

# Clickjacking (UI redressing)

## What is Clickjacking?

Clickjacking, also known as a “UI redress attack” is an interface-based attack in which a user is tricked into clicking on actionable content on a hidden website by clicking on some other content in a decoy website. In simple words, an attacker uses multiple transparent or opaque layers to trick a user into clicking on a button or link on another page when they intend to click on the top-level page. Thus, the attacker is “hijacking” clicks meant for their page and routing them to another page, most likely owned by another application, domain, or both. Using a similar technique, keystrokes can also be hijacked. With a carefully crafted combination of stylesheets, iframes, and text boxes, a user can be led to believe they are typing in the password to their email or bank account, but are instead typing into an invisible frame controlled by the attacker.

## How Clickjacking Works?

Clickjacking uses an iframe to load a vulnerable website on top of an attacker's controlled domain. In the above diagram, you can see there are two web pages let us call them A and B. Website A: A website vulnerable to clickjacking that says "You won a free iPhone!" so that the victim gets lured by "Claiming the Prize!" Website B: It is an attacker-controlled website wherein you have the option to "Pay" This is the exact scenario of how Clickjacking works. Attackers create an attractive website, in our case website A, to lure the victim. Create an iframe and load it in the attacker-controlled domain, website B. In this manner, the attacker manages to fool the victim and make him pay while attracting him for a free prize! So this is how the Clickjacking attack is performed.

## Severity

Clickjacking-based vulnerabilities are one of simple bugs to find and are classified into two types:

- Clickjacking on Non-Sensitive Pages
- Clickjacking on Sensitive Pages

Clickjacking on Non-Sensitive Pages is generally considered Informational and categorized as P5 vulnerability whereas Clickjacking on Sensitive Pages is categorized as P4. Clickjacking on sensitive pages can also increase the impact of account takeover and hence can sometimes go up to the P3 category.

## Lab 1: Let's Hijack!

Observation:

As soon as we land on our lab page we see a user account form where we can fill in user details, and below it is the delete account button which if I have to guess is going to delete this user (WOW... like that's not pretty obvious! :D)

As we explore more we can see a 'Test' button, which looks as follows,

# User Profile

First Name: ABC

Last Name: XYZ

Email: admin@gmail.com

Password: ●●●●●●●●

Delete Account

Test

Solution:

This lab was no fun, to be honest, I thought we were supposed to write an HTML + CSS code to create an iframe over the delete account button such that when the user clicks on the iframe the account gets deleted.

To kill the fun we have all that done by our stupid 'Test' button. :X

Now as we can see as soon as we click on the 'Delete Account' button a pop-up arises with the following message,

