



CYBER THREAT DEFENSE

Web Application Penetration Test Case Study



SpaceCube
ARCHITECTS



***Note: No real company details have been used for this case study**

Introduction

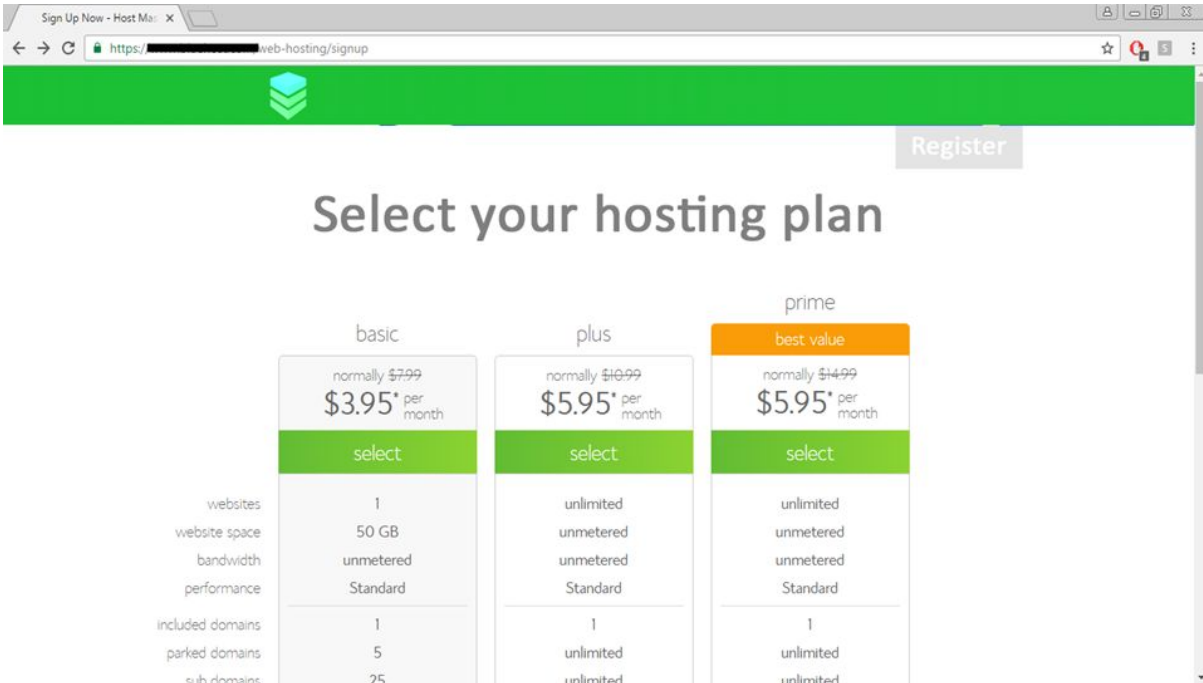
A client that had a SaaS developed to manage server hosting systems, requested an external penetration test against the application, a report with all the vulnerabilities discovered and the remediation solutions.

The client wanted to launch the new application in a month after the penetration test was done.

The security team discovered that the client could have launched the product with a critical vulnerability, that may have allowed any hacker to take complete control over the admin account.

