

NAME-ANMOL KANT

OWASP TOP 5 APPLICATION SECURITY RISKS 2023

WE TALK ABOUT THE TOP 5 WEB SECURITY THREATS OR WE CAN SAY VULNERABILITIES OF WEB APPLICATIONS.SO HACKERS FIND A WAY TO HACK THE WEB APPLICATION AND ACCESS THEIR DATABASE.SO HERE WE DISCUSS SOME WAYS TO HACK A WEB APPLICATION.

1) Injection

WE TAKE A WEBSITE WHICH ARE VERY VULNERABLE AND TRY TO ACCESS THREE LOGIN PAGES THROUGH SQL INJECTOR. WE ALSO TRY AS A USER OR ADMIN.BASICALLY, IT INJECTS THE SQL QUERIES IN THE WEB APPLICATION TO ACCESS THAT WEB APPLICATION SO THAT WE CAN MAKE CHANGES IN THE DATABASE.

HERE I PERFORM THE INJECTION ATTACK I SHOW SCREENSHOTS OF LOGIN PAGES BEFORE INJECTION AND AFTER INJECTION.

BEFORE INJECTION.....

The screenshot shows a web browser window with the URL `testfire.net/login.jsp`. The page title is "Online Banking Login". On the left, there's a sidebar with navigation links for "PERSONAL" (Deposit Product, Checking, Loan Products, Cards, Investments & Insurance, Other Services), "SMALL BUSINESS" (Deposit Products, Lending Services, Cards, Insurance, Retirement, Other Services), and "INSIDE ALTORO MUTUAL" (About Us, Contact Us, Locations, Investor Relations, Press Room, Careers, Subscribe). The main content area has fields for "Username" and "Password" with a "Login" button below them. At the bottom, there are links for "Privacy Policy", "Security Statement", "Server Status Check", "REST API", and copyright information from 2023 Altoro Mutual, Inc. A note at the bottom right says "This web application is open source! Get your copy from GitHub and take advantage of advanced features". A red dashed box highlights a legal disclaimer at the very bottom of the page.

The Altoro website is published by IBM Corporation for the sole purpose of demonstrating the effectiveness of IBM products in detecting web application vulnerabilities and website defects. This site is not a real banking site. Similarities, if any, to third party products and/or websites are purely coincidental. This site is provided "as is" without warranty of any kind, either express or implied. IBM does not assume any risk in relation to your use of this website. For more information, please go to <http://www-142.ibm.com/software/us/en/subcategory/SW110>.

HERE WE SEE AN ALTORO MANUAL WEBSITE WHICH IS A VERY VULNERABLE OR WEAKER WEBSITE SO WE TRY TO BYPASS THE LOGIN USING INJECTION.

SO WE STUDIED IN THE DATABASE SYSTEM IF WE ANT TO CHANGE SOMETHING WE USE “OR” OPERATOR AND USE THIS “select * FROM USERS WHERE NAME='TOM' AND PASSWORD='1234”. SO HERE WE CAN SEE DATABASE WHICH WE STUDIED IN COLLEGE TIME. NOW WE USE “OR” INSTEAD OF AND AND MANIPULATE LIKE THIS “ or 1=1–” AND USE ANY PASSWORD BECAUSE AFTER USING O FDASH(-) WE DONT CARE WHAT WE WRITE IN PASSWORD SECTION JUST CONFIRM WE WRITE SOMETHING AFTER THAT WE ENTER IN THE WEB APPLICATION AS A ADMIN AND ACCESS THE DATABSE.

HERE IS ANOTHER SCREENSHOT AFTER WE USED INJECTION ON THIS APPLICATION.

The screenshot shows a web browser window with multiple tabs open. The active tab is 'testfire.net/bank/main.jsp'. The page itself is a仿冒的 Altoro Mutual website, featuring a green header with the Altoro Mutual logo and a 'DEMO SITE ONLY' watermark. The main content area displays a 'Hello Admin User' message and a congratulatory message about being pre-approved for a Gold Visa card with a credit limit of \$10000. The status bar at the bottom of the browser window contains the text 'This web application is open source! Get your copy from GitHub and take advantage of advanced features'.

