Closed Issue created 2 years ago by GitLab SecurityBot

Stored XSS on the Error Tracking page

HackerOne report #859888 by mike12 on 2020-04-26, assigned to @jeremymatos:

Hello Gitlab!

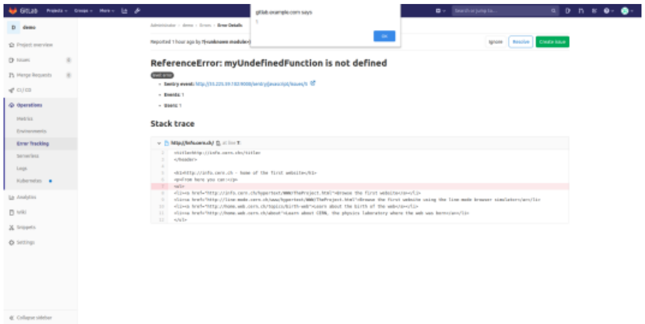
Steps to reproduce

1. Set up a Sentry server or **just use this server:** http://REDACTED .
    1. Install Sentry Server, use this doc
    2. Connect to the Sentry container: `docker exec -it sentry_onpremise_web_1 /bin/bash`
    3. Install Vim: `apt update && apt install vim -y`
    4. Open `stacktrace.py` file: `vim /usr/local/lib/python2.7/site-packages/sentry/interfaces/stacktrace.py`
    5. Replace `"colNo": self.colno,` with `"colNo": "<img src=x onerror=alert(1)>",` on line 210

```
def get_api_context(self, is_public=False, pad_addr=None, platform=None):
    from sentry.stacktraces.functions import get_function_name_for_frame

    function = get_function_name_for_frame(self, platform)
    data = {
        "filename": self.filename,
        "absPath": self.abs_path,
        "module": self.module,
        "package": self.package,
        "platform": self.platform,
        "instructionAddr": pad_hex_addr(self.instruction_addr, pad_addr),
        "symbolAddr": pad_hex_addr(self.symbol_addr, pad_addr),
        "function": function,
        "rawFunction": self.raw_function,
        "symbol": self.symbol,
        "context": get_context(
            lineno=self.lineno,
            context_line=self.context_line,
            pre_context=self.pre_context,
            post_context=self.post_context,
        ),
        "lineNo": self.lineno,
-       "colNo": self.colno,
+       "colNo": "<img src=x onerror=alert(1)>",
        "inApp": self.in_app,
        "trust": self.trust,
        "errors": self.errors,
    }
    if not is_public:
        data["vars"] = self.vars
```

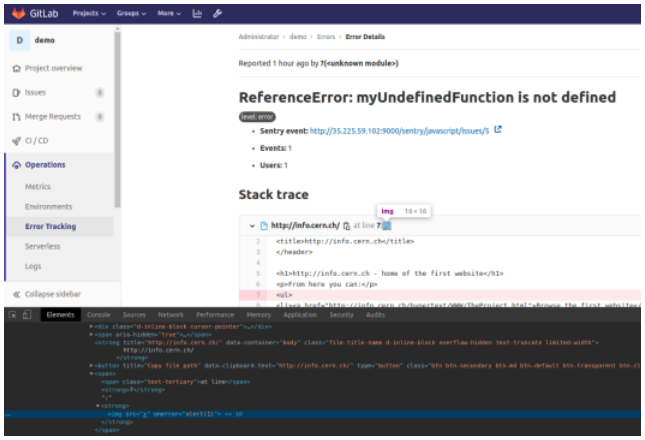
    6. Exit from the container
    7. Restart the container: `docker restart sentry_onpremise_web_1`
    8. Create a new Sentry project, use this doc
    9. Capture an error using Sentry SDK. Use docs: Initialize Sentry SDK and Capture your First Error
    10. Generate a Sentry auth token, use this doc
2. Run Gitlab: `docker run --detach --hostname gitlab.example.com --publish 443:443 --publish 80:80 --publish 22:22 --name gitlab gitlab/gitlab-ce:latest`
3. Create a new Gitlab project
4. Go to `Settings->Operations->Error Tracking` and connect Sentry to the Gitlab project
    1. Check the "Active" checkbox
    2. Sentry API URL: http://REDACTED:9000 (or your Sentry server URL)
    3. Auth Token: `REDACTED2` (or your Sentry auth token)
    4. Project: `Sentry | javascript` (or your Sentry project)
5. Go to `Operations->Error Tracking` and open details for an error.



