New issue

Jump to bottom

CVE-2020-12725 - Authenticated Server-Side Request Forgery (SSRF) in the JSON data source / internal addresses restriction bypass #4869

⊙ Closed 0xBADCA7 opened this issue on May 7, 2020 · 4 comments

0xBADCA7 commented on May 7, 2020 • edited ▾

Summary

Havoc Research discovered an authenticated Server-Side Request Forgery (SSRF) via the "JSON" data source of Redash open-source version. Possibly, other connectors are affected however the JSON data source provides a lot of flexibility in terms of being able to craft HTTP requests eg., by adding headers, selecting any HTTP verb, etc., making it a very handy tool for pivoting or persistence within internal networks. The final impact depends on the specifics of backend systems in use. To read more about this vulnerability class and its impact, please refer to https://portswigger.net/web-security/ssrf

Steps to reproduce the issue

1. Log in to your Redash instance. Our version under test was 8.0.0+b32245 (`a16f551`) (by way of pulling the redash/redash Docker image).
2. Add a new data source of "JSON" type. Skip other fields.
3. Create a new query backed by this data source you have just added.
4. In the query editor use this:

```
url: https://httpbin.org/redirect-to?url=http://172.17.0.1:22&status_code=302
```

Then execute the statement. Depending on how you deployed Redash you could see something like:

```
Error running query: ('Connection aborted.', BadStatusLine('SSH-2.0-OpenSSH_7.6\r\n'))
```

The httpbin.org service simply sends the `Location` header with the contents of the `url` parameter. 172.17.0.1 is the Docker container our Redash instance is running on and port 22 is assigned to the OpenSSH daemon on that box. Naturally, the address and port will vary depending on attackers' needs. For example, to retrieve the AWS meta-data, the payload could look like:

```
url: https://httpbin.org/redirect-to?url=http://169.254.169.254/latest/meta-data/&status_code=302
```

Remediation and root causes

It appears that there is a check that attempts to determine whether the address passed in, is a private address and then fails if so:

redash/redash/query_runner/json_ds.py
Line 38 in `9790b07`

```
38      def is_private_address(url):
```

This check can easily be bypassed by using HTTP redirects. To fix the issue, Redash can prevent its HTTP client from following redirects. This would not mitigate DNS-rebinding when using plain text connections, however. Other known approaches to fix the issue: a) to forward network calls via an outbound proxy, or b) allow connections only to specific addresses (or address ranges) through an ACL.

Notes

The proposed CVSSv3 score is 9.1 (Critical) worst-case scenario (https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator?vector=AV:N/AC:L/PR:H/UI:N/S:C/C:H/I:H/A:H&version=3.1) however the actual impact will depend on the environment the application is used in, typical to this vulnerability class. Exploitation requires a) authentication b) available "JSON" data source.

This SSRF can be very potent because this particular data source also allows setting custom headers, body and query parameters, method, etc.

Other data sources present the same vulnerability however the actual impact may vary depending on the amount of information returned back to the user. Some data sources are more verbose than others.

Mark Art
Havoc Research team

WD **weekly-digest** bot mentioned this issue on May 11, 2020

Weekly Digest (4 May, 2020 - 11 May, 2020) #4877

⊙ Open

0xBADCA7 commented on Jun 4, 2020 Author

The vendor requested to postpone the disclosure until June 11, 2020.

👍 1

0xBADCA7 changed the title ~~Vulnerability report: details TBA~~ **CVE-2020-12725 - Authenticated Server-Side Request Forgery (SSRF) in the JSON data source / internal addresses restriction bypass** on Jun 11, 2020

arikfr commented on Jun 15, 2020 Member

More details on possible workarounds and published fix:
GHSA-4599-9qr8-ccj6

Closing this now, but comments are still welcome.

arikfr closed this as completed on Jun 15, 2020

justinclift commented on Jun 15, 2020 Member

Trivial typo fix for the advisory page, if that's useful:

```
s/moment and concerned/moment and are concerned/
```

👍 1

arikfr commented on Jun 15, 2020 Member

@justinclift thanks, fixed.

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

3 participants