# User Profile

First Name: ABC

🌐 labs.hacktify.in

Admin account deleted

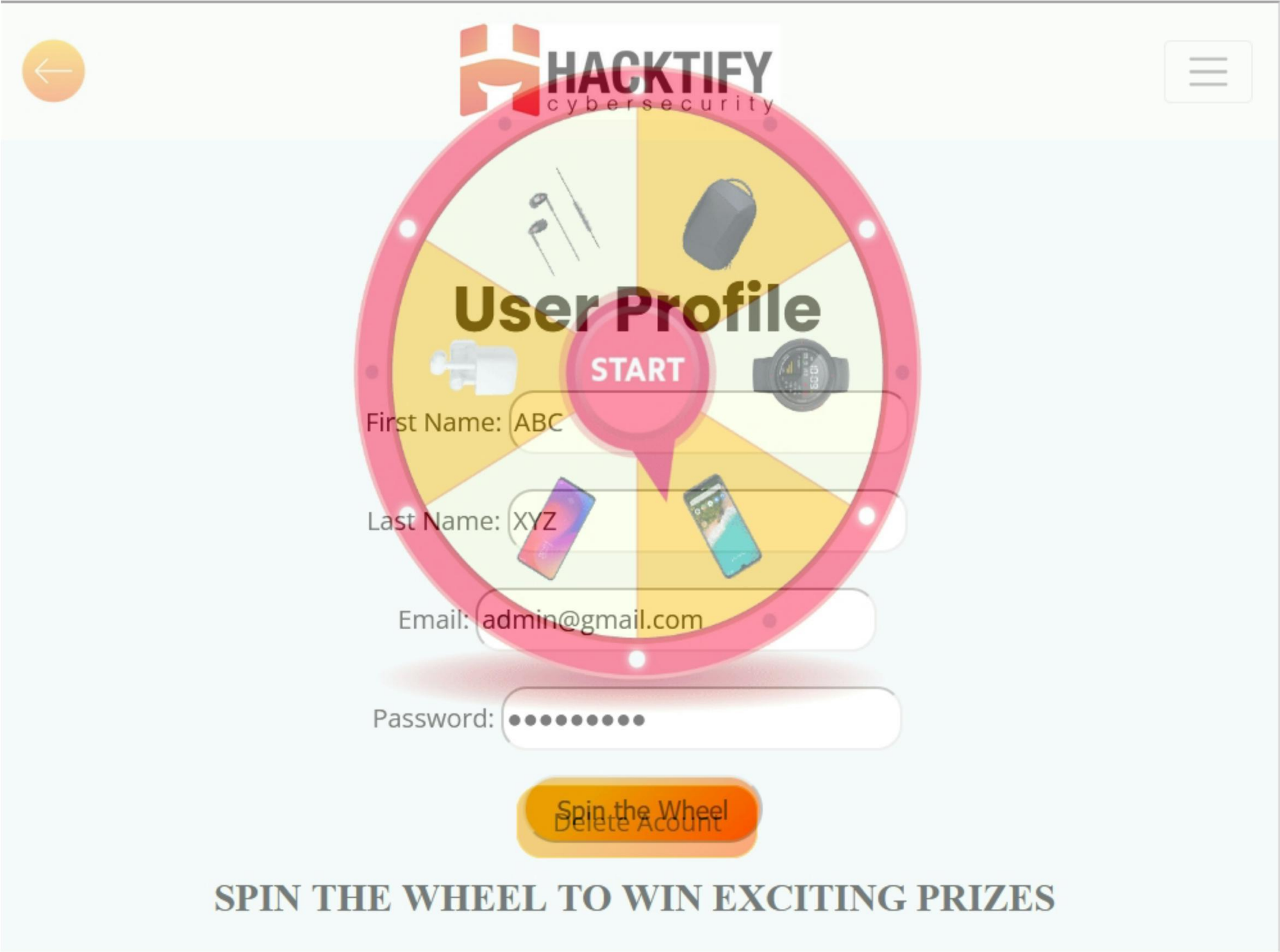
OK

Password: ●●●●●●●●

Delete Acount

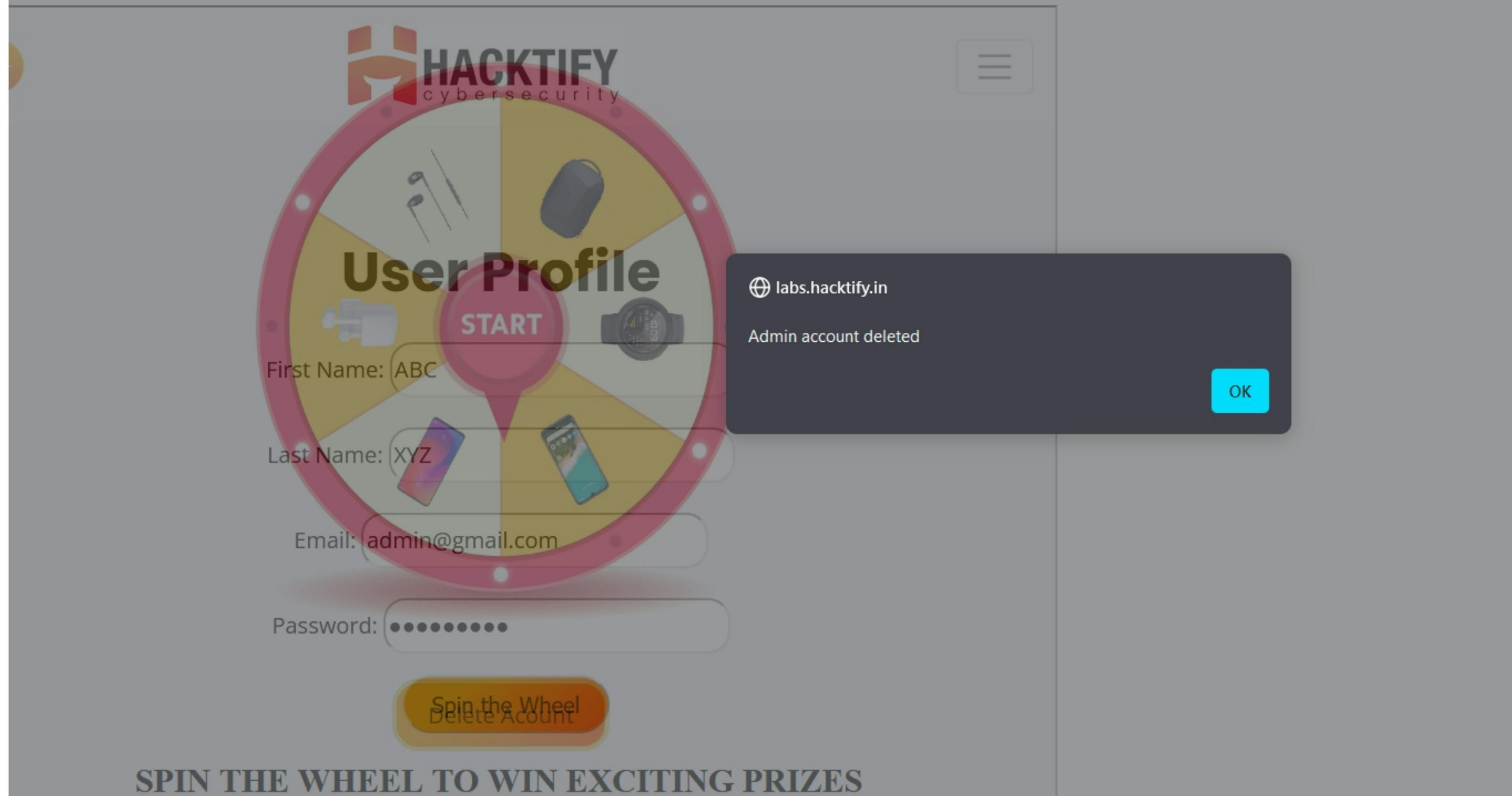
And now let's see our 'Test' button as soon as we click on the button we are redirected to a screen that looks like,

# ← Clickjacking Vulnerability



Now, on careful observation 'Spin the Wheel' button overlaps the 'Delete Account' button, and as we click the prior button we get,

