Closed · Issue created 1 year ago by GitLab SecurityBot

Stored XSS in DataDog Integration affects maintainers/owners

HackerOne report #1257383 by shells3c on 2021-07-11, assigned to GitLab Team:

Report | How To Reproduce

Report

Summary

The code: https://gitlab.com/gitlab-org/gitlab/-/blob/master/app/models/integrations/datadog.rb

```
def fields
 ...
 {
       type: 'password',
       name: 'api_key',
       title: _('API key'),
       non_empty_password_title: s_('ProjectService|Enter new API key'),
       non_empty_password_help: s_('ProjectService|Leave blank to use your current API key'),
       help: "<a href=\"#{api_keys_url}\" target=\"_blank\">API key</a> used for authentication with Datadog",
       required: true
 }

 ...

 def api_keys_url
   return URL_API_KEYS_DOCS unless datadog_site.presence

   sprintf(URL_TEMPLATE_API_KEYS, datadog_site: datadog_site)
 end
```

The `api_keys_url` is not sanitized before being rendered, leads to XSS, affects any maintainer/owner of the project when they visit the DataDog Integration

Steps to reproduce

1. Go to DataDog Integration setting: https://gitlab.com/user/project/-/services/datadog/edit (requires permission)
2. In the **Datadog site** field, set the value to `"><script>alert()</script>`, then **Save**
3. Now XSS will be activated in https://gitlab.com/user/project/-/services/datadog/edit, send this link to victims

Impact

Stored XSS affects maintainers/owners

How To Reproduce

Please add reproducibility information to this section:

1.
2.
3.

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.

Related merge requests 2

Revise helptext on Datadog page
!66391 · 14.2

Use v-safe-html for field help texts in integration form
!67890 · 14.2

Activity

GitLab SecurityBot changed due date to September 19, 2021 1 year ago

GitLab SecurityBot added HackerOne security labels 1 year ago

GitLab SecurityBot added Weakness CWE-79 priority 2 severity 2 scoped labels 1 year ago

GitLab SecurityBot @gitlab-securitybot · 1 year ago · Author · Reporter

HackerOne comment by shells3c :

After consideration, I believe the severity is high because I can create my own project, invite the victim to the project as maintainer and then send the link. So no privilege is required either the attacker and victim

GitLab SecurityBot @gitlab-securitybot · 1 year ago · Author · Reporter

HackerOne comment by princeofpersia :

Hi [@]shells3c,

Thank you for your submission. I hope you are well. Your report is currently being reviewed and the HackerOne triage team will get back to you once there is additional information to share.

Have a great day!

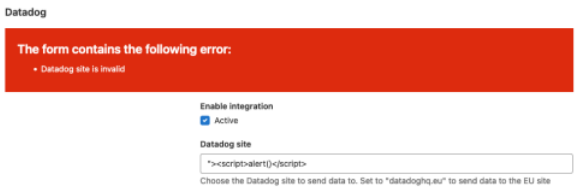
Kind regards, [@]princeofpersia

GitLab SecurityBot @gitlab-securitybot · 1 year ago · Author · Reporter

HackerOne comment by princeofpersia :

Hi [@]shells3c,

I am unable to reproduce the issue on my side, as seen below the payload is not accepted in the `Datadog site` field.



Regards, [@]princeofpersia

Attachments

Warning: Attachments received through HackerOne, please exercise caution!

- Screen_Shot_2021-07-12_at_3.15.31_PM.png

GitLab SecurityBot @gitlab-securitybot · 1 year ago · Author · Reporter

HackerOne comment by shells3c :

Hi [@]princeofpersia, please don't check the **Active** box

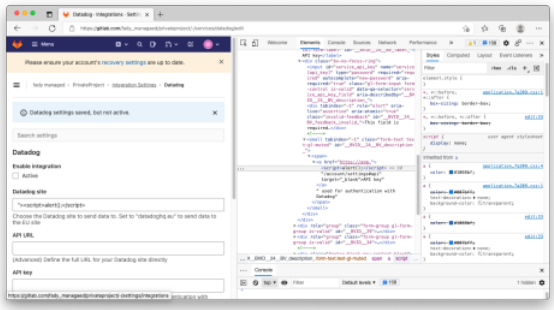
GitLab SecurityBot @gitlab-securitybot · 1 year ago Author Reporter

HackerOne comment by princeofpersia :

Hi [@]shells3c,

Thanks for the update!