SO THIS IS ALL ABOUT INJECTION BUT IF YOU WANT TO KNOW MORE ABOUT SQL INJECTION THEN YOU GO TO KALI LINUX OPEN TERMINAL TYPE cd Injections/ls and headSQL.txt. After that you get more SQL injectors you can also use them for your purpose.

2) Security Misconfiguration

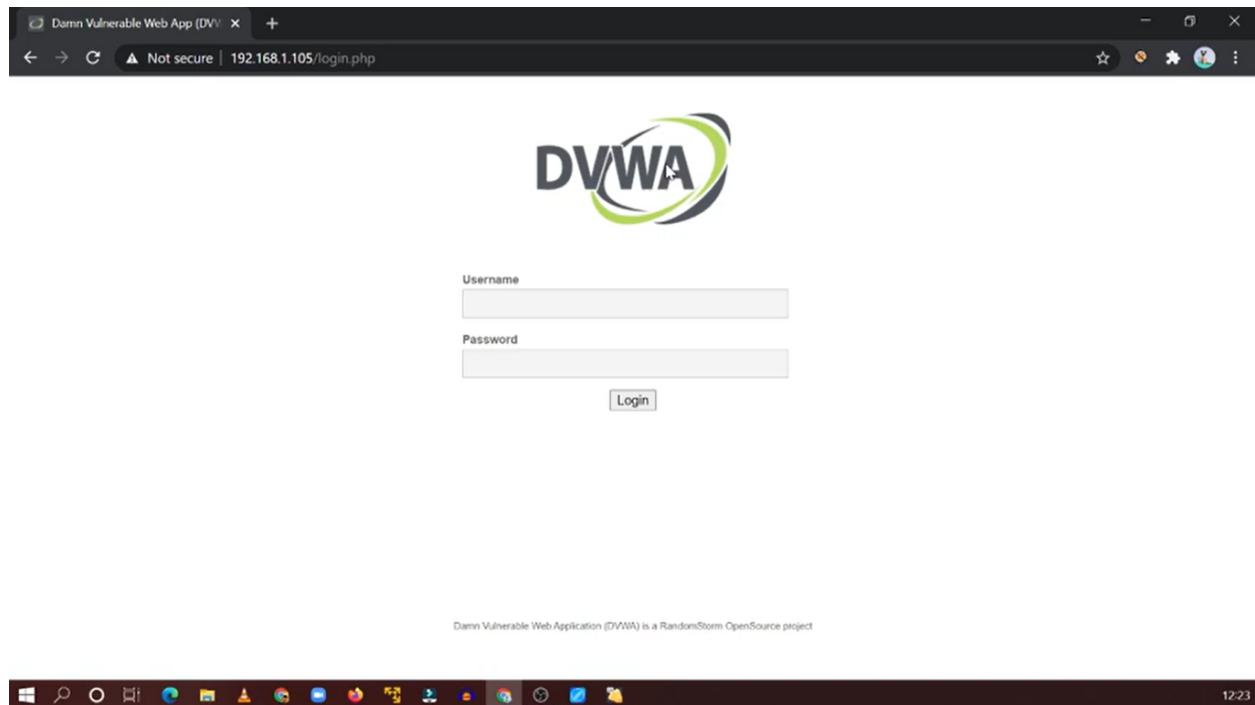
SO NOW WE TALK ABOUT SECURITY MISCONFIGURATION IN THIS VULNERABILITY. SUPPOSE WE USE A DVWA THAT IS BASICALLY A TOOL IN KALI LINUX TO PRACTICE THE MOST COMMON WEB VULNERABILITIES.

