DOM-based Cross-Site Scripting (XSS) in OpenEMR 7.0.0 and below at White list files in openemr/openemr

0

Valid Reported on Jul 21st 2022

Description

We would like to report the vulnerability we found during software testing. The OpenEMR 7.0.0 (latest version) and below version; Open Source electronic health records and medical practice management application; has DOM-based Cross-Site Scripting (XSS) vulnerability in the add-manually-input field on white list file page that never been reported before (We've checked from CVE Official website).

Vulnerability Type

DOM-based Cross-Site Scripting (XSS)

Affected Page/URL

https://<openemrurl>/interface/super/manage_site_files.php

Sample XSS Payload

```
"><script>alert(`CVE_Hunting`)</script>
```

Vulnerable Source Code

/var/www/localhost/htdocs/openemr/interface/super/manage_site_files.php (Please see more details in the occurrences section)

Implication

Client-side scripts are used extensively by modern web applications. They pe
functions (such as the formatting of text) up to full manipulation of client-side data and
Operating System interaction

Operating System interaction.
Unlike traditional Cross-Site Scripting (XSS), where the client is able to inject scripts into a request and have the server return the script to the client, DOM XSS does not require that a request be sent to the server and may be abused entirely within the loaded page.
This occurs when elements of the DOM (known as the sources) are able to be manipulated to contain untrusted data, which the client-side scripts (known as the sinks) use or execute an unsafe way.
Scanner has discovered that by inserting an HTML element into the page's DOM inputs (sources), it was possible to then have the HTML element rendered as part of the page by the sink.

Recommendation

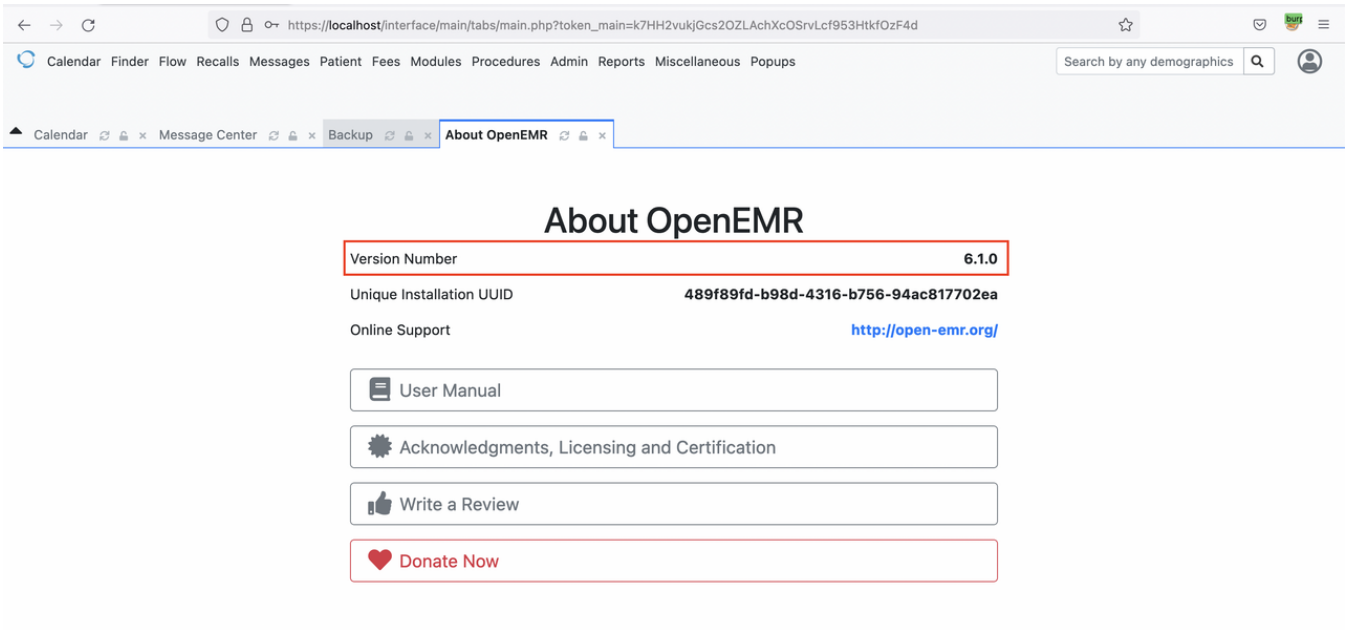
Client-side document rewriting, redirection, or other sensitive action, using untrusted data, should be avoided wherever possible, as these may not be inspected by server side filtering.
To remedy DOM XSS vulnerabilities where these sensitive document actions must be used, it is essential to
Ensure any untrusted data is treated as text, as opposed to being interpreted as code or mark-up within the page.
Escape untrusted data prior to being used within the page. Escaping methods will vary depending on where the untrusted data is being used. (See references for details.)
Use `document.createElement`, `element.setAttribute`, `element.appendChild`, etc. to build dynamic interfaces as opposed to HTML rendering methods such as `document.write`, `document.writeIn`, `element.innerHTML`, or `element.outerHTML`etc.

