RESET PASSWORD VULNERABILITIES

~ PREM KUMAR VERMA

1. Password reset link not expiring

When a user requests changing password then he gets a password reset link to reset the password, that's the normal behaviour but it also should expire after some period of time. If it is not expiring and you can use the password reset link multiple times to reset the password. Then you can consider it as vulnerability

FIRST SCENARIO:

1.) Go to https://site.com/users/forgotten_password and send the password reset link to your email.
2) Go to your email inbox you see reset token like this https://site.com/users/new_password?reset_token=your-reset-token and click the link to change password. you can use this link many times to reset your password.

Impact:
Password Reset Link not expiring after changing password
https://hackerone.com/reports/898841

SECOND SCENARIO:

1) Go to https://infogram.com/forgot and ask for the password reset link.
2) Don't use the link, keep it in the Email inbox.
3) After some time, repeat step 1.
4) This time use the password reset link which was asked in step 3. means the 2nd link.
5) After changing the password, use the password reset link that was captured in step 1.
6) You'll see the password reset link is not expired even after password change.

Impact:
Password Reset Link not expiring after changing password
https://hackerone.com/reports/283550

THIRD SCENARIO:

1) Create a account or use existing one.
2) Confirm Your email address.
3) Now log out from your account and request for password reset code for your account .
4) Don't use the code that has been sent to your email address.
5) In new browser log in back to your account.
6) Go to account setting and change your password .
7) Now go to email and check the password reset code that we requested in step 3.
8) Change Your password using that reset password code .
9) You can see that your password has been changed The reset code is not expired after changing the password

Impact:
If the site has a token issue, The result is the reset password token in the Step 3 is still usable and did not expire yet. If the victims opens his mail in cybercafe or in attackers device and forgot to log out then attacker can access that system and can reset the password of his account.
https://hackerone.com/reports/948345

Fourth scenario:

1) Send the password reset link to your email.
2) Don`t open the password link just copy it and paste into any editor.
3) Open your account.

4) Go to your account settings.
5) Under account, you will see Account Overview.
6) Go to the Email and password Option and change the email and verify it.
7) After changing the email go to your password reset link which you copied.
8) Change your password.
9) BooM password Change
.
Impact: The attacker can still change the password if victim thinks his/her account is compromised and decided to change his/her email.
https://hackerone.com/reports/685007

Fifth scenario:
1) Attacker visits https://card.starbucks.com.sg/forgetPassword.php and enters his account's email
2) The link is sent to the attacks email's inbox and he clicks on the link while having a proxy monitor the request(burp)
3) The attacker then modifies the email to put the victim's email in these 2 requests as shown in the image below 1.PNG (F263235) & 2.PNG (F263234)
4) Upon submitting the request, the password will be changed and the victim's password will be changed to the desired password

Impact:
This attack does not require the victim to perform any actions and yet the account can be taken over by the attacker and this leaks the victim's personal information and starbucks card which can be used to purchase items. The attacker can also capture the session id.
https://hackerone.com/reports/315879

SIXTH SCENARIO:

- *Not using the phone number or email of the user corresponding to the session.(take the phone number/email from the request)*
  1) Web app is sending a verification code to email before changing some sensitive fields.
  2) Intercepted the request in burp and found the email parameter (eg: email: victim@gmail.com) in the POST request.
  3) Changed the email to another gmail (eg: attacker@gmail.com) 4) Boom! Got the verification code at attacker@gmail.com

- *Password reset token leak via referer*
  1) Go To https://ucp.nordvpn.com/lost-password Page
  2) Enter Your Email And Click On Reset Password
  3) Go To Email & Click on Password Reset Link
  4) On Password Reset Page Click On Social Media Links Given Below And Capture The Request Using Burp Suite
  5) Check if the referer header is a leaking password reset token?

GET /nordvpn/ HTTP/1.1

Host: www.facebook.com

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:71.0) Gecko/20100101 Firefox/71.0

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,/;q=0.8

Accept-Language: en-US,en;q=0.5 Accept-Encoding: gzip, deflate Connection: close

Referer: https://ucp.nordvpn.com/reset-password/ 7ac4b7c4654797e8f2a061676314b0959e7de179d33a73dfaa0152b1ec617f46/ Cookie: fr=1oATYBf0BJ0DXndaC.AWV1Ag1KKbwlzMPBUI8vUz2NaP0.BduttV.ZM.F3B.0.0.BdwUoa.AWWnaTR3; sb=Vdu6XRBcBVULAbruWmZbgfPF; datr=Vdu6Xar5xVM18qIAfGd61Mh3; c_user=100003539320116; xs=4%3AcaHYJilvx4Lkqg%3A2%3A1572948495%3A6163%3A4671 Upgrade-Insecure-Requests: 1