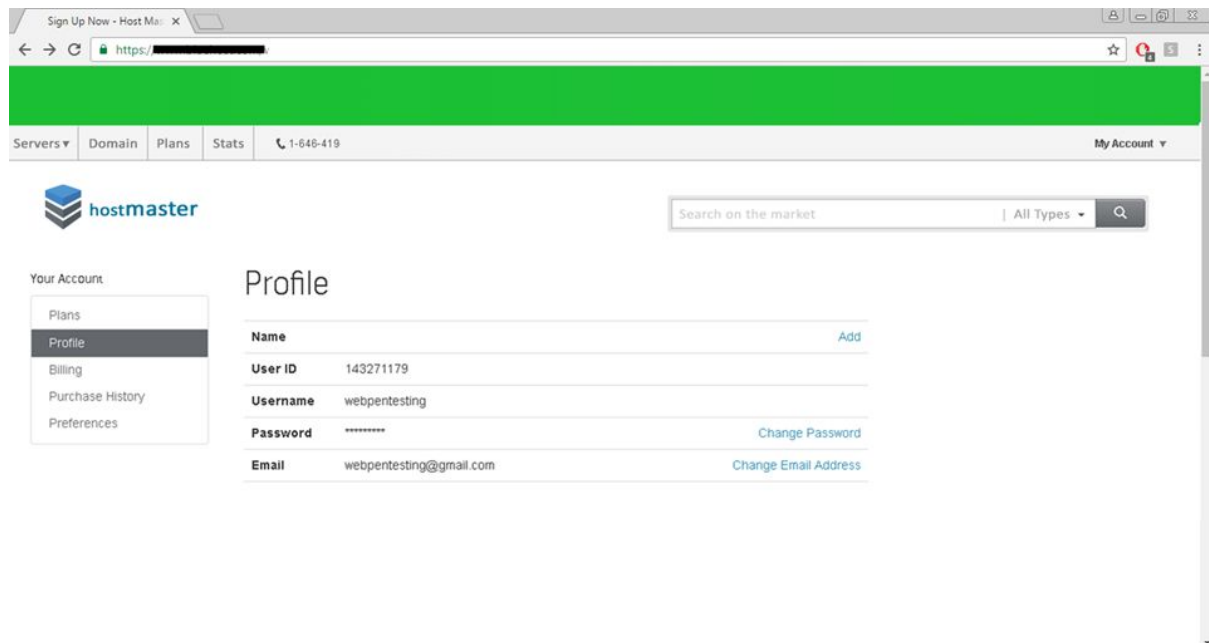
Server hosting management application

The application was built for hosting companies to manage their servers and resources and it allowed users to register accounts and buy server hosting solutions.



The screenshot shows a web browser window with the URL <https://web-hosting/signup>. The page has a green header with a logo and a 'Register' button. The main heading is 'Select your hosting plan'. Below it, there are three columns representing different hosting plans: 'basic', 'plus', and 'prime'. The 'prime' plan is highlighted with an orange header and labeled 'best value'. Each plan shows a 'select' button and a list of features and prices.

	basic	plus	prime best value
	normally \$7.99 \$3.95* per month	normally \$10.99 \$5.95* per month	normally \$14.99 \$5.95* per month
	select	select	select
websites	1	unlimited	unlimited
website space	50 GB	unmetered	unmetered
bandwidth	unmetered	unmetered	unmetered
performance	Standard	Standard	Standard
included domains	1	1	1
parked domains	5	unlimited	unlimited
sub domains	25	unlimited	unlimited



Classic cyber attacks had no impact

The security team attempted to penetrate the application using attacks specific to web applications, such as: SQL injection, XSS, XXE, Missing Authorizations, user brute-force, input validations and so on but they were unsuccessful due to the fact that the application was built with security in mind since the beginning.

This happens very rarely so congrats to the development team!

Programming error

Giving the fact that the application had defenses against most common vulnerabilities, the security team started to investigate for programming and business logic flaws.

On My Account page, each user had a number assigned to their profile. Usually this number is the number that increments inside the database when a new user is registered.

Name		Add
User ID	143271179	
Username	webpentesting	
Password	*****	Change Password
Email	webpentesting@gmail.com	Change Email Address

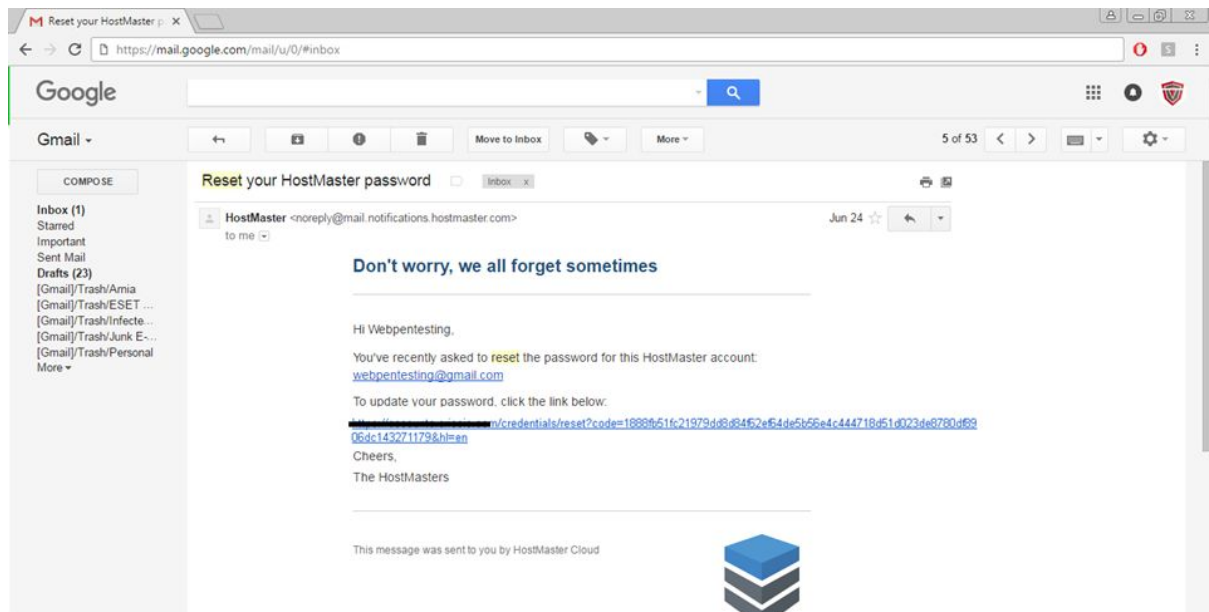
The team was looking for programming errors in **Forgot Password** section and requested a password reset mail.