Discoverer/Reporters

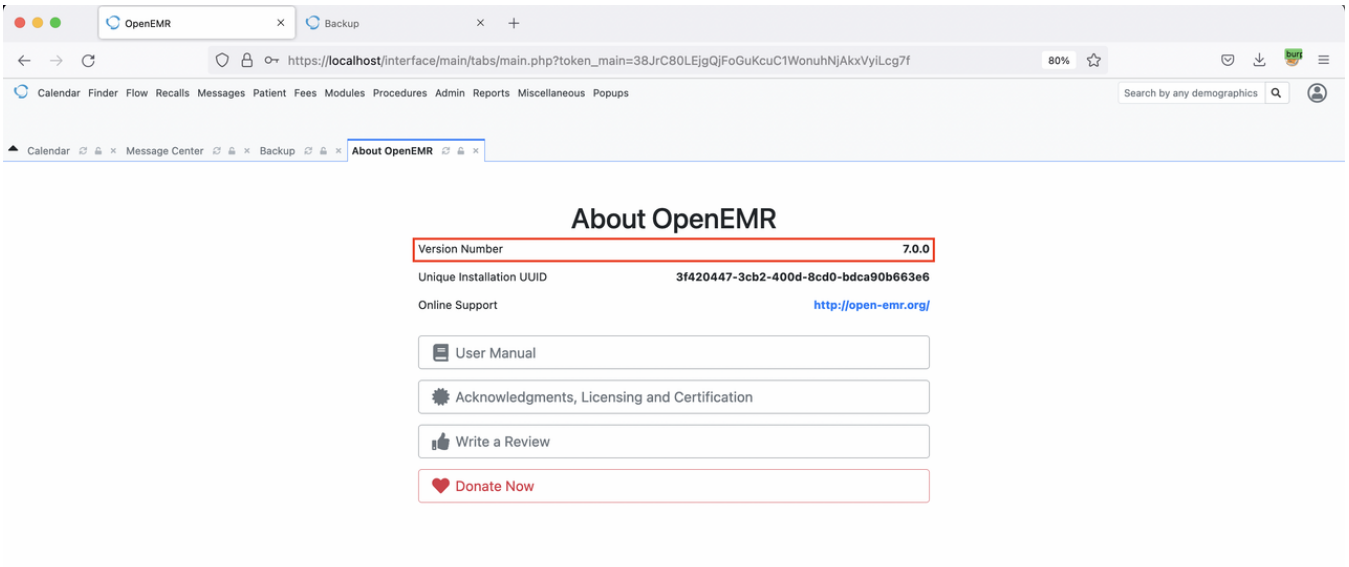
Ammarit Thongthua, Rattapon Jitprajong and Nattakit Intarasorn from Secure D Center Research Team