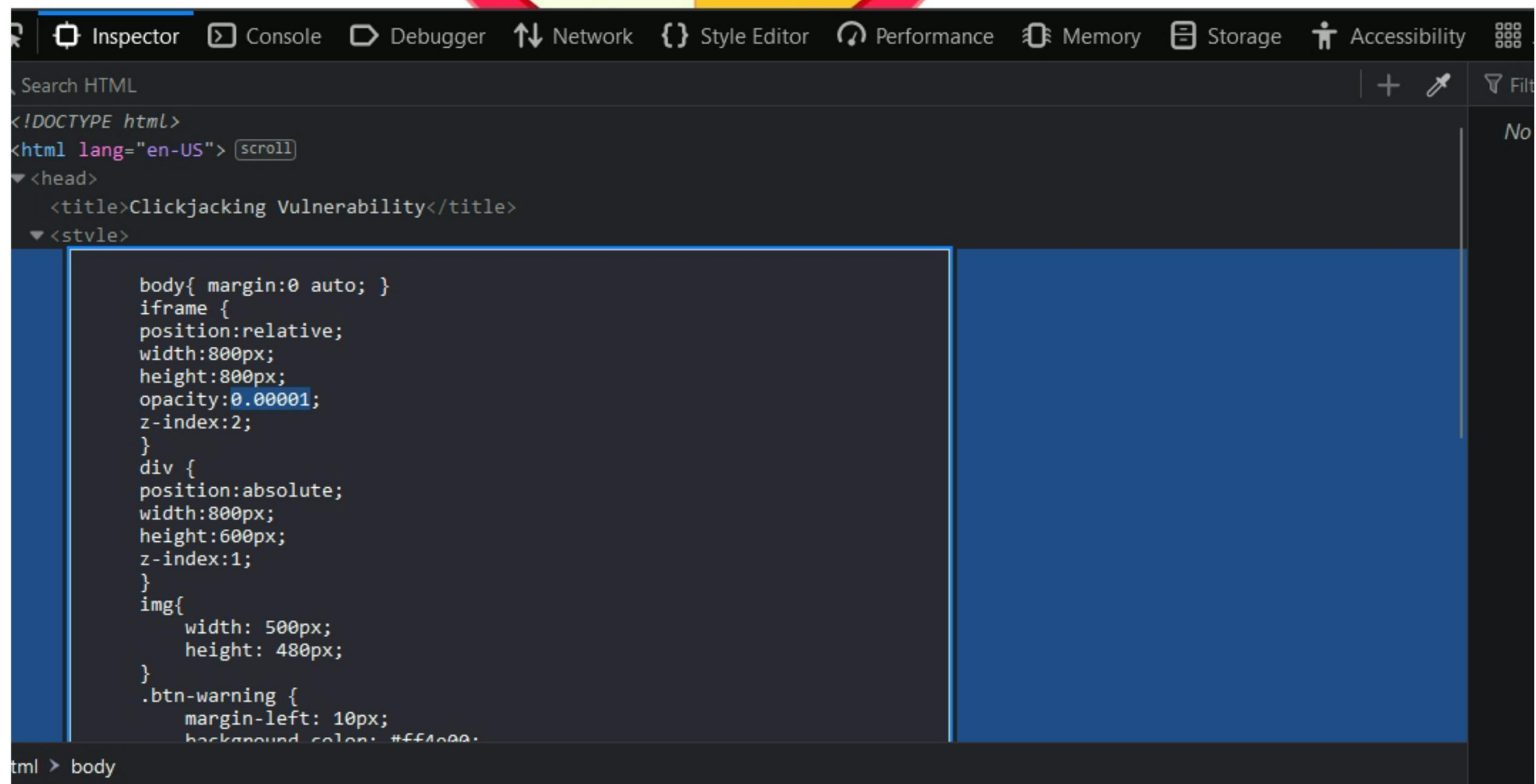
# Clickjacking vulnerability



Let's make this clickjacking more serious, and play with our buddy inspector in our browser, :)

Path: Test button > right click > inspect