Forgot your password?

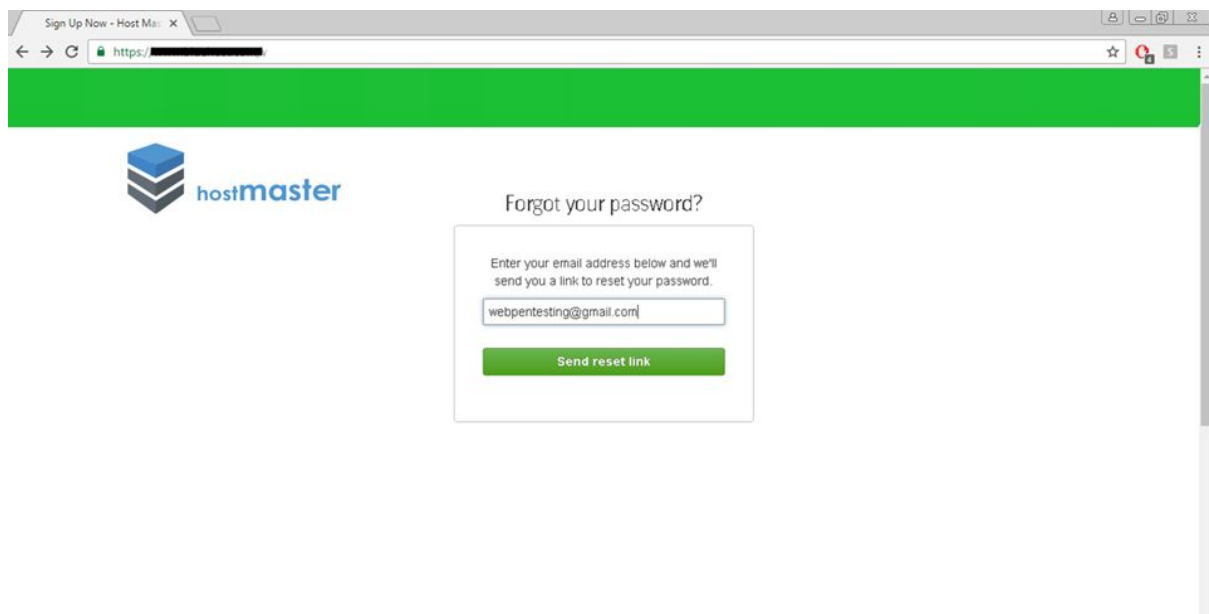
Enter your email address below and we'll send you a link to reset your password.

[Send reset link](#)

It received the email with the link.



In order to compare multiple reset links, the team requested multiple Forgot Password mails on the same address.