Example PoC Screenshots

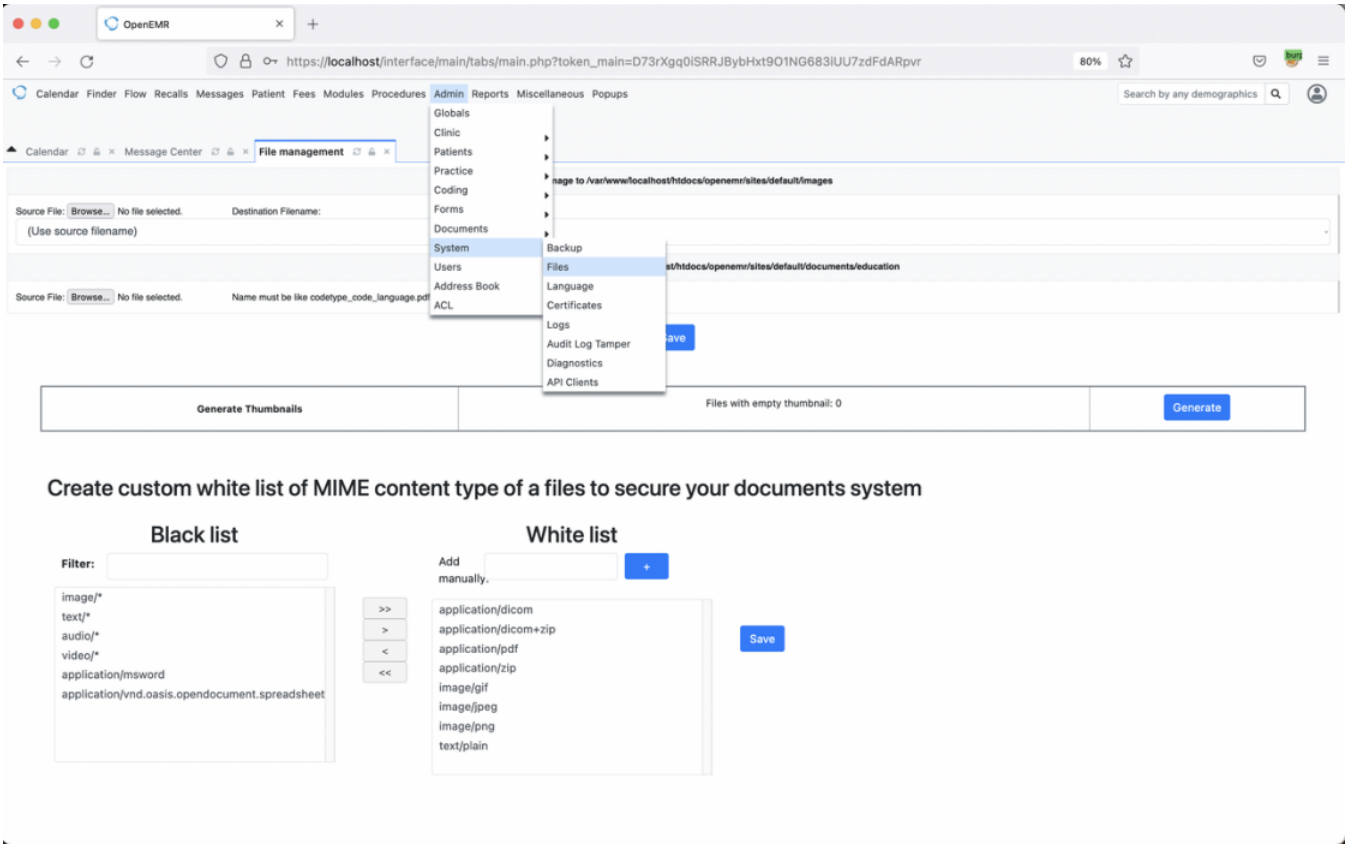
OpenEMR Version 6.1.0



OpenEMR Version 7.0.0

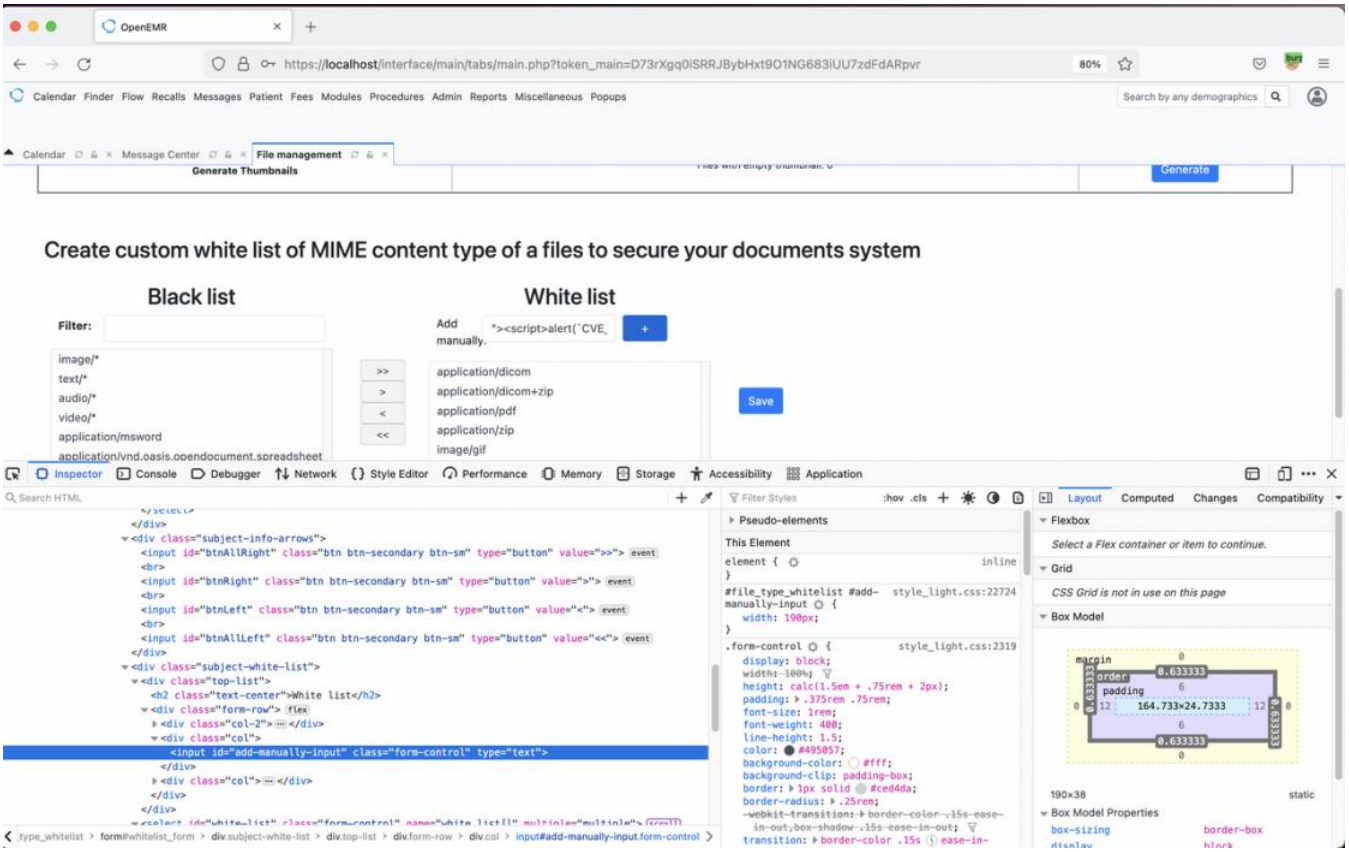


Admin > System > Files

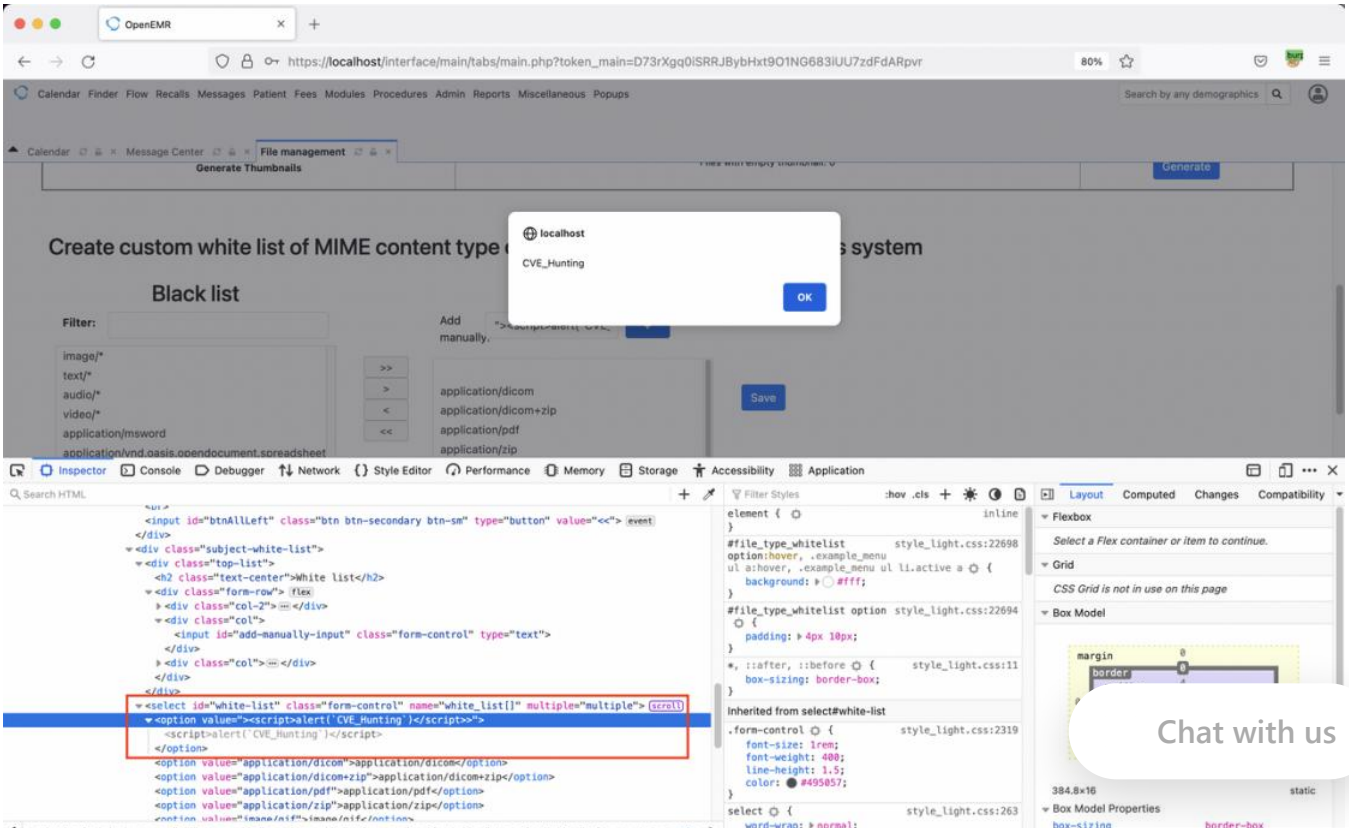


At the White list we Injected malicious JavaScript Payload in to {add-manually-input} with Sample XSS Payload

```
"><script>alert(`CVE_Hunting`)</script>
```



Renderred Malicious JavaScript (XSS)



Impact

DOM-based vulnerabilities arise when a website contains JavaScript that takes an attacker-controllable value, known as a source, and passes it into a dangerous function, known as a sink.

Occurrences

manage_site_files.php L537-L541

```
$('#add-manually').on('click', function () {