Next, into the HEAD tag then into Style, and change the value of the opacity parameter for iframes from 0.5 to 0.000001,





# Clickjacking Vulnerability



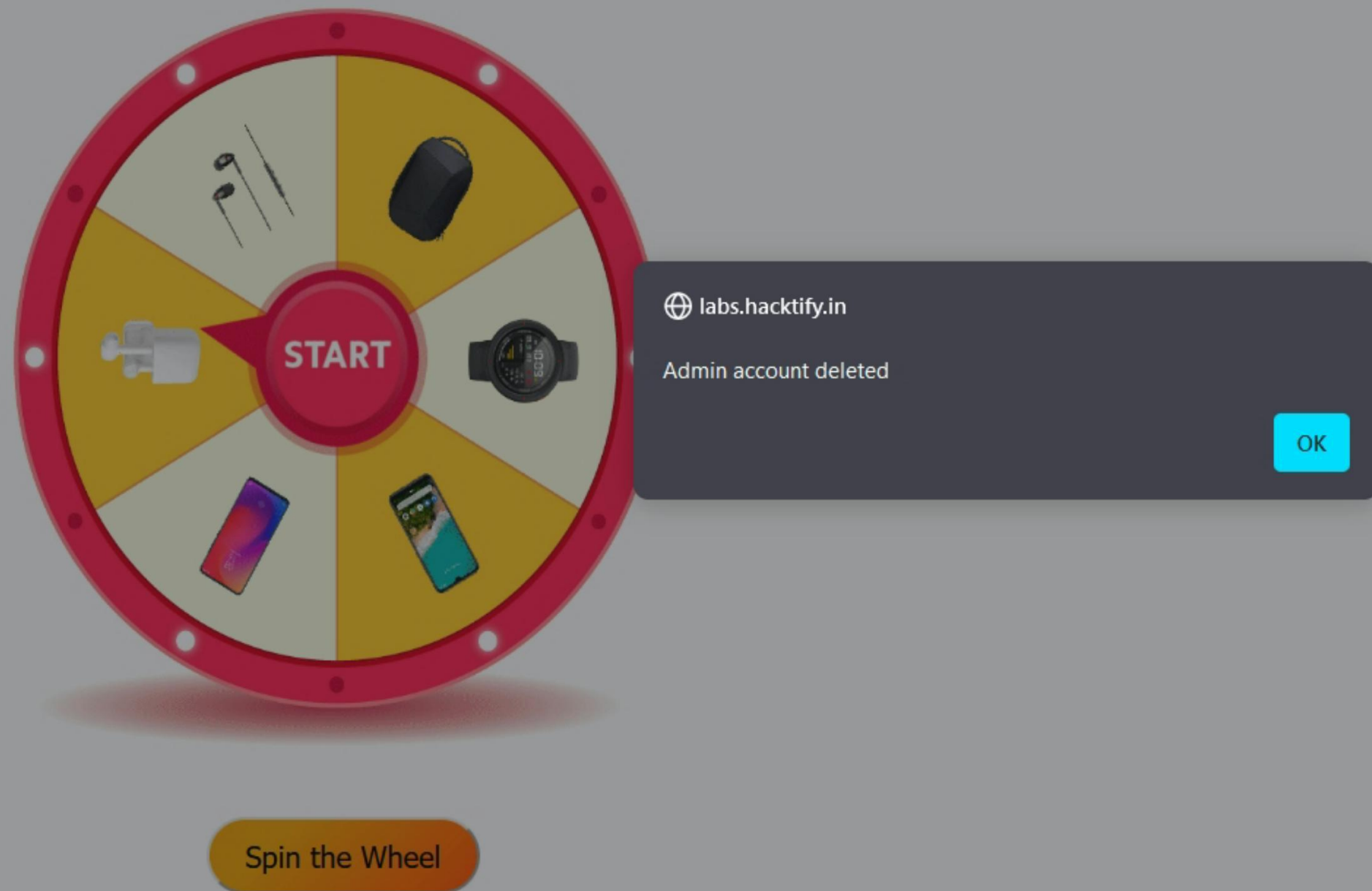
## SPIN THE WHEEL TO WIN EXCITING PRIZES

