#BugBounty — “Let me reset your password and login into your account “-How I was able to Compromise any User Account via Reset Password Functionality

Avinash Jain (@logicbomb_1) Follow

Mar 14, 2018 · 3 min read

Hi Guys,

One more interesting blog explaining an interesting vulnerability that I founded a while back in one of the Mobile Wallet Companies of India.

> *To login into any online website , we need to have an username which can be user’s registered mail id and password that he has set for it and if he doesn’t remember his password, there is a Reset Password Feature which comes to help.*

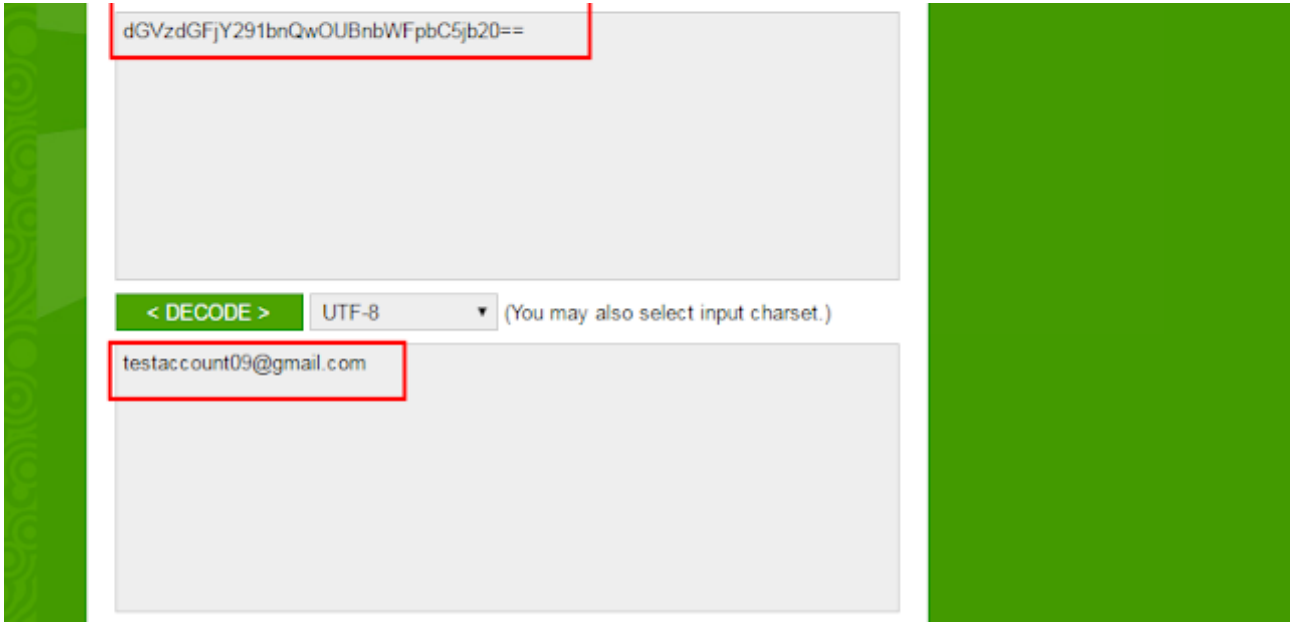
> *While researching out for the vulnerability around this feature , I found a logical flaw by which I was able to reset any user password and login with the same to takeover any user's account.*

Let's now enter into the explanation-

1. When I clicked on Reset password functionality for the account "testaccount09@gmail.com", I received a mail saying "To reset the password , please click on the below link-" and the link was something—

http://www._____.com/account/resetpassword?id=296417&token=dGVzdGFjY291bnQwOUBnbWFpbC5jb20=&vit=MjAxNi8xMC8yNQ==

2. Here 'id' is the identification number associated with the user account and 'token' is the base64 decoded registered mail ID of the user which here is "testaccount09@gmail.com" and 'vit' is the base64 decoded time stamp whose value in this case is "2016/10/25"



3. Researching more, I have found that the timestamp parameter is the expiry date of the reset password link which here was 2 days ahead from the time user clicked on the reset password option.

4. Here comes the step of compromising user account.By user enumeration on the same page, I found one valid user account, generated forgot password link for it and now begins the task for finding the right reset password link, I replaced the mail id of the user and encoded it to base64 and kept the timestamp value to 2 days ahead of the current date.

Victim mail id—varun09811@gmail.com

Base64 encoded value (Parameter = token)—
dmFydW4wOTgxMUBnbWFpbC5jb20=

Timestamp value (Parameter = vit)—MjAxNi8xMC8yNQ=

5. Another part comes here is to find “id” associated with that particular user mail id. Since it’s a 6 digit code so I tried brute forcing it (fortunately no rate limiting was set) and after a while, I found the right id associated with the victim mail id which happens to be id=254346 .(yes, this is something time consuming).

6. So the tampered URL looks like -

http://www. .com/account/resetpassword/?id=254346&token=dmFydW4wOTgxMUBnbWFpbC5jb20=&vit=MjAxNi8xMC8yNQ=

I loaded the link in the browser, and I was presented with “Set new password” ,

New Password Set Page

I reset his password and was successfully able to login into his account. I had the complete access to his account, can use his wallet money , change registered mobile number and everything!

. . .

I reported this vulnerability to the concerned enterprise, and they were quick to patch it within 2 days. I thank the company for the small token of appreciation :)

Thanks for reading!

~Logicbomb (https://twitter.com/logicbomb_1)

Security | Ethical Hacking | Hacking | Bug Bounty | Penetration Testing

627 claps 8

Avinash Jain (@logicbomb_1)

Lead Infrastructure Security Engineer @groferseng | DevSecops | Part time BugBounty Hunter | Acknowledged by Google, NASA, Yahoo, United Nations, BBC etc.

Follow

Writing a Password Protected Bind

0x0FFB347
Mar 8 · 5 min read
246

Ping Power—ICMP Tunnel

Nir Chako
Dec 17, 2018 · 8 min read
488

How to Make a Captive Portal of

Trevor Phillips
Dec 18, 2018 · 6 min read
280

Responses

Write a response...

Conversation with Avinash Jain (@logicbomb_1).

Ak1T4
Mar 15, 2018

great write up! what was the fix?

1 1 response

Avinash Jain (@logicbomb_1)
Mar 15, 2018

Thanks Ak1T4. :)
The mitigation for this vulnerability is to have some strong encryption rather than weak base64 encoding. Send a encrypted token bind it with the particular user and that's it.

19

Applause from Avinash Jain (@logicbomb_1) (author)

Tamás Tóth
Apr 11, 2018

Take-away: use real encryption, and avoid enumerable IDs in front-end as plague. Nice write-up.

1

Conversation with Avinash Jain (@logicbomb_1).

Nikhil Dhyani
Mar 18, 2018

Thanks for the writeup! Did you determine token is base64 encoded because of "==”?

1 response

Avinash Jain (@logicbomb_1)
Mar 18, 2018

Thanks Nikhil. Yes by simple look, it is clear that it is base64 encoded

2

Conversation with Avinash Jain (@logicbomb_1).

Utkarsh Agrawal
Mar 21, 2018

Hi Avinash, Nice write-up, but my opinion for mitigate this vulnerability is to use a particular access-token, emails id's encryption is not the good way to mitigate it? What's your opinion?

1

1 response

Avinash Jain (@logicbomb_1)
Mar 21, 2018

Yes Utkarsh, you are absolutely right.

Show all responses