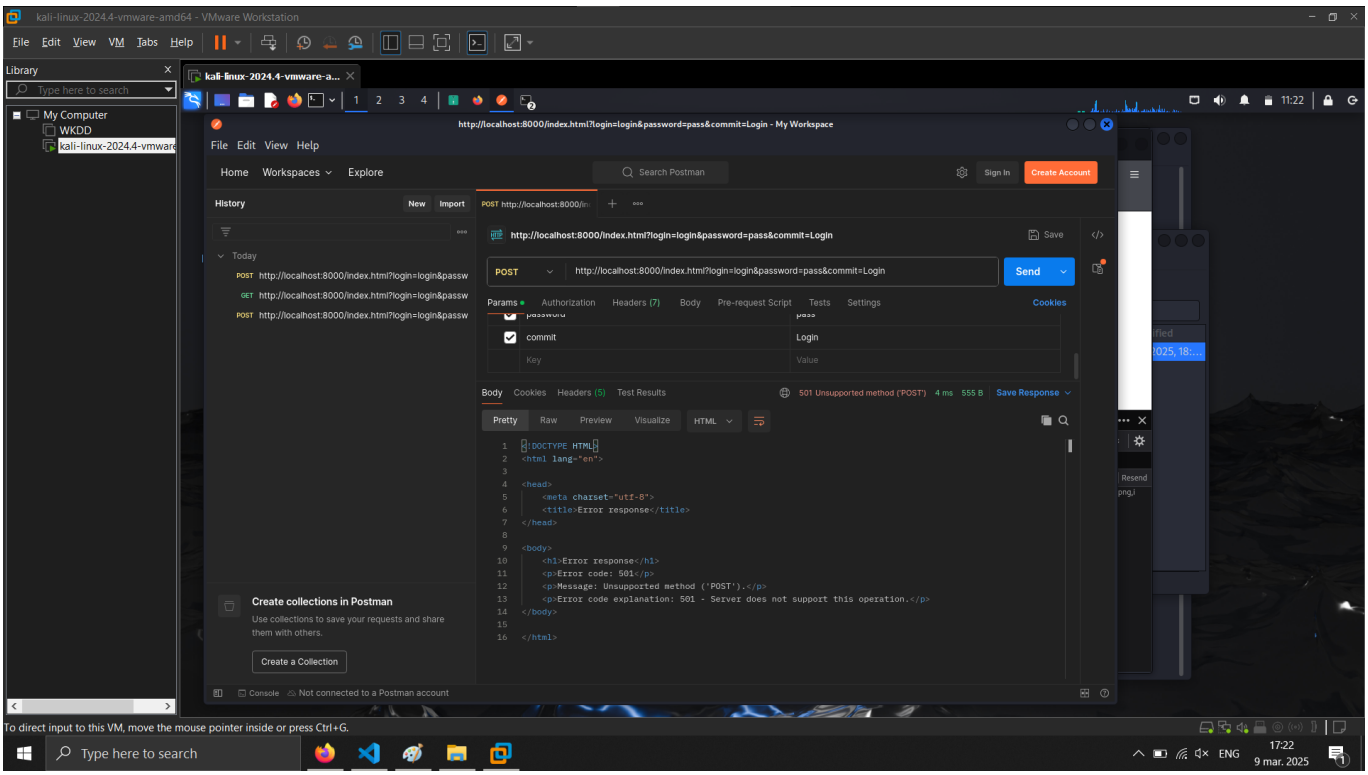
```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
<title>Login Form</title>
</head>
<body>
<section class="container">
<div class="login">
<h1>Login to Web App</h1>
<form method="get" action="index.html">
<p><input type="text" name="login" placeholder="Username or Email"></p>
<p><input type="password" name="password" placeholder="Password"></p>
<p class="submit"><input type="submit" name="commit" value="Login"></p>
</form>
</div>
</section>
</body>
</html>
```

Access the page using a browser: http://localhost/test.html. Click the Login button, intercept the request using Tamper Dev, and analyze the URL and headers. Modify the request type from GET to POST, intercept the request again, and explain the differences.