WE CHECK FIRST CASE,

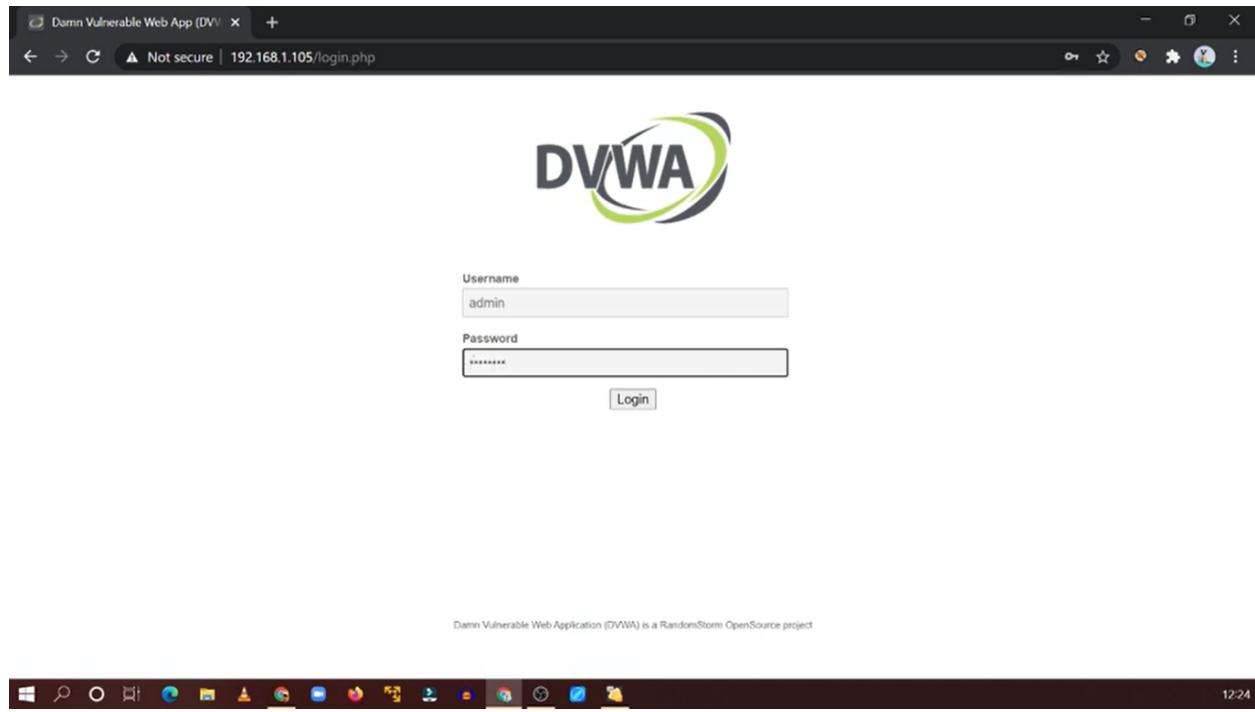
THE FIRST VULNERABILITY WE SAW WAS WHEN WE WENT TO THE LOGIN PAGE OF ANY WEB APPLICATION, FOR EXAMPLE, DVWA WE TRIED TO LOG IN USING A USERNAME AND PASSWORD SO JUST WE TRIED THE USERNAME AS ADMIN AND PASSWORD AS PASSWORD AND IT WORKED.SO THIS IS THE FIRST SECURITY VULENARBILTES WE SEE IF SOMEONE DOES NOT KNOW THE PASSWORD THEN HE/SHE MIGHT ALSO TRY BRUTE FROCE'S METHOD TO CRACK THE PASSWORD AND USERNAME SO THIS IS DEFINITELY A MISCONFIGURATION OF A WEB APPLICATION IF ANYONE VISITS THIS APPLICATION. THIS IS THE FIRST SECURITY MISCONFIGURATION.

WE ALSO SHOW SCREENSHOTS SO THAT IT MIGHT BE EASY FOR YOU TO UNDERSTAND.