    var new_type = $("#add-manually-input").val();
    if(new_type.length < 1)return;
    $('#white-list').prepend("<option value="+new_type+">"+new_
})
```

References

- Reporting Security Vulnerabilities
- DOM-based XSS
- DOM Based XSS

CVE
CVE-2022-2729
(Published)

Vulnerability Type
CWE-79: Cross-site Scripting (XSS) - DOM

Severity
Medium (5.4)

Registry
Other

Affected Version
OpenEMR 7.0.0 and below

Visibility
Public

Status
Fixed

Found by

JohnNattakit
@johnnattakit
unranked

This report was seen 514 times.

We are processing your report and will contact the **openemr** team within 24 hours. 4 months ago

JohnNattakit modified the report 4 months ago

We have contacted a member of the **openemr** team and are waiting to hear back 4 months ago

Brady Miller validated this vulnerability 4 months ago

Thanks for the report. We are working on a fix.

JohnNattakit has been awarded the disclosure bounty ✔

The fix bounty is now up for grabs

The researcher's credibility has increased: +7

We have sent a fix follow up to the **openemr** team. We will try again in 7 days. 4 months ago

Brady Miller 4 months ago Maintainer

A preliminary fix has been posted in commit 74d21039aec641b2c406e3baf238ae4602a968b6

Please do not create a CVE # or make this vulnerability public at this time. I w
official about 1 week after we release 7.0.0 patch 1 (7.0.0.1), which will likely be i
After I do that, then will be ok to make CVE # and make it public.

Thanks!