Impact:
For Example User A Forgots His Password He Got To Forgot Password Page Reset And Receive Link By Email And Opened The Password Forgot Page Then Eventually User Remembers His Password And Though To Use Some Facebook he Opened Nordvpn facebook page or click on social media page then the link will be leaked in refer header. It allows the person who has control of particular site to change the user's password (CSRF attack), because this person knows reset password token of the user. Other Impact -Account Takeover Possible The Person Operating NordVpn Social Media Page Can Also Exploit like if they have enabled page analytics then they may see from where users are referring onto there page and from there they see that password reset link and can reset the password for victim
https://hackerone.com/reports/751581

- *Password reset token leak via response*
  1) send the password reset link request
  2) intercept response
  3) see the reset-password link was returned in response

request:
POST /access/forgotPassword HTTP/1.1
Host: api.xprogram.com
*User-Agent: Mozilla Accept: application/json, text/plain, */**
Accept-Language: en-US,en;q=0.5
Content-Type: application/json;charset=utf-8
Referer: https://app.xprogram.com/account/forgot-password
Content-Length: 52
origin: https://app.xprogram.com
Cookie: redacted=yes;
Connection: close {"email":"foobar@gmail.com"}

response:
HTTP/1.1 200
OK{“name”:”send-email”,”resetPasswordLink”:”https://app.xprogram.com/account/reset-password?token=xxxxxxxxxzzzzzzzzyyyyyyyy" ,
“many-other-parameters”:”many values”}

https://medium.com/@yassergersy/account-take-over-via-reset-password-f2e9d887bce1

- *Password reset with manipulating email parameter.Password reset token leak via response*
  while requesting a password reset link for the victim user, we can try the below parameter manipulation to get a copy of the reset link of the victim on the attacker email.

  Double parameter (aka. HPP / HTTP parameter pollution):
  email=victim@xyz.tld&email=hacker@xyz.tld

  Carbon copy
  email=victim@xyz.tld%0a%0dcc:hacker@xyz.tld

  Using separators:
  email=victim@xyz.tld,hacker@xyz.tld
  email=victim@xyz.tld%20hacker@xyz.tld
  email=victim@xyz.tld|hacker@xyz.tld

  No domain:
  email=victim

  No TLD (Top Level Domain):
  email=victim@xyz

  XML:
  try XXE!
  JSON table: {
  "email":["victim@xyz.tld", "hacker@xyz.tld"]
  }

  ***use content type converter burp ext*

  Capture

  https://hackerone.com/reports/1175081

- *.Host Header Poisoning.*
  A common way to implement password reset functionality is to generate a secret token and send an email with a link containing the token. If an attacker is able to change the host header they can then redirect the token to their website or server which can lead to password reset poisoning

  1) intercept the request and change the Host header to attacker.com.
  2) Now check your mail if you have received the password reset link and contains attacker.com in the url.
  If it does then its vulnerable to password reset poisoning

Changing the host directly to any website doesn't work most of the time. You can try to bypass this with below methods.

Add X-Forwarded-Host header : *Host: attacker.com* *X-Forwarded-Host: target.com* *or :* *Host: bing.com* *X-Forwarded-Host: target.com* *or :* *Host: target.com* *Host: attacker.com* *You can use ngrok server as your attacker server*

Capture

https://shahjerry33.medium.com/otp-bypass-developers-check-5786885d55c6

https://hackerone.com/reports/226659

- *.No rate limiting on password reset*

  Rate limiting is used to control the amount of incoming and outgoing traffic to or from a network. Basically, no rate limit means there is no mechanism to protect against requests you made in a short frame of time. So try to send lots of requests, if it is not blocking you then you can consider it as vulnerability

  1) Start the burp suite and intercept the password reset request
  2) Send to intruder
  3) Use null payload

  https://hackerone.com/reports/838572

- *User enumeration via Password reset page*

  The username enumeration is an activity in which an attacker tries to retrieve valid usernames from a web application. You can check this type of bugs on login pages, registration form pages or password reset pages.

  1) Go to the password reset page
  2) Enter a username that exists, there would be no error, and it will be redirected to the login page
  3) Enter a username that doesn't exist, there would be an error saying something like 'user account doesn't exist' etc.

  https://hackerone.com/reports/77067

- *HTML injection in password reset page*

  HTML Injection which is also referred in Content spoofing, also referred to as content injection, or "arbitrary text injection" or virtual defacement

  The steps were as follow:

Open the Create New Account Page of the application, enter your email id and Password.
In the First Name parameter, HTML Injection payload

(<h1>Please click here to login to your account<h1> is inserted
https://hackerone.com/reports/111094