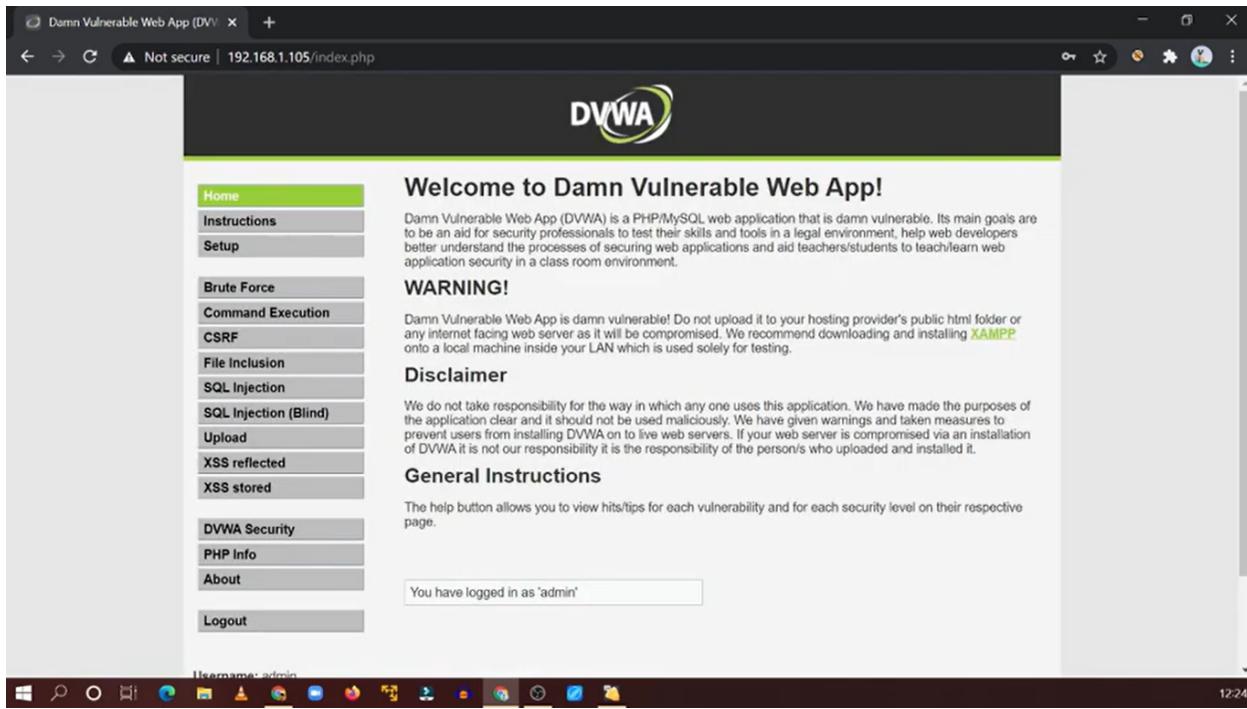
1)WE GO TO LOGIN PAGE OF A WEB APPLICATION.



2)WE TRY BRUTE FORCE OR RANDOM TYPE ADMIN AND PASSWORD INSTEAD OF USERNAME AND PASSWORD.



3)WE FINLAY LOGIN INTO A WEB APPLICATION WHICH IS A MISCONFIGURATION OF SECURITY BECAUSE IF ANYONE GOES TO THAT WEBSITE IT MIGHT BE EASIER TO CRACK THE PASSWORD AND USER NAME.



2)SECOND CASE OF MISCONFIGURATION,

THE SECOND THING WE TALK ABOUT IS FILE UPLOAD WEB SECURITY THREAT. SUPPOSE WE UPLOAD SOME FILE LIKE AN IMAGE AND WE TRY TO COPY THE LINK OF THE UPLOADED FILE AND PASTE IT INTO THE SEARCH BAR WE SAW THE IMAGE OF WILL COME THERE IS NO BASELINE OR MINIMAL SECURITY SO THAT ANYONE CAN HACK THAT EASILY WHEN A HACKER UPLOAD ANY THING LIKE KEYLOGGER FILE. HACKERS CAN ALSO SEND SHELL SCRIPT FILES AND WHEN WE OPEN THAT FILE WITHOUT ANY SECURITY OUR SYSTEM IS HACKED AND THEN ENTERED IN OUR SERVERS.SO THIS IS A SECURITY MISCONFIGURATION.

WE ALSO SHOW THEM WITH SCREENSHOTS FOR BETTER UNDERSTANDING,

The screenshot shows a web browser window for the Damn Vulnerable Web Application (DVWA) at the URL <http://192.168.1.105/vulnerabilities/upload/#>. The title bar says "Damn Vulnerable Web App (DVWA)". The main content area displays the DVWA logo and the heading "Vulnerability: File Upload". On the left, a sidebar menu lists various attack types: Home, Instructions, Setup, Brute Force, Command Execution, CSRF, File Inclusion, SQL Injection, SQL Injection (Blind), **Upload**, XSS reflected, XSS stored, DVWA Security, PHP Info, About, and Logout. The "Upload" option is highlighted with a green background. Below the menu, there is a form field labeled "Choose an image to upload:" with a "Choose File" button. A message indicates "No file chosen". Below the form is a "Upload" button. A success message at the bottom right of the form area says "..././.../hackable/uploads/logo.jpeg successfully uploaded!". Under the main content, there is a section titled "More info" with three links: http://www.owasp.org/index.php/Unrestricted_File_Upload, <http://blogs.securiteam.com/index.php/archives/1268>, and <http://www.acunetix.com/websesecurity/upload-forms-threat.htm>. The bottom of the screen shows a Windows taskbar with various icons and the system clock at 12:26.