gitlab.example.com says

ReferenceError: myUndefinedFunction is not defined

Stack trace



Administrator › demo › Errors › Error Details

Reported 1 hour ago by ?(<unknown module>)

ReferenceError: myUndefinedFunction is not defined

level: error

- Sentry event: http://35.225.59.102:9000/sentry/javascript/issues/5
- Events: 1
- Users: 1

Stack trace

http://info.cern.ch/ at line 7:

```
<title>http://info.cern.ch</title>
</header>

<h1>http://info.cern.ch - home of the first website</h1>
<p>From here you can:</p>
<ul>
```

```
<img src="x" onerror="alert(1)"> == $0
```

My GitLab version

```
root@gitlab:/# gitlab-rake gitlab:env:info

System information
System:
Current User:   git
Using RVM:      no
Ruby Version:   2.6.5p114
Gem Version:    2.7.10
Bundler Version:1.17.3
```

```
Rake Version:   12.3.3
Redis Version:  5.0.7
Git Version:    2.26.2
Sidekiq Version:5.2.7
Go Version:     unknown

GitLab information
Version:        12.10.1
Revision:       e658772bd63
Directory:      /opt/gitlab/embedded/service/gitlab-rails
DB Adapter:     PostgreSQL
DB Version:     11.7
URL:            http://gitlab.example.com
HTTP Clone URL: http://gitlab.example.com/some-group/some-project.git
SSH Clone URL:  git@gitlab.example.com:some-group/some-project.git
Using LDAP:     no
Using Omniauth: yes
Omniauth Providers:

GitLab Shell
Version:        12.2.0
Repository storage paths:
- default:      /var/opt/gitlab/git-data/repositories
GitLab Shell path:              /opt/gitlab/embedded/service/gitlab-shell
Git:            /opt/gitlab/embedded/bin/git
```

Impact

An attacker can:

1. Perform any action within the application that a user can perform
2. Steal sensitive user data
3. Steal user's credentials

Attachments

Warning: Attachments received through HackerOne, please exercise caution!

- 2.png
- 1.png

Edited 2 years ago by Jeremy Matos

Drag your designs here or click to upload.

Tasks 0

No tasks are currently assigned. Use tasks to break down this issue into smaller parts.

Linked items 0

Link issues together to show that they're related or that one is blocking others. Learn more.

Activity

GitLab SecurityBot added HackerOne security labels 2 years ago

GitLab SecurityBot added priority 4 severity 4 scoped labels 2 years ago

GitLab SecurityBot @gitlab-securitybot · 2 years ago Author Reporter

HackerOne comment by `jmatos_bgtvf` :

Hello [@]mike12,

Thank you for this report. I confirm that a XSS alert can be triggered when visiting Operations/Error Tracking page if sentry contains an error with such content. Yet I am struggling to assess the impact.

Do you have in mind a scenario where

- victim has configured its project with its own sentry instance/project
- attacker is able to push an error with such content in this sentry instance/project

Best regards, GitLab Security Team

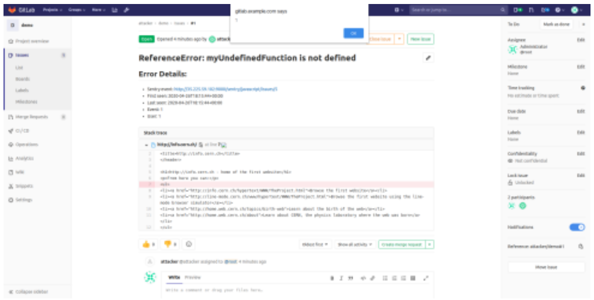
GitLab SecurityBot @gitlab-securitybot · 2 years ago Author Reporter

HackerOne comment by `mike12` :

Thank you for your response!

