

# Authentication Flaws

## 1. Password Strength

The Accounts of your Webapplication are only as save as the passwords. For this exercise, your job is to test several passwords on <https://www.cnlab.ch/codecheck>. You must test all 5 passwords at the same time...

**On your applications you should set good password requirements!**

How much time you need for these passwords?

Password = 123456	<input type="text"/>	seconds
Password = abzfz	<input type="text"/>	seconds
Password = a9z1ez	<input type="text"/>	hours
Password = aB8fEz	<input type="text"/>	days
Password = z8!E?7	<input type="text"/>	days

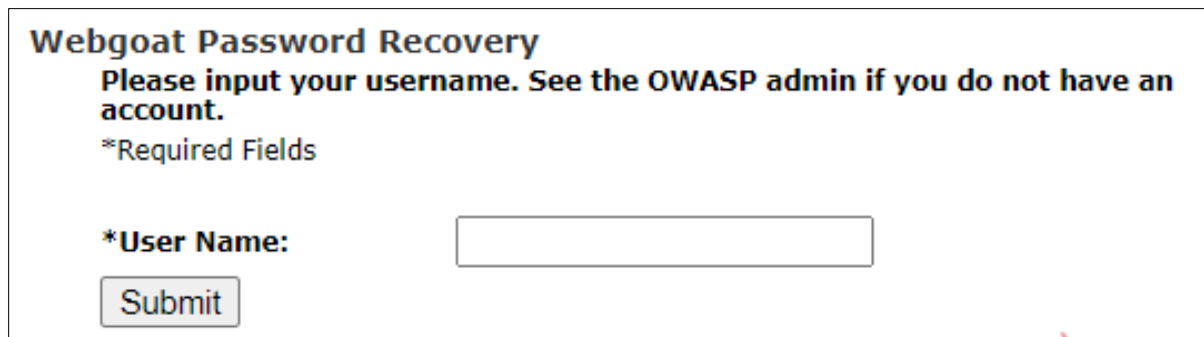
Bài này chỉ ra lý do tại sao mà chúng ta nên đặt mật khẩu 1 cách cẩn thận và khó để BruteForce ra.

Chỉ dẫn đưa ra cho ta 1 trang web, trang web đó chính là trang dùng để dự đoán thời gian và khả năng vét cạn ra mật khẩu của chúng ta. Hiện tại thì website đó đã chết rồi, không thể sử dụng vào bài tập được, vì phải có nó ta mới dùng để dự đoán được kết quả của 5 mật khẩu mà bài tập đưa ra.

Bạn cũng có thể thử bằng các trang web khác đang hoạt động bình thường hiện nay với từ khóa tìm kiếm "How strong is my password". Tất nhiên, không thể dùng những trang web đó vào bài này được đâu, vì mỗi trang khác nhau sẽ có các thuật toán dự đoán khác nhau.

## 2. Forgot Password

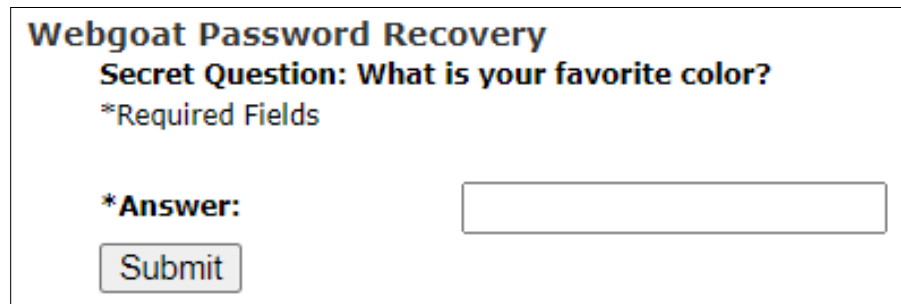
Các ứng dụng web thường xuyên cung cấp cho người dùng khả năng lấy lại mật khẩu đã quên. Thật không may, nhiều ứng dụng web không thực hiện đúng cơ chế này. Thông tin cần thiết để xác minh danh tính của người dùng thường quá đơn giản. Người dùng có thể lấy lại mật khẩu nếu trả lời đúng câu hỏi bí mật, hơn hết, 1 số câu hỏi bí mật rất dễ đoán ra kết quả.