The screenshot shows a Sublime Text editor window with the file path "C:\Users\amand\Desktop\demo.php - Sublime Text (UNREGISTERED)". The file contains the following PHP code:

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4
5 <h1>My first PHP page</h1>
6
7 <?php
8 echo "Hello Hacker!";
9 ?>
10
11 </body>
12 </html>
```

The code includes a PHP echo statement that outputs "Hello Hacker!". The Sublime Text interface shows multiple tabs at the top and a status bar at the bottom indicating "Line 12, Column 8", "Tab Size: 4", and "PHP".

Damn Vulnerable Web App (DVWA) Not secure | 192.168.1.105/vulnerabilities/upload/#

DVWA

Vulnerability: File Upload

Choose an image to upload:
 Choose File No file chosen

Upload

.../.../hackable/uploads/demo.php successfully uploaded!

More info

http://www.owasp.org/index.php/Unrestricted_File_Upload
<http://blogs.securiteam.com/index.php/archives/1268>
<http://www.acunetix.com/websitedevelopment/upload-forms-threat.htm>

Home
Instructions
Setup

Brute Force
Command Execution
CSRF
File Inclusion
SQL Injection
SQL Injection (Blind)
Upload
XSS reflected
XSS stored

DVWA Security
PHP Info
About

Logout

Username: admin

192.168.1.105/hackable/uploads/ X +

Not secure | 192.168.1.105/hackable/uploads/demo.php

My first PHP page

Hello Hacker!



3) INSECURE DESIGN

IT IS BASICALLY A VULNERABILITY IT IS A BROAD CATEGORY RELATED TO DESIGN AND ARCHITECTURAL FLAWS IN WEB APPLICATIONS THAT ARE EXPLOITED BY HACKERS. SUPPOSE A WEB DEVELOPER BUILDS A WEB APPLICATION OR SOFTWARE BUT HE/SHE DOES NOT MUCH FOCUS ON DESIGN OR WE CAN SAY AN ARCHITECTURAL VIEW THAT CAN CREATE A VULNERABILITY IN THE APPLICATION THAT IS AN INSECURE DESIGN. MANY TIMES WE SEE PEOPLE FIND BUGS ON MANY WEB APPLICATIONS AND COMPANIES LIKE GOOGLE REWARD WHOEVER FINDS THAT BUG. HOW DOES A BUG HAPPEN BECAUSE OF INSECURE DESIGN?

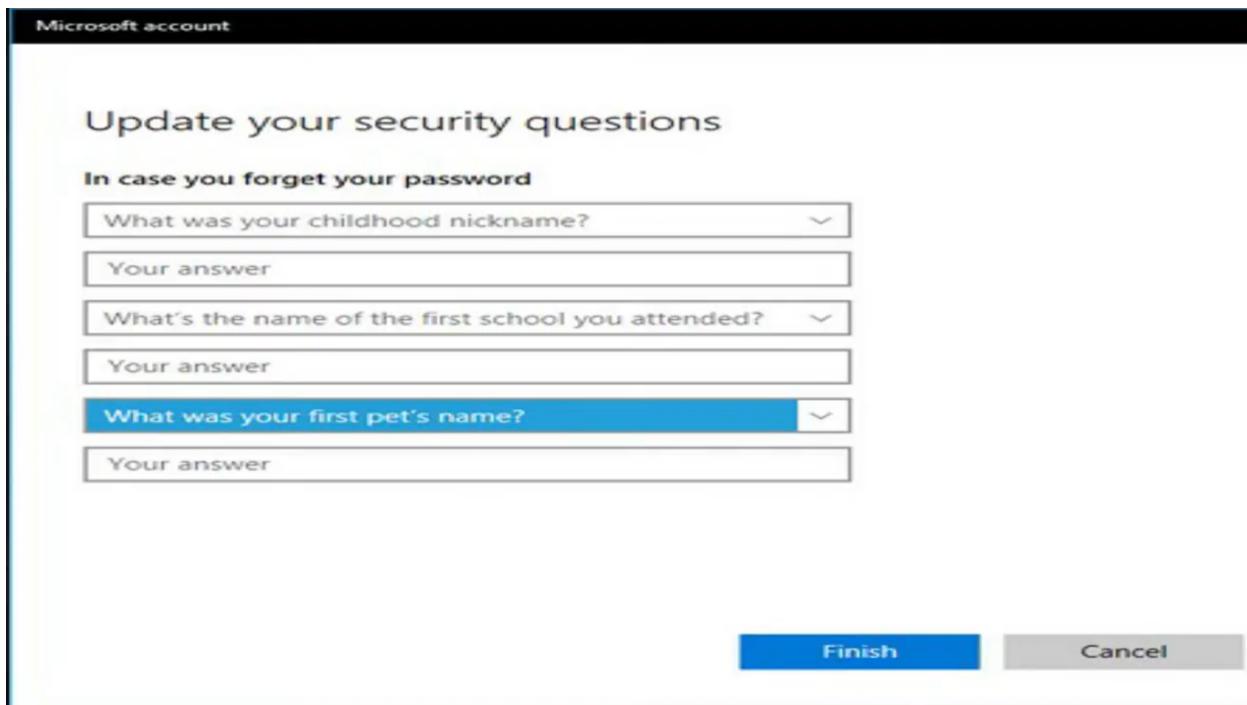
SO NOW WE SEE SOME BUGS THAT HAPPEN BECAUSE OF INSECURE DESIGN OR PEOPLE FIND THOSE BUGS AND REPORT TO WEB APPLICATION ADMIN.

- 1)OTP BYPASS**
- 2)BRUTE FORCE ATTACK**