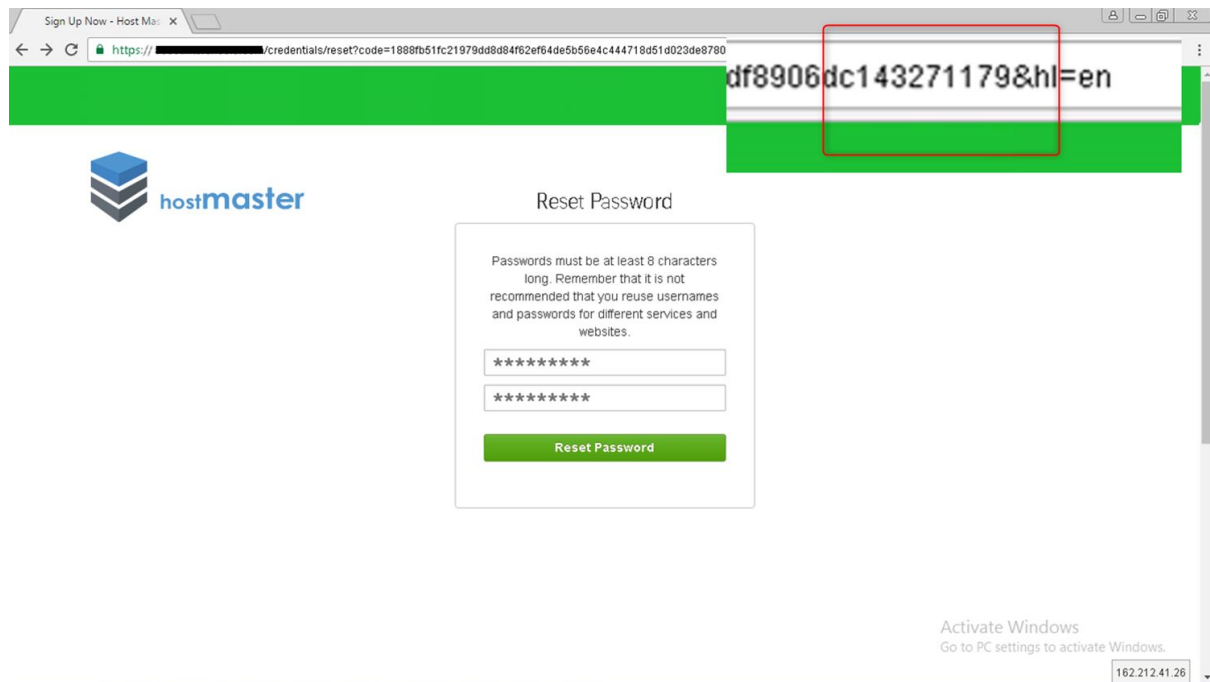


Looking at the reset password links the team observed a small coincidence. All reset links had a portion of the code identical and it was discovered that it was actually the ID of the user that requests the reset.