**Webgoat Password Recovery**  
**Please input your username. See the OWASP admin if you do not have an account.**  
\*Required Fields

\*User Name:

Bài tập yêu cầu ta lấy lại mật khẩu cho người dùng admin.



**Webgoat Password Recovery**  
**Secret Question: What is your favorite color?**  
\*Required Fields

\*Answer:

Okay, nhập admin thì nó yêu cầu trả lời 1 câu hỏi "Màu sắc yêu thích của bạn là gì?" Câu hỏi hết sức đơn giản, màu thì chỉ có mười mấy màu đổ lại thôi, BruteForce 1 xúu là ra (tất nhiên là k chơi mấy màu kiểu mã HEX, màu kiểu vàng nhạt, vàng đậm đâu nhé). Đây là 1 câu hỏi đóng, kết quả nó chỉ cho phép người dùng chọn các màu cơ bản nhất trên thế giới, đây chính là vấn đề trong việc sử dụng cơ chế xác thực bằng câu hỏi bảo mật.

Ngồi mò 1 lúc thì đáp án là màu xanh lá cây "**Green**" nhé.

**\* Congratulations. You have successfully completed this lesson.**

### Webgoat Password Recovery

**For security reasons, please change your password immediately.**

**Results:**

Username: admin

Color: green

Password: 2275\$starBo0rn3

## 3. Basic Authentication

Chứng thực cơ bản. Hiểu 1 cách đơn giản thì nó là phương thức để xác thực người dùng khi truy cập tài nguyên thông qua HTTP(s). Nó chứng thực bằng cách:

+Thông tin đăng nhập được gửi kèm theo mỗi request.

+Cấu trúc header sẽ có thêm : Authorization: Basic (Base64 encode username:password).

Okay, hiểu vậy là đủ để làm bài này rồi.

Basic Authentication is used to protect server side resources. The web server will send a 401 authentication request with the response for the requested resource. The client side browser will then prompt the user for a user name and password using a browser supplied dialog box. The browser will base64 encode the user name and password and send those credentials back to the web server. The web server will then validate the credentials and return the requested resource if the credentials are correct. These credentials are automatically resent for each page protected with this mechanism without requiring the user to enter their credentials again.

**General Goal(s):**

For this lesson, your goal is to understand Basic Authentication and answer the questions below.

What is the name of the authentication header:

What is the decoded value of the authentication header:

Submit

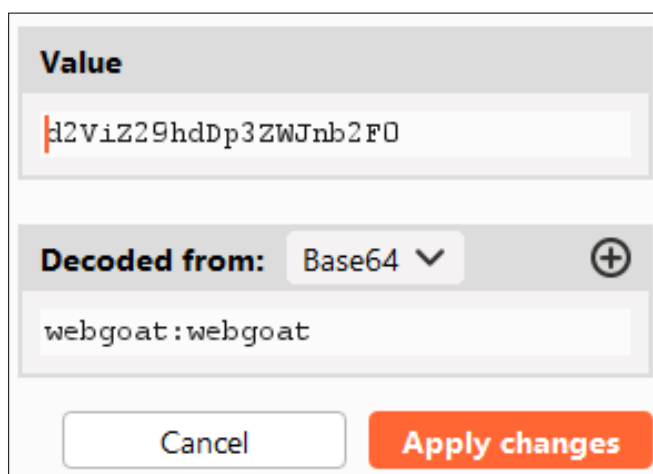
Bài tập yêu cầu ta kiểm tra thông tin chứng thực của website và báo cáo lại chúng để hoàn thành. Các thông tin này xem trong request của gói tin HTTP nhé.

Dùng BurpSuite bắt 1 gói tin lại.

```
1 POST /WebGoat/attack?Screen=111&menu=500 HTTP/1.1
2 Host: 192.168.48.131
3 Content-Length: 34
4 Cache-Control: max-age=0
5 Authorization: Basic d2ViZ29hdDp3ZWJnb2F0
6 Upgrade-Insecure-Requests: 1
7 Origin: http://192.168.48.131
8 Content-Type: application/x-www-form-urlencoded
9 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/
  Safari/537.36
10 Accept:
  text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,i
  d-exchange;v=b3;q=0.9
11 Referer: http://192.168.48.131/WebGoat/attack?Screen=111&menu=500
12 Accept-Encoding: gzip, deflate
13 Accept-Language: en-US,en;q=0.9
14 Cookie: JSESSIONID=C12F921B3738ACE62521B4C6F7A2BD44
15 Connection: close
16
17 header=123&value=123&SUBMIT=Submit
```