- 3)SUPPOSE FOR RESETTING PASSWORD THEY USE A SECURITY QUESTION THEN THIS IS ALSO A VULNERABILITY.**
- 4)NO PROPER HANDLING OF ERRORS.**
- 5)HE ALSO DID NOT INCLUDE A FIREWALL TO PREVENT THE WANTED ATTACK.**
- 6)ACCOUNT TAKEOVER.**

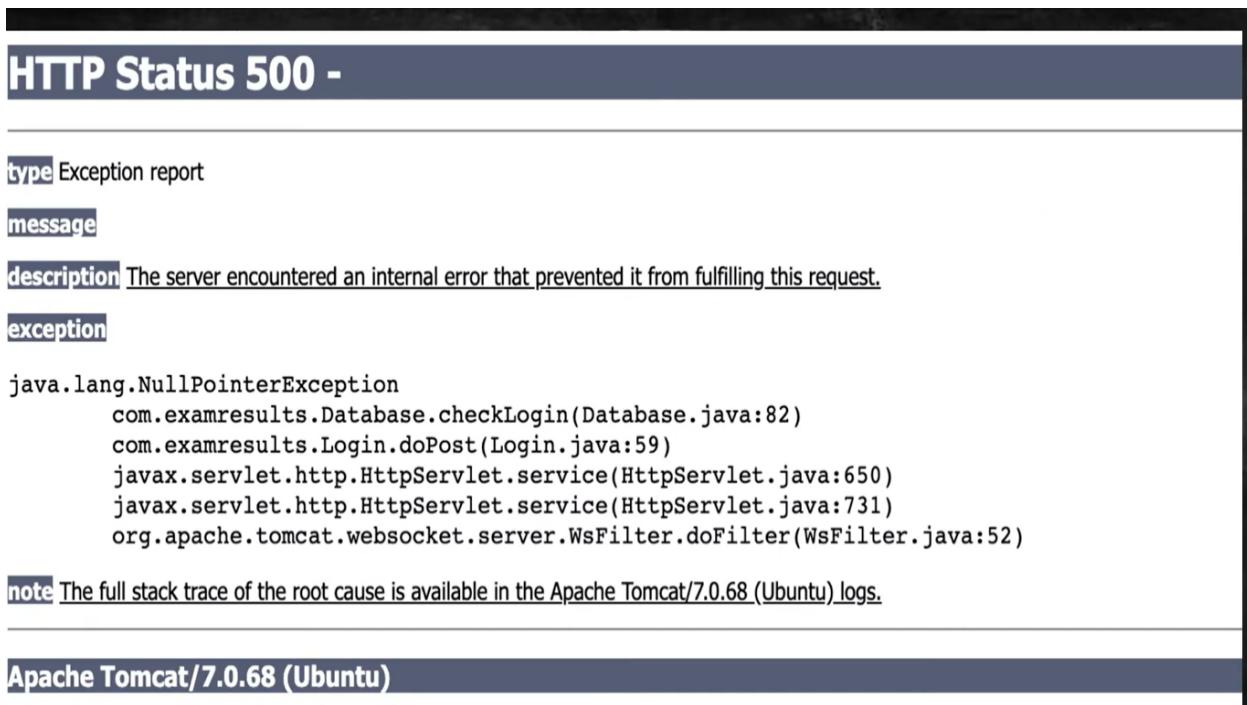
SO THESE ARE THE BUGS WE GENERALLY FIND IN ANY VULNERABLE WEB APPLICATION. WHEN YOU FIND SOMETHING SPECIFIC ABOUT INSECURE DESIGN YOU CAN NOT FIND ANYTHING YOU JUST FIND THESE BUGS BECAUSE THESE ALL COME UNDER INSECURE DESIGN.

SO HERE WE SHOW SOME SCREENSHOTS OF SOME BUGS FOR A BETTER UNDERSTANDING.

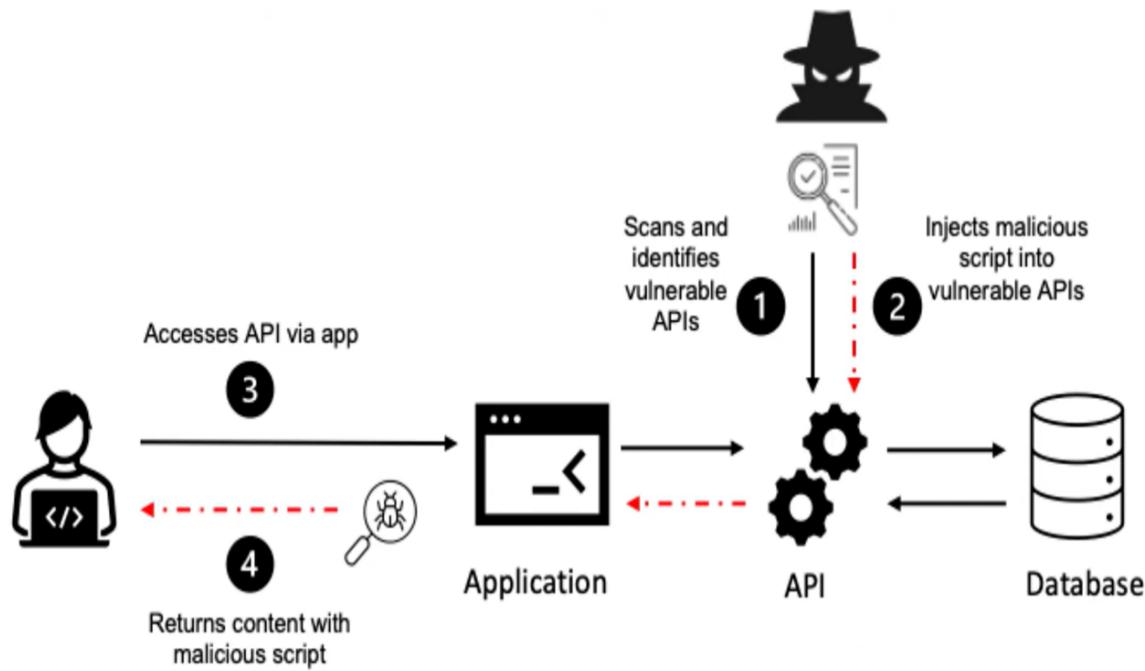
1) SUPPOSE FOR RESETTING PASSWORD THEY USE A SECURITY QUESTION THEN THIS IS ALSO A VULNERABILITY.



2) NO PROPER ERROR HANDLING.



DESIGN OF INSURE VULNERABILITY.



4) Broken Access Control

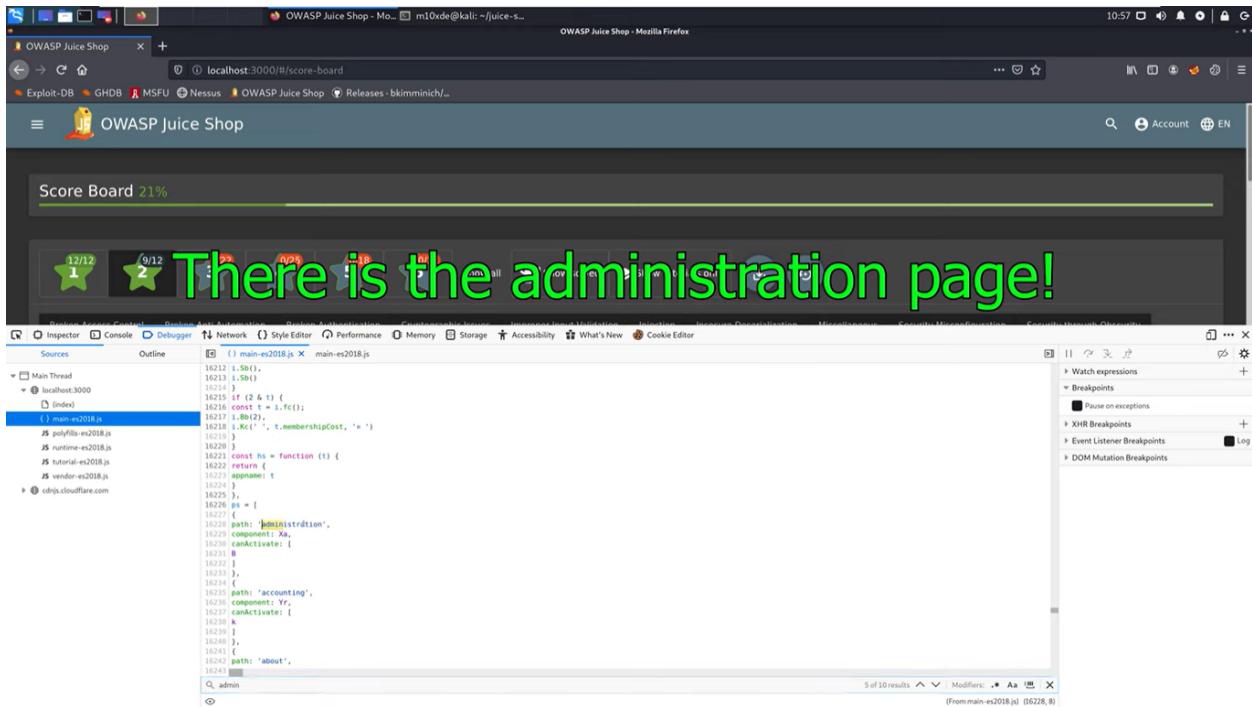
IT IS BASICALLY A VULNERABILITY WHERE AN ATTACKER GAINS UNAUTHORIZED ACCESS TO A RESTRICTED OR MANIPULATED SYSTEM OR INFORMATION.