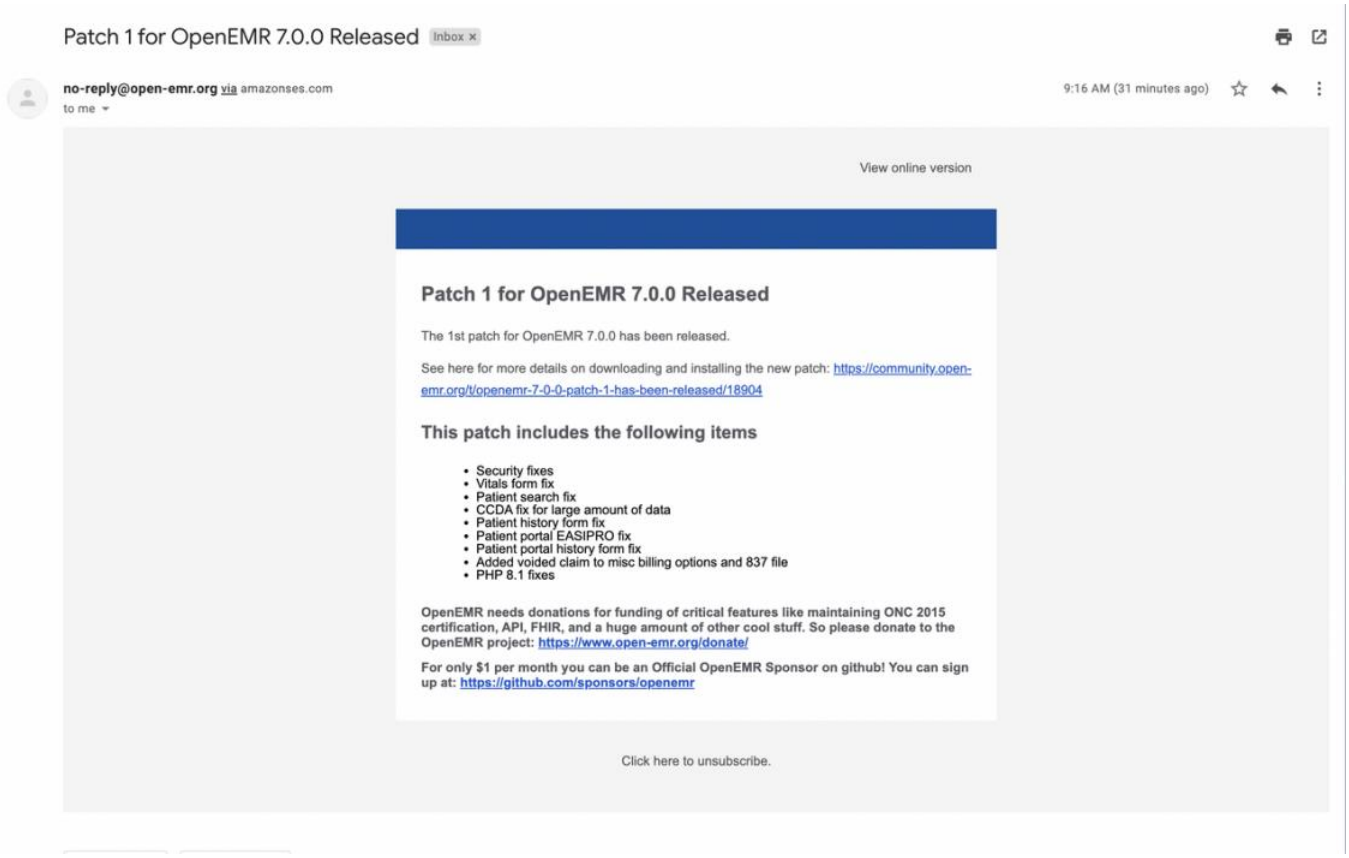
We have sent a second fix follow up to the **openemr** team. We will try again in 10 days.
4 months ago

JohnNattakit 4 months ago Researcher

Dear @Brady Miller, @admin

Hope you are doing well. We have got the notification email that the 1st patch for OpenEMR 7.0.0 has been released.

Can the CVE be assigned to this issue?



Jamie Slome 4 months ago Admin

Just waiting on the go-ahead from the maintainer, then I can arrange and publish the CVE 👍

Brady Miller marked this as fixed in 7.0.0.1 with commit 74d210 4 months a

The fix bounty has been dropped ✖

This vulnerability will not receive a CVE ✖

manage_site_files.php#L537-L541 has been validated ✔

Brady Miller 4 months ago Maintainer

OpenEMR patch 1 (7.0.0.1) has been released, so this has been fixed. You have permission to make CVE # and make this public.

JohnNattakit 4 months ago Researcher

Dear @Admin, @Jamie Slome
Could you please help to assign the CVE for this finding please?
Thanks and appreciate for your help 👍

@Brady Miller Thanks for your response 👍

Jamie Slome 4 months ago Admin

CVE assigned and will be automatically published in the next couple of hours 👍 ♥

JohnNattakit 4 months ago Researcher

@Jamie Slome Appreciate for your contribution 👍♥

Sign in to join this conversation