Tiêu đề chứng thực: Authorization

Giá trị của tiêu đề chứng thực sau khi giải mã: webgoat:webgoat



The screenshot shows a dialog box titled "Value". It contains a text input field with the value "d2ViZ29hdDp3ZWJnb2F0". Below this, there is a section labeled "Decoded from:" with a dropdown menu set to "Base64" and a plus icon. The decoded value "webgoat:webgoat" is displayed in a text input field below the dropdown. At the bottom of the dialog, there are two buttons: "Cancel" and "Apply changes".

**General Goal(s):**

For this lesson, your goal is to understand Basic Authentication and answer the questions below.

**\* Congratulations, you have figured out the mechanics of basic authentication. - Now you must try to make WebGoat reauthenticate you as: - username: basic - password: basic. Use the Basic Authentication Menu to start at login page.**

Use the hints! One at a time...

Okay, sau khi submit thì bài tập yêu cầu ta phải login lại website với thông tin basic:basic, phần này bạn làm cũng được, không cũng được.

**General Goal(s):**

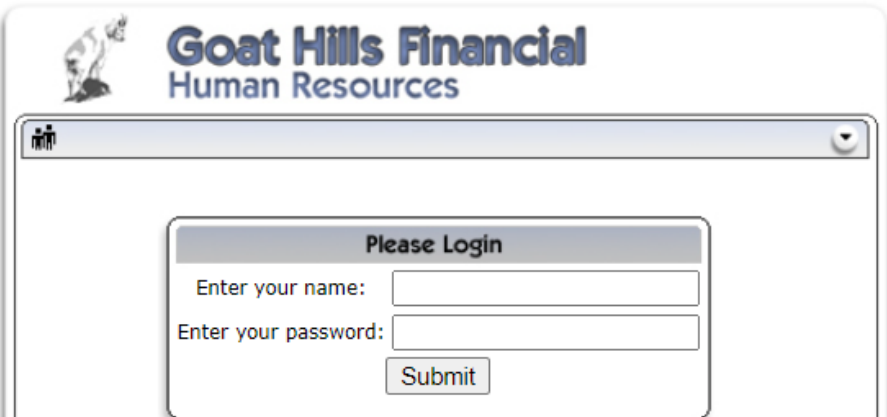
For this lesson, your goal is to understand Basic Authentication and answer the questions below.

- \* Congratulations. You have successfully completed this lesson.**
- \* Error generating org.owasp.webgoat.lessons.BasicAuthentication**

## 4. Multi Level Login 1

STAGE 1: This stage is just to show how a classic multi login works. Your goal is to do a regular login as **Jane** with password **tarzan**. You have following TANs:

Tan #1 = 15648  
Tan #2 = 92156  
Tan #3 = 4879  
Tan #4 = 9458  
Tan #5 = 4879



**Goat Hills Financial**  
Human Resources

**Please Login**

Enter your name:

Enter your password:

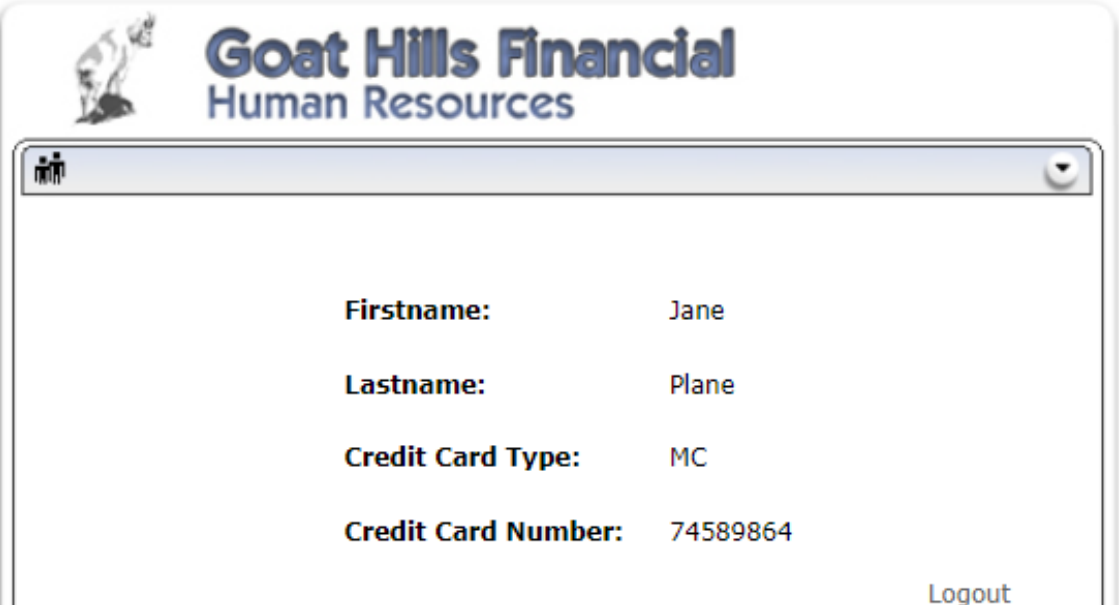
Mình không hiểu ý bài này lắm, liệu thông tin đưa ra của bài có bị nhầm lẫn không, nhưng theo kết quả tìm kiếm, Multi Login là 1 ứng dụng cho phép ta lưu trữ nhiều người dùng để tiện thay đổi qua lại trong quá trình sử dụng các ứng dụng khác. Liệu ý của bài có phải là việc đăng nhập có thể nhiều mức chứng thực khác nhau không nhỉ ?

Okay, dù sao thì yêu cầu của bài tập là hãy đăng nhập với Jane : tarzan. Khi đăng nhập, người dùng sẽ được cấp 1 mã TAN, và bài cũng cho 1 danh sách các mã TAN sẽ được cấp để mình làm, mỗi mã được sử dụng 1 lần.

Tiến hành đăng nhập với các thông tin mà bài tập đưa ra.

STAGE 2: Now you are a hacker who already has stolen some information from Jane by a phishing mail. You have the password which is tarzan and the Tan #1 which is 15648. The problem is that the first tan is already used... try to break into the system anyway.

**\* Stage 1 completed.**



The screenshot shows a web browser window with the title bar "Goat Hills Financial Human Resources". The page content displays the following information:

Firstname:	Jane
Lastname:	Plane
Credit Card Type:	MC
Credit Card Number:	74589864

At the bottom right of the page, there is a "Logout" link.

Giờ thì STAGE 2 xuất hiện, nó yêu cầu ta hãy đăng nhập bằng người dùng Jane, và giả sử ta đã lừa được cô ta bằng cách nào đó, và lấy được mã TAN cô ấy đã sử dụng "15648", hãy cố gắng đăng nhập bằng mã TAN đó. Lưu ý là mã đó đã được sử dụng rồi (mã chỉ được sử dụng 1 lần).



The screenshot shows the same web browser window as before, but with a "Please Login" dialog box in the center. The dialog box contains the following text:

Please Login

Enter TAN #2:

Submit

At the bottom right of the page, there is a "Logout" link.

Login lại thì nó hỏi mã TAN thứ 2 (vì mã thứ nhất đã xài lúc này rồi). Ở đây mình nhập lại mã TAN 1 thì nó k chấp nhận. Bắt thử request xem có gì đặc biệt, thì mình phát hiện ra, trang web nó có 1 field ẩn chỉ định số TAN mà nó sẽ chấp nhận.

```
3 Content-Length: 36
4 Cache-Control: max-age=0
5 Authorization: Basic d2ViZ29hdDp3ZWJnb2F0
6 Upgrade-Insecure-Requests: 1
7 Origin: http://192.168.48.131
8 Content-Type: application/x-www-form-urlencoded
9 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
  Safari/537.36
10 Accept:
  text/html,application/xhtml+xml,application/xml;q=0.9,
  d-exchange;v=b3;q=0.9
11 Referer: http://192.168.48.131/WebGoat/attack?Screen=
12 Accept-Encoding: gzip, deflate
13 Accept-Language: en-US,en;q=0.9
14 Cookie: JSESSIONID=C12F921B3738ACE62521B4C6F7A2BD44
15 Connection: close
16
17 hidden_tan=2&tan=15648&Submit=Submit|
```

Mình thử thay số này bằng 15648 và submit thì không được. Nhưng ngẫm lại thì, số này nó giống giá trị với cái thông tin TAN mà website yêu cầu người dùng nhập vào "Enter TAN #2". Vậy mình thử đổi nó thành 1, có thể hiểu thành kiểu "Enter TAN #1" vậy, và submit xem sao.