Now the page looks like,

This looks completely different page and a page that could lead you to an exciting prize by spinning the wheel, but there is nothing such as so-called a easy way for anything. We tried to win a gift and lost our admin account,



# Clickjacking Vulnerability



This concludes our website is vulnerable to UI redressing or Clickjacking attacks. :)

## Lab 2: Re-Hijack!

Observation:

As soon as we land on our lab page we see a user login form where we can fill in user details as provided(admin@gmail.com:admin@123), As we explore more we can see a 'Test' button, which looks as follows,



# Google Log In

Email

Password

Login

Use email as admin@gmail.com and password as  
admin@123

Test

Solution:

I hate these labs with no actual practical implementations!!

Let's see what the actual site throws at us as soon as we log in using the provided credentials, i.e, admin@gmail.com:admin@123,

# Google Log In

Email admin@gmail.com

🌐 labs.hacktify.in

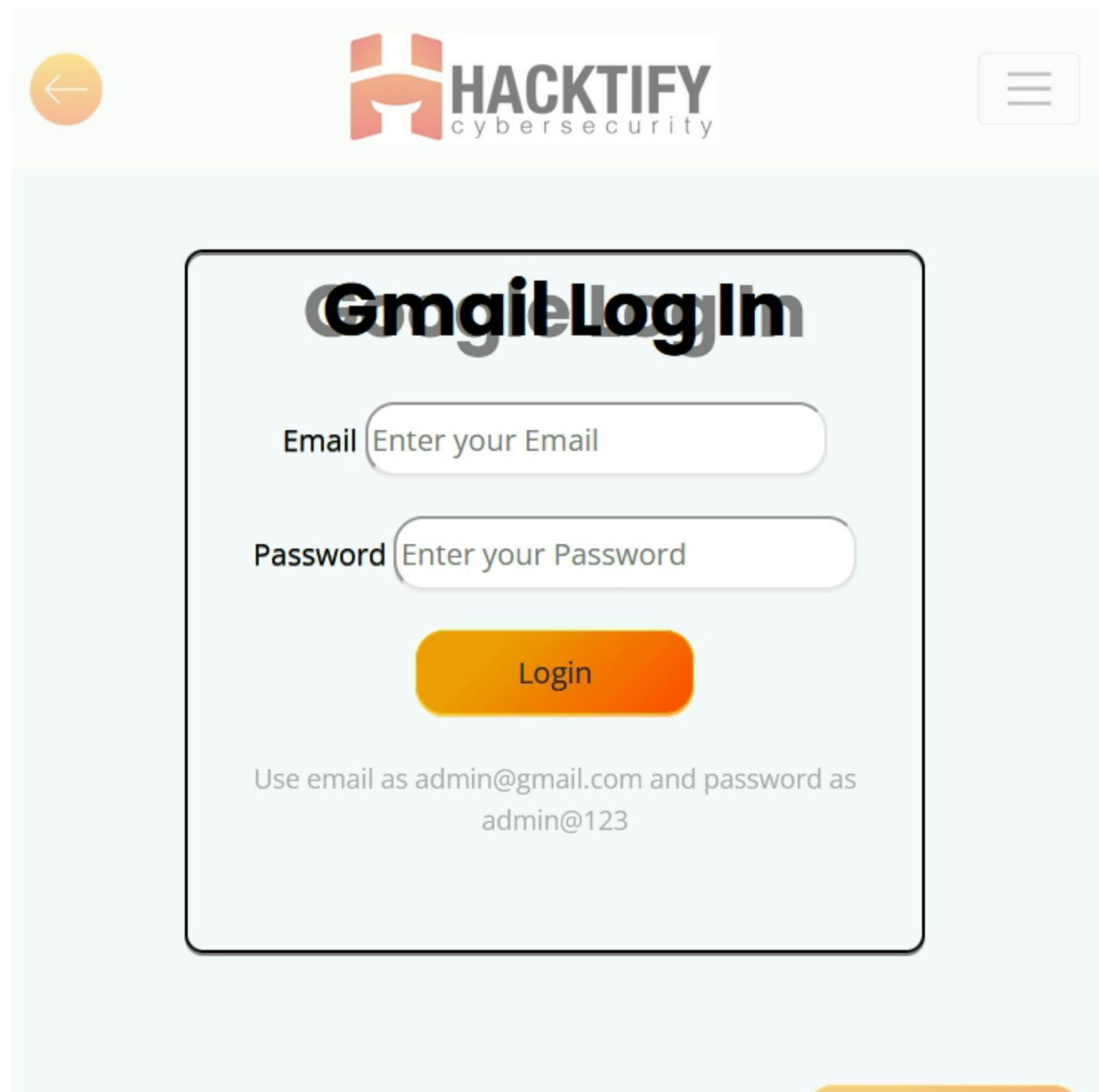
Login Successfull

OK

Use email as admin@gmail.com and password as  
admin@123

Now let's see what the 'Test' button has for us, on clicking the 'Test' button we are redirected to the following page,