Hint: to complete this exercise, you need a locally installed web server. Installing and configuring a complex server (e.g., Apache) is not necessary; instead, you can use built-in extensions from Python or Node. For example, to serve resources from a directory using Python, run the following command in that directory: python3 -m http.server. This will start a mini web server on the default port 8000. You can access the web page by navigating to: http://localhost:8000/

Task solving





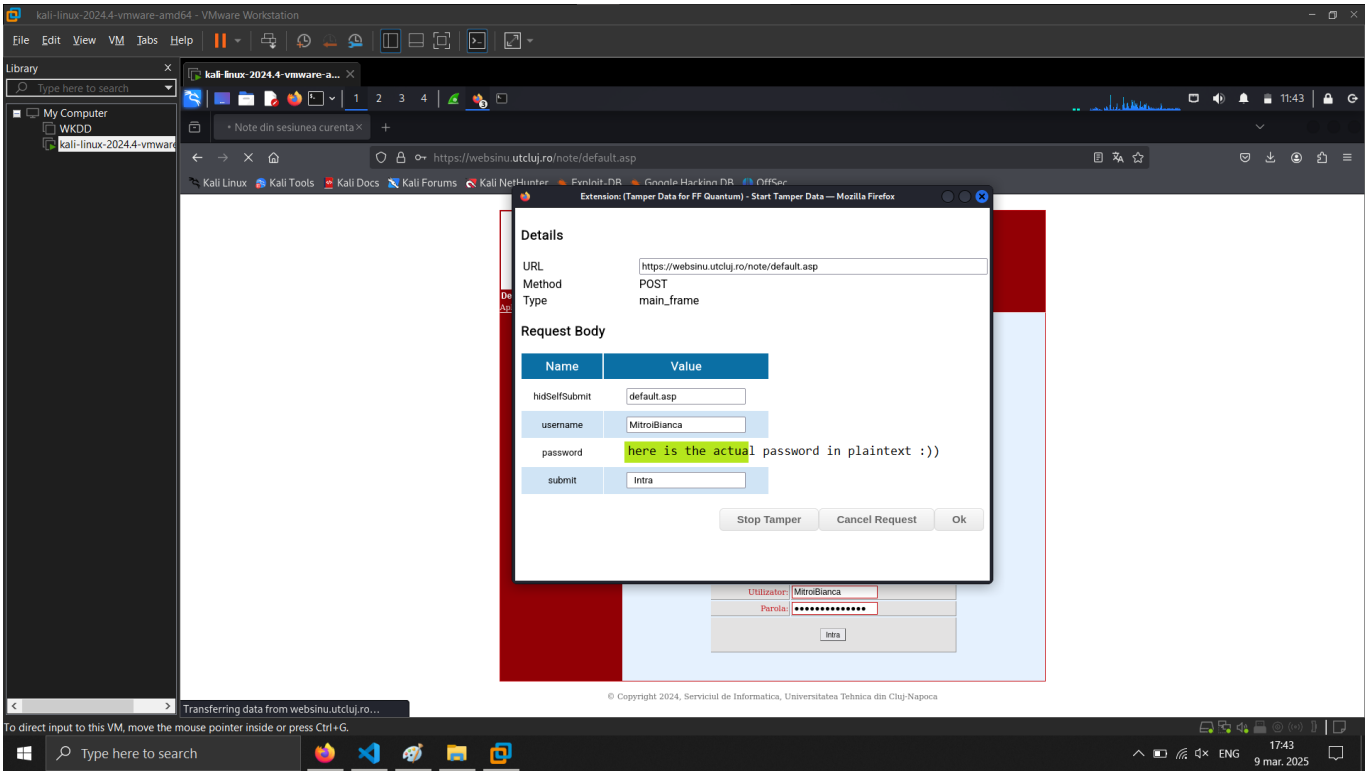
Via Temper Data, the GET request has in url the credentials. When we try to change the method, the server throws an error page.