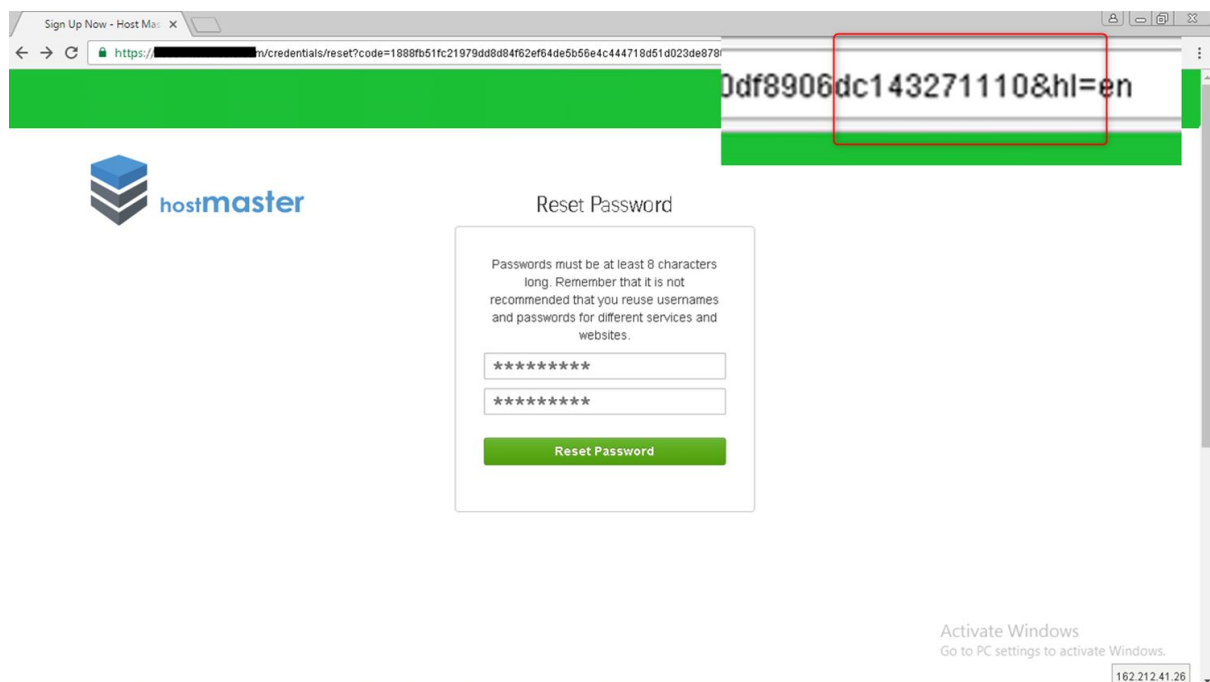
```
1
2
3
4
5
6
7
8
9
10
11 code=1888fb51fc21979dd8d84f62ef64de5b56e4c444718d51d023de8780df8906dc143271179&hl=en
12 code=316a963129374de9d8cc108c0b4ea464316a963129374de9d8cc108c0b4ea464143271179&hl=en
13 code=144d9dbb60b6c2c24922cd62d249412f944d9dbb60b6c2c24922cd62d249412f143271179&hl=en
14 code=94b9796a888b4cd1a2c1322d73999f5294b9796a888b4cd1a2c1322d73999f52143271179&hl=en
15 code=3f7103488be6cfd1328ee12e40cdda63f7103488be6cfd1328ee12e40cdda6143271179&hl=en
16 code=557faf8e54ec20deace42de8868e0e4e557faf8e54ec20deace42de8868e0e4e143271179&hl=en
17 code=32bbf0c829ec5b7f780cd5231abbf91632bbf0c829ec5b7f780cd5231abbf916143271179&hl=en
18 code=2c49353669a4b803a493b0677f8045ca2c49353669a4b803a493b0677f8045ca143271179&hl=en
19
```

```
1
2
3
4
5
6
7
8
9
10
11 code=1888fb51fc21979dd8d84f62ef64de5b56e4c444718d51d023de8780df8906dc143271179&hl=en
12 code=316a963129374de9d8cc108c0b4ea464316a963129374de9d8cc108c0b4ea464143271179&hl=en
13 code=144d9dbb60b6c2c24922cd62d249412f944d9dbb60b6c2c24922cd62d249412f143271179&hl=en
14 code=94b9796a888b4cd1a2c1322d73999f5294b9796a888b4cd1a2c1322d73999f52143271179&hl=en
15 code=3f7103488be6cfd1328ee12e40cdda63f7103488be6cfd1328ee12e40cdda6143271179&hl=en
16 code=557faf8e54ec20deace42de8868e0e4e557faf8e54ec20deace42de8868e0e4e143271179&hl=en
17 code=32bbf0c829ec5b7f780cd5231abbf91632bbf0c829ec5b7f780cd5231abbf916143271179&hl=en
18 code=2c49353669a4b803a493b0677f8045ca2c49353669a4b803a493b0677f8045ca143271179&hl=en
19
```

The next step was to access the link in the URL and change the ID of the user with other user's IDs, by incrementing its value.



Other user ID.



Once the request was sent the application redirected to the Account page of each user where the password was reset.

The screenshot shows a web browser window with the URL <https://...>. The page has a green header bar. Below the header, there is a navigation bar with links: Servers, Domain, Plans, Stats, and a phone number 1-646-419. On the right of the navigation bar is a link to My Account. The main content area features the Hostmaster logo on the left and a search bar on the right. A sidebar on the left lists 'Your Account' options: Plans, Profile (selected), Billing, Purchase History, and Preferences. The main section is titled 'Profile' and contains a table with the following information:

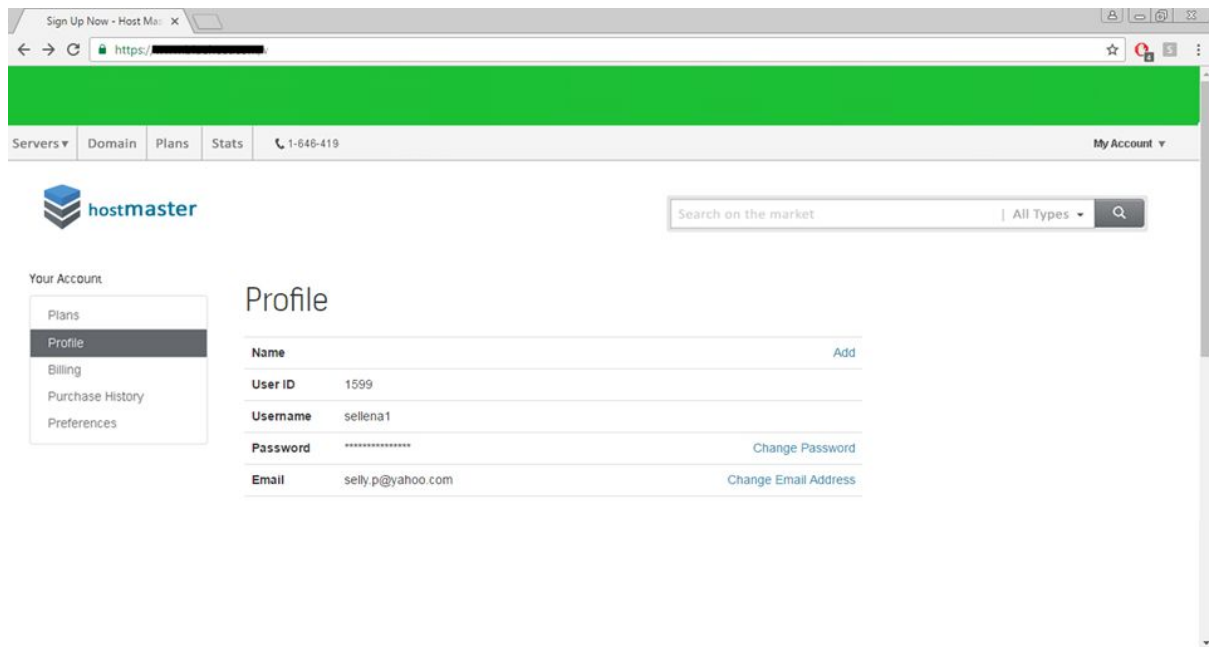
Name		Add
User ID	143271110	
Username	craigparker	
Password	*****	Change Password
Email	craigparker@hotmail.com	Change Email Address

And other

The screenshot shows a web browser window with the URL <https://...>. The page has a green header bar. Below the header, there is a navigation bar with links: Servers, Domain, Plans, Stats, and a phone number 1-646-419. On the right of the navigation bar is a link to My Account. The main content area features the Hostmaster logo on the left and a search bar on the right. A sidebar on the left lists 'Your Account' options: Plans, Profile (selected), Billing, Purchase History, and Preferences. The main section is titled 'Profile' and contains a table with the following information:

Name		Add
User ID	166003180	
Username	nathanpal7957	
Password	*****	Change Password
Email	nathanpal79@gmail.com	Change Email Address