Yes, I have an attack scenario using this vulnerability

1. Create a new Gitlab project
2. Connect Sentry to the Gitlab project using the instructions above
3. Go to `Settings->Members` and add a victim to the project. In my case, I added `root` user
   1. GitLab member or Email address: root
   2. Choose a role permission: Reporter (or higher)
4. Now we should convince the victim to visit the page with stored XSS. We can simply send a link to the Error Tracking page to the victim. But I think we have a better alternative:
   1. Go to `Operations->Error Tracking`
   2. Open the error details and click on the "Create issue" button
   3. Assign the created issue to the victim
   4. The victim will be notified about the assigned issue
5. The victim opens the assigned issue
6. The XSS fires on the issue page


gitlab.example.com says

ReferenceError: myUndefinedFunction is not defined

Error Details:

Attachments

Warning: Attachments received through HackerOne, please exercise caution!

- scenario1.png

GitLab SecurityBot added security-group-missing security-triage-appsec labels 2 years ago

Jeremy Matos @jeremymatos · 2 years ago Contributor

@sarahwaldner @ClemMakesApps Confirmed by using the (modified) sentry server of the reporter. Even if this scenario is not easy to exploit, we should be output encoding any data coming from Sentry to avoid XSS (cf https://docs.gitlab.com/ee/development/secure_coding_guidelines.html#xss-guidelines)

Edited by Jeremy Matos 2 years ago

Clement Ho @ClemMakesApps · 2 years ago Contributor

Yes, we should be sanitizing the data. This is an interesting one by compromising the Sentry data.

@ohoral could you help determine if this is frontend or backend so that we can set the right labels? I don't remember if we just pass the error data through the API to the frontend or whether the backend can do processing. Thinking out loud, it would probably be a good idea to sanitize on both frontend and backend if available

Sarah Waldner @sarahwaldner · 2 years ago Developer

Thanks @jeremymatos. I will let @ClemMakesApps and @crystalpoole help determine when we should work on this.

Olena Horal-Koretska @ohoral · 2 years ago Maintainer

I think the primary security issue here is that hacker gets access to Sentry server code and can update it. This should be addressed first. If it still happens, that Sentry instance is hacked, any field should be sanitized. As far as I know, BE doesn't do any processing of data coming from Sentry (besides stack trace code). @seanarnold, could you confirm this, please?

Also, it makes me think that any 3rd party we integrate with can be compromised 😔

I'll still have to discuss it more with BE engineers to decide which side should take care.

Sean Arnold @seanarnold · 2 years ago Maintainer

It looks like the data is escaped when returned from the API:

```
{"error":{"issue_id":"474080","date_received":"2020-05-03T07:40:04.274Z","stack_trace_entries":[{"function"
```

I tried adding that to a few other fields (title, file path) & I didn't get an XSS with the alert showing. Maybe there is something different with how the frontend is handling & rendering the `colNo` vs other data into the page?

Clement Ho @ClemMakesApps · 2 years ago Contributor

@ohoral could you help investigate what level of effort is needed here so that we can determine when to schedule this work?

Peter Leitzen @splattael · 2 years ago Maintainer

@seanarnold

> I tried adding that to a few other fields (title, file path) & I didn't get an XSS with the alert showing.

Have you tried `lineNo` ? 🤔

> Maybe there is something different with how the frontend is handling & rendering the `colNo` vs other data into the page?