OR WE CAN SAY WHEN HACKERS ENTER AN UNAUTHORIZED AREA AND TRY TO RESTRICT THE INFORMATION FROM OTHERS IT IS BASICALLY CALLED BROKEN ACCESS CONTROL.

WE ALSO ELABORATE ON THAT WAY SUPPOSE THERE ARE TWO USER WITH ID1 AND ID2 ONE WITH ID1 ACCESS ALL INFORMATION OF THE ID2 USER

WITHOUT PERMISSION IN A WEB APPLICATION THEN IT IS A BROKEN ACCESS CONTROL.

NOW WE TRY TO UNDERSTAND BY FOLLOWING EXAMPLE.



We aren't allowed to acces it. Good, that we already figured out how to log in as admin (see Login Admin or Password Strength)

You successfully solved a challenge: Admin Section (Access the administration section of the store.)

5)Cryptographic Failures

where attackers often target sensitive data, such as passwords, credit card numbers, and personal information, when you do not properly protect them.

Cryptographic failures can lead to serious security breaches, as attackers may be able to bypass encryption or decrypt sensitive data

SO WE JUST SHOW SOME DEMO OF HOW SOMEONE STEALS OUR CREDENTIALS USING MAN IN A MIDDLE ATTACK OR SNIFFING ATTACK WHICH COMES UNDER CRYPTOGRAPHIC FAILURE.