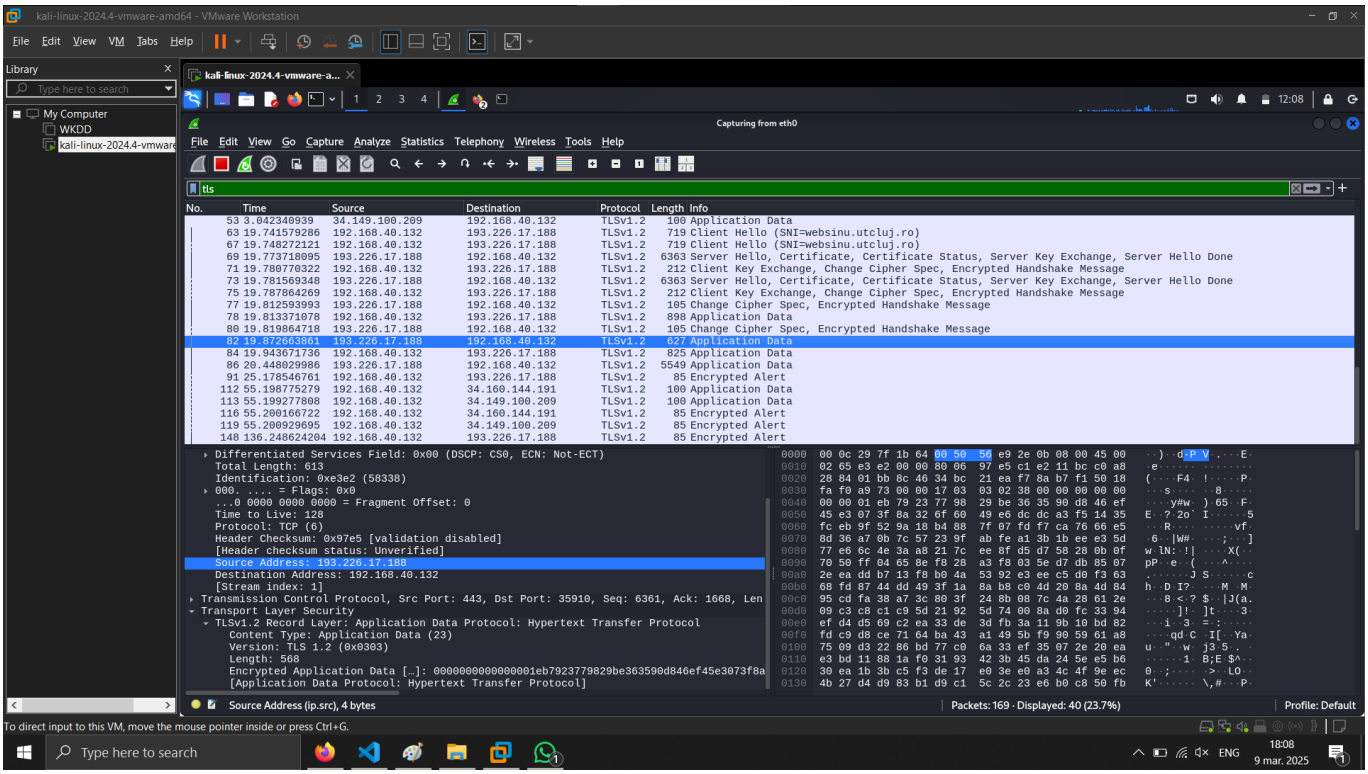
Lab exercise 3

Task description

Authenticate on https://websinu.utcluj.ro and analyze transmitted packets using Wireshark. Observe username and password parameters. Perform the same research using Tamper Dev and explain differences between browser- level parameters and intercepted packets.

Task solving





With Temper Data, the credentials are shown in plaintext because they are taken before they are sent via https. With wireshark the packages can be seen only. The credentials, headers and other content are sent via https and they are encrypted.