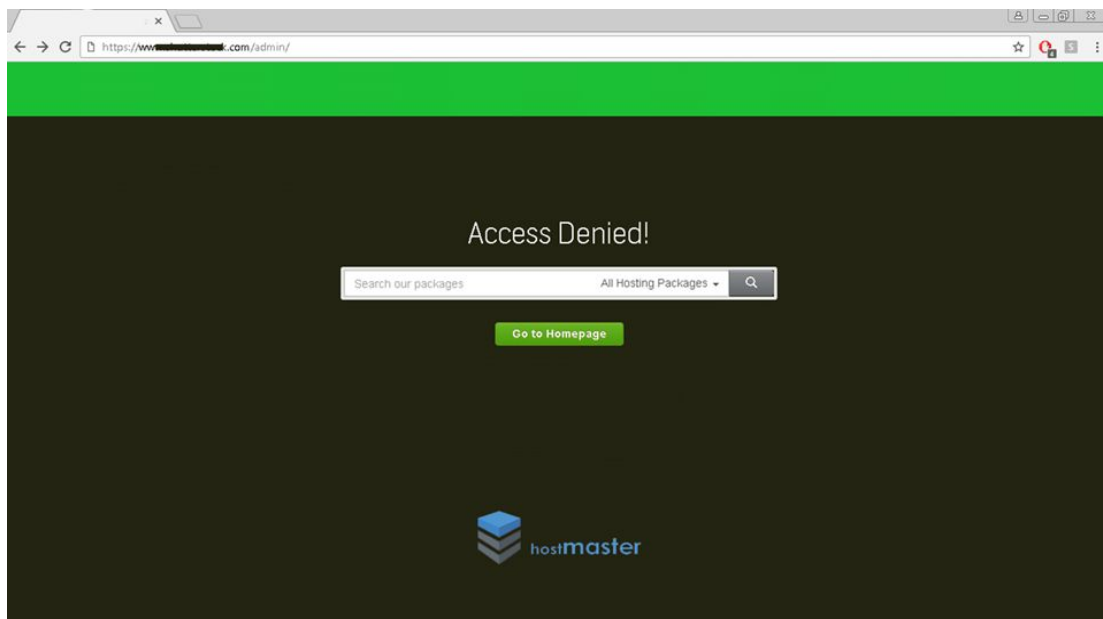
And other



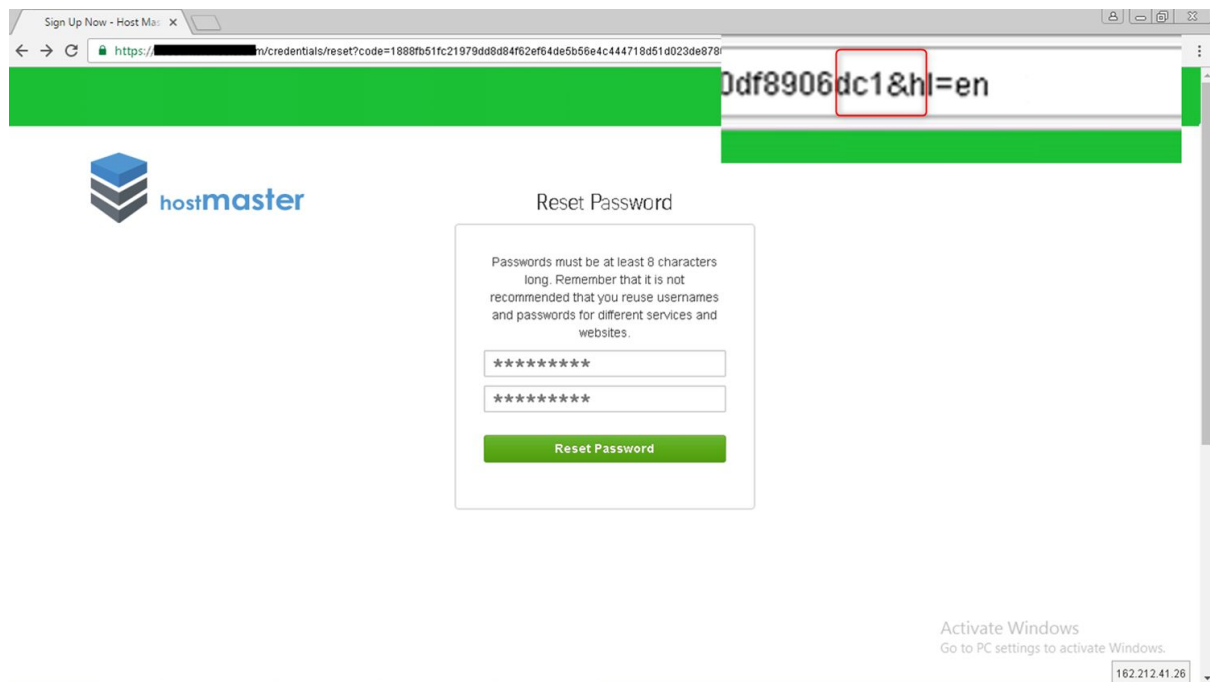
In this manner the team managed to change the passwords of all users in the system and obtain access to all accounts.

Admin page access

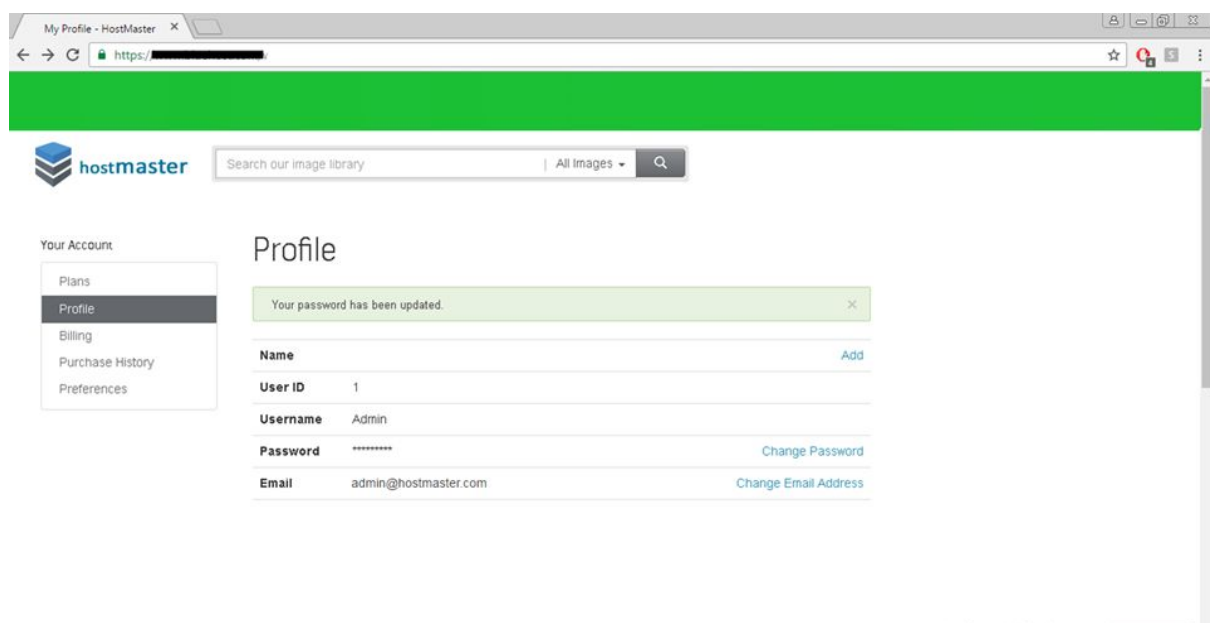
The next step was to try to access the admin panel, but it looked like the team didn't have enough permissions with any of the users.