## ← Clickjacking Vulnerability

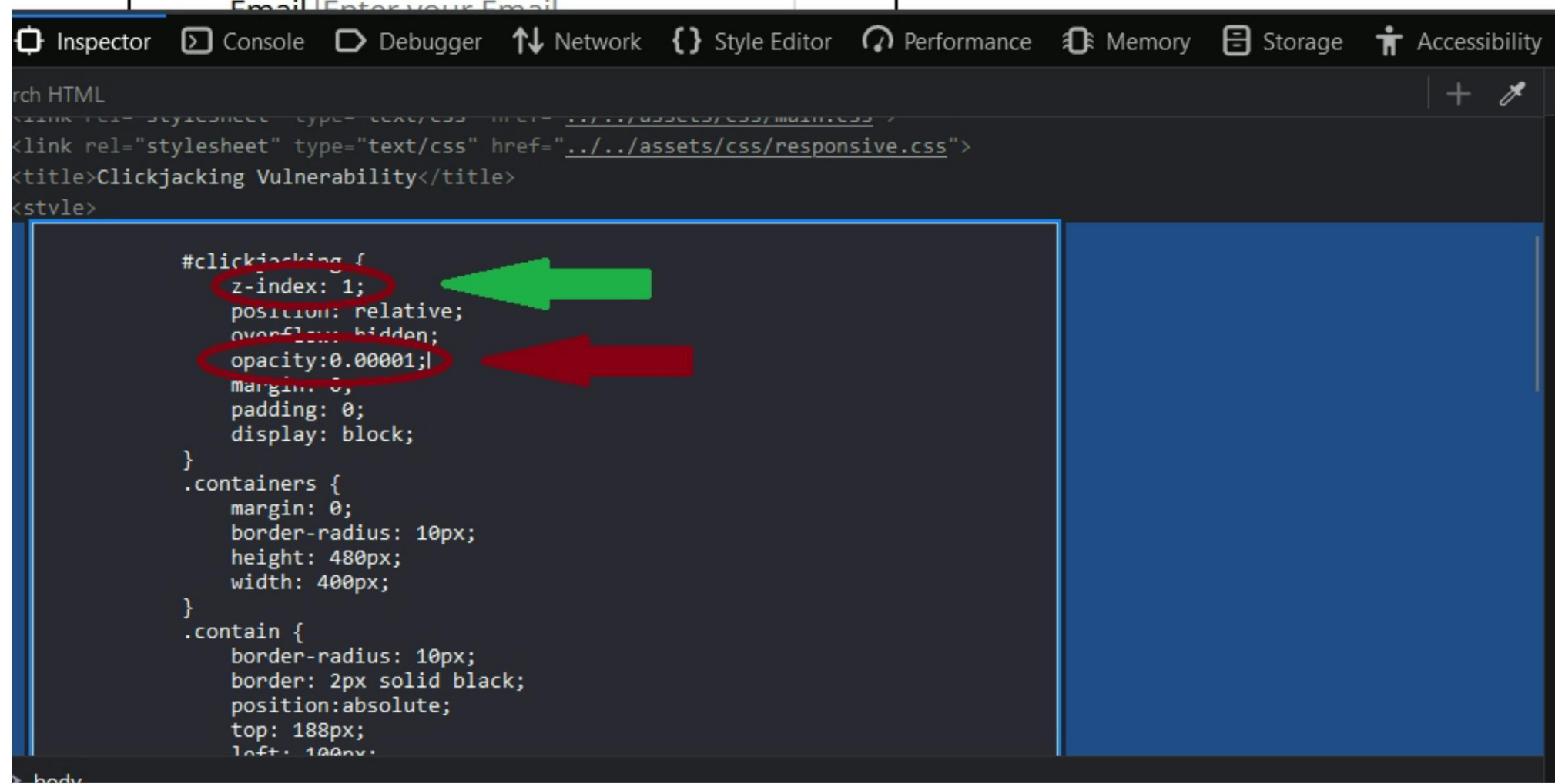


The screenshot displays a web application interface. At the top, there is a navigation bar with a back arrow icon on the left, the 'HACKTIFY cybersecurity' logo in the center, and a hamburger menu icon on the right. The main content area has a light blue background. In the center, there is a white rounded rectangle with a black border, which serves as a login form. The form is titled 'Gmail Login' in a large, bold, black font. Below the title, there are two input fields: 'Email' with the placeholder text 'Enter your Email' and 'Password' with the placeholder text 'Enter your Password'. Below these fields is an orange 'Login' button. At the bottom of the form, there is a line of text: 'Use email as admin@gmail.com and password as admin@123'. The form is overlaid on a background that appears to be a blurred version of the Hacktify Cybersecurity website.

Okay, we can observe that there is an overlapping form which we can make more visible using the method we used in Lab 1. And when checking the page source we can notice a very crucial thing.

# Gmail Log In

Email:



Now as we enter the credentials here we get the following response,

# Gmail Login

Email admin@gmail.com

Password ●●●●●●●●

Login

Use email as admin@gmail.com and password as  
admin@123

🌐 labs.hacktify.in

Secret Credentials:  
Email: admin@gmail.com  
Password: admin@123

OK

If we revisit both the login forms there was a slight difference in both when we entered the credentials in the actual login form we got the response as "login successful" whereas the other one gave us our exact username password, An attacker can use this in a similar way for phishing and gain password and other sensitive information about their target by giving them similar looking but fake web pages, and no matter how cautious we may be attacker thinks one step further. It is important for us to realize that the severity mentioned above is judged as per the bounty and impact P.O.V. So that does not make the attack less or more important but equally as important.

Important Read: FluHorse- Android Malware (Most of it won't make a lot of sense but just give it a read and you'll understand that what we saw is just a very tiny picture of what UI redressing or UI can impact, "Severity/impact of an attack increases several times if we chain them with others!")

Happy Hacking.....:)  
IMRAN.