**\* Congratulations. You have successfully completed this lesson.**

**Goat Hills Financial**  
Human Resources



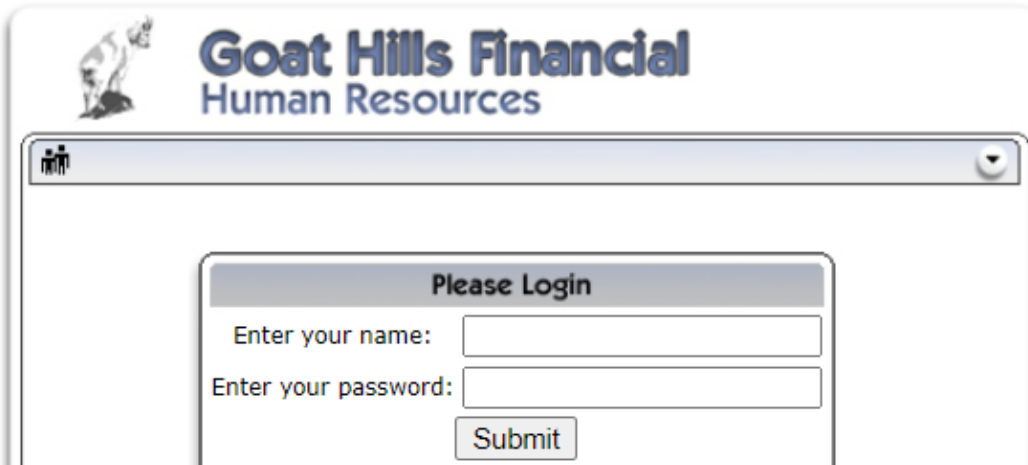
Firstname:	Jane
Lastname:	Plane
Credit Card Type:	MC
Credit Card Number:	74589864



## 5. Multi Level Login 2

You are an attacker called Joe. You have a valid account by webgoat financial. Your goal is to log in as Jane. Your username is **Joe** and your password is **banana**. This are your TANS:

Tan #1 = 15161  
Tan #2 = 4894  
Tan #3 = 18794  
Tan #4 = 1564  
Tan #5 = 45751



**Goat Hills Financial**  
Human Resources

**Please Login**

Enter your name:

Enter your password:

Giống bài trên quá @@ Vẫn yêu cầu ta đăng nhập sử dụng TAN. Nhưng điểm khác này, bài cung cấp cho ta tài khoản và mật khẩu của Joe, nhưng yêu cầu giả mạo đăng nhập thành Jane.

Nghe chả suy nghĩ ra cái gì cả, cứ bắt thử 1 gói tin trong lúc login xem sao.

```
3 Content-Length: 40
4 Cache-Control: max-age=0
5 Authorization: Basic d2ViZ29hdDp3ZWJnb2F0
6 Upgrade-Insecure-Requests: 1
7 Origin: http://192.168.48.131
8 Content-Type: application/x-www-form-urlencoded
9 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
10 Accept:
    text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
11 Referer: http://192.168.48.131/WebGoat/attack?Screen=10
12 Accept-Encoding: gzip, deflate
13 Accept-Language: en-US,en;q=0.9
14 Cookie: JSESSIONID=C12F921B3738ACE62521B4C6F7A2BD44
15 Connection: close
16
17 hidden_user=Joe&tan2=15161&Submit=Submit
```

Gotcha! Sau khi login và tới bước nhập mã TAN, nó gửi đi ngoài mã TAN, nó còn gửi thêm 1 trường dữ liệu ẩn là tên của người dùng đang đăng nhập. Rõ ràng là ta phải đổi thông tin này thành Jane để giả mạo đăng nhập thôi.

Okay, giờ thì mình mới hiểu cái bước đăng nhập đầu tiên làm gì rồi. Chúng ta dùng người dùng mà bài tập đã cho, đăng nhập để lấy mã xác minh (mã sẽ cấp đã hiển thị trước cho ta biết để sử dụng). Tiếp đó, lỗ hổng của bài là sử dụng lại tên đăng nhập của người dùng + mã chứng thực để hoàn thành việc xác thực, việc này có thể khiến ta giả mạo 1 người dùng trong hệ thống.