nu11secur1ty Update README.MD ... on Sep 7, 2021 History

..	
Vulnerable-Code	last year
docs	last year
PoC-CVE-2021-38758-SQL-injection.py	last year
README.MD	last year
catering_1.zip	last year
chromedriver.exe	last year
install.txt	last year

README.MD

Base on checks for other issues for CVE-2021-38758 and I found SQL injection bypass login. :D

Software version and vendor:

- Online-Catering-Reservation-DT Food Catering (by: oretnom23) v1.0
- Type `1=1`
- [+] Proof

[+] Vulnerable Code:

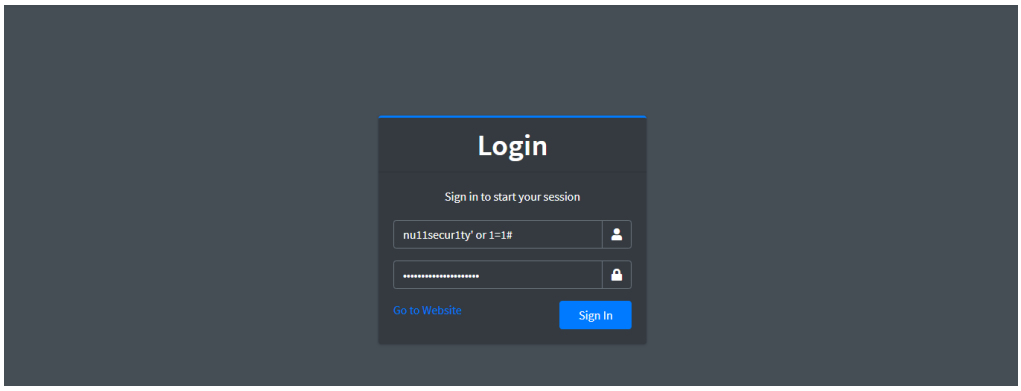
[+] Path: /catering/classes/Login.php

```
$qry = $this->conn->query("SELECT * from users where username = '$username' and password = md5('$password') ");
if($qry->num_rows > 0){
        foreach($qry->fetch_array() as $k => $v){
                if(!is_numeric($k) && $k != 'password'){
                        $this->settings->set_userdata($k,$v);
                }

        }
```

[+] Temporary fix of the problem, but not strong: :)

```
$qry = $this->conn->query("SELECT * from users where username = ('$username') and password = md5('$password') ");
if($qry->num_rows > 0){
        foreach($qry->fetch_array() as $k => $v){
                if(!is_numeric($k) && $k != 'password'){
                        $this->settings->set_userdata($k,$v);
                }

        }
```

- [+] Proof



Description:

```
The Online-Catering-Reservation-DT Food-Catering(by: oretnom23)v1.0 is vulnerable
in the application /catering/classes/Login.php which is called from /catering/dist/js/script.js app.
The parameter (username) from the login form is not protected correctly and there is no security and escaping from malicious
payloads.
When the user is sending a request to the MySQL server he can bypass the login credentials and take control of the administer
account.
```