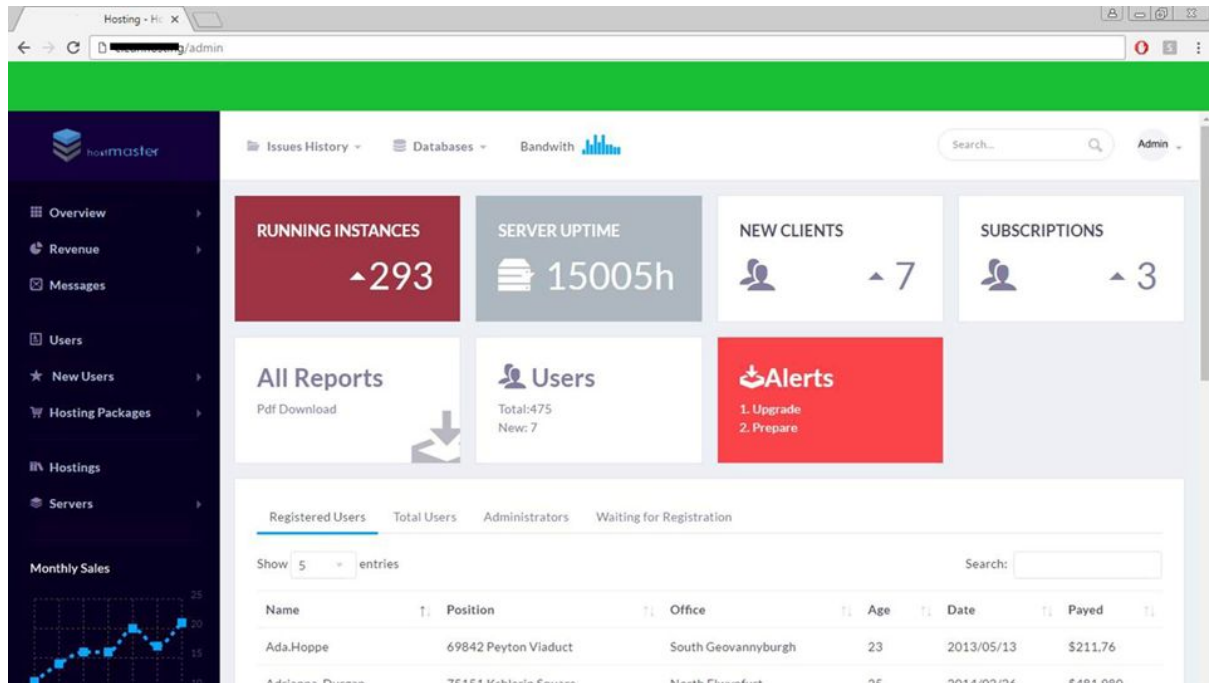
Coming back to the new vulnerability detected the team tried to reset the password of the user with the id=1. Why? When a new application is developed usually the first user created is the admin and has ID 1 in the database. If this doesn't work, then it could be -1.



And what was discovered is that the team managed to reset the password for the admin account.



Next step was to access the admin page once again.



The team had complete access over the application.

Conclusion

Even though the application is build with security requirements, an experienced penetration tester, with a background in development can find business logic vulnerabilities that can still produce a lot of risk. In this case, an attacker with complete admin access could have had the possibility to redirect payment transactions to his account, deny users to buy products, close servers of existing customers making them lose users and money.

The client was helped to mitigate the risk with support from the security team. Also, the client decided to include a penetration tester in the requirements and implementation phase of next web application developed.