👍 My first impression was that `errorPositionText()` might be missing some `escape` ing similar what we've done in `errorFnText()` 🤷

Allison Browne @allison.browne · 2 years ago Maintainer

It looks like we are interpolating directly into a string of html and then calling v-html on the whole thing which will render user input as html/code even if escaped:

https://gitlab.com/gitlab-org/gitlab/-/blob/d12e63eb59d24a5fcc821b4de1dae664a9128b16/app/assets/javascripts/error_tracking/components/stacktrace_entry.vue#L79

https://gitlab.com/gitlab-org/gitlab/-/blob/d12e63eb59d24a5fcc821b4de1dae664a9128b16/app/assets/javascripts/error_tracking/components/stacktrace_entry.vue#L135

> Only use v-html on trusted content and never on user-provided content.

- https://vuejs.org/v2/api/#v-html

JSFiddles demonstrating:

with v-html

versus without

without v-html

Edited by Allison Browne 2 years ago

Sean Arnold @seanarnold · 2 years ago Maintainer

Confirmed this also happens on `lineNo` . @splattael

Great find @allison.browne - that indeed seems like the issue. Thanks for providing JSFiddles too.

Olena Horal-Koretska @ohoral · 2 years ago Maintainer

> Great find @allison.browne - that indeed seems like the issue indeed, good catch

thank you @seanarnold @splattael and @allison.browne for your help

> @ohoral could you help investigate what level of effort is needed here so that we can determine when to schedule this work?

@ClemMakesApps this would be a small frontend change. But as this is a security issue, we'd have to follow security flow with creating backports to 3 latest releases. The deadline for security fixes for the next security release is May 28th. I'd say we still have some time

Clement Ho @ClemMakesApps · 2 years ago Contributor

Since it's a ~P4, there isn't any pressure to push for May 28th due date. I'd be happy to include this into %13.1 planning where we have some more capacity compared to right now (may still end up making the May 28 due date but not committing to it). @sarahwaldner thoughts on my proposal?

Sarah Waldner @sarahwaldner · 2 years ago Developer

@ClemMakesApps Proposal sounds great. I'll add to %13.1 .

Please register or sign in to reply

Jeremy Matos added group respond devops monitor scoped labels 2 years ago

GitLab SecurityBot removed security-group-missing security-triage-appsec labels 2 years ago

Olena Horal-Koretska assigned to @ohoral 2 years ago

Sarah Waldner changed milestone to %13.1 2 years ago

Clement Ho added Filler label 2 years ago

Olena Horal-Koretska added workflow in dev scoped label 2 years ago

Olena Horal-Koretska @ohoral · 2 years ago Maintainer

1. Please provide a quick summary of the current status (one sentence)

MR targeting `security` repo `master` branch was sent for initial review today

1. When do you predict this feature to be ready for maintainer review
   Early next week
2. Are there any opportunities to further break the issue or merge request into smaller pieces (if applicable)? Backport MRs are coming after MR targeting master is approved

/cc @ClemMakesApps

Olena Horal-Koretska removed workflow in dev label 2 years ago

Costel Maxim @cmaxim · 2 years ago Developer

Issue fixed in 13.1.2

Costel Maxim closed 2 years ago

GitLab SecurityBot @gitlab-securitybot · 2 years ago

Author Reporter

This HackerOne security issue was closed 30 days ago and may become public.

Please ensure the following items are true and add a ✅ reaction:

- Issue description and comments do not contain sensitive data belonging to GitLab.
- Issue does not reveal private information of the reporter (i.e. session IDs, passwords).

If the issue needs to stay confidential, please add the keep confidential label.

If you removed confidential data from the issue description before making it public, make sure that the description history entry is deleted.

Jeremy Matos @jeremymatos · 2 years ago

Contributor

Making issue public after redacting IP address and auth token

Please register or sign in to reply

Jeremy Matos changed the description 2 years ago ·

Jeremy Matos made the issue visible to everyone 2 years ago

Please register or sign in to reply