I can insert the payload now, however the XSS is not triggered, I can see the payload in the source but it doesn't get executed (No CSP errors are displayed and I tried on an instance without CSP and it didn't work as well).



Regards, [@]princeofpersia

Attachments

Warning: Attachments received through HackerOne, please exercise caution!

- Screen_Shot_2021-07-17_at_3.24.56_PM.png

GitLab SecurityBot @gitlab-securitybot · 1 year ago Author Reporter

HackerOne comment by shells3c :

Hi [@]princeofpersia,

I don't know why by somehow I was unable to execute the javascript code inside the `<script>` tag too, but I'm still exploit the XSS.

Payload:

```
" style=animation-name:blinking-dot onanimationstart=alert(document.domain) other
```

GitLab SecurityBot @gitlab-securitybot · 1 year ago Author Reporter

HackerOne comment by princeofpersia :

Hi [@]shells3c,

We were able to verify your report in our side, we forwarded the report to gitlab team, we will let you know once we hear back from them.

Regards, [@]princeofpersia

GitLab SecurityBot @gitlab-securitybot · 1 year ago Author Reporter

HackerOne comment by shells3c :

Hi, here is the CSP bypass payload:

```
"><iframe xmlns="http://www.w3.org/1999/xhtml" srcdoc="<script src=https://gitlab.com/bugbountyuser1/csp/-/jobs/1
```

GitLab SecurityBot @gitlab-securitybot · 1 year ago Author Reporter

HackerOne comment by dcouture :

Upgrading severity given CSP bypass. The payload fires on an obscure page but it's not worse than a reflected XSS at this point.

GitLab SecurityBot added security-group-missing security-triage-appsec labels 1 year ago

Nikhil George @ngeorge1 · 1 year ago Developer

As this issue is related to integration, it looks like this belongs to the ~"group::ecosystem" and ~devops ~create. Adding the same group labels, but please feel free to change them.

Nikhil George added group ecosystem [DEPRECATED] devops create scoped labels 1 year ago

GitLab SecurityBot removed security-group-missing security-triage-appsec labels 1 year ago

GitLab SecurityBot @gitlab-securitybot · 1 year ago Author Reporter

@mushakov @arturoherrero @leipert @mhenriksen This issue is ready for triage as per HackerOne process.

About this automation: AppSec Escalation Engine

GitLab SecurityBot added security-set-milestone label 1 year ago

Lukas 'Eipi' Eipert @leipert · 1 year ago Developer

Interesting. This comes all down to us allowing HTML in the `help` text of a field. We should definitely use `safe-html`.

In this specific case the HTML is injected via https://gitlab.com/gitlab-org/gitlab/-/blob/2b2c4e7e6a0796603827231 8d00568174d5dad13/app/models/integrations/datadog.rb#L83. I wonder if any other help text would be vulnerable to the same thing... and did a quick search and didn't come up with other places...

Lukas 'Eipi' Eipert @leipert · 1 year ago Developer

I think the smallest iteration to fix this: Just remove the link from the URL for now? Or maybe only use it during `sprintf` if it is actually valid.

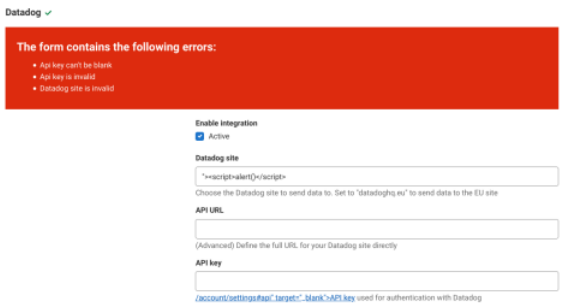
@toupeira Could you have a look and see whether I am making sense?

Edited by Lukas 'Eipi' Eipert 1 year ago

Markus Koller @toupeira · 1 year ago Contributor

@leipert yes that makes sense, we don't really need to link from the help text and can just remove it! 👍

I'm not sure how to reproduce this vulnerability though, we have a format validation on `Datadog site` which only allows alphanumeric domain names. I do see the broken link while editing but can't get the form to save:



You can set the value directly in the DB but I don't think it's possible to circumvent through the app (UI or API), unless I'm missing something.

@dcouture were you able to reproduce?

Lukas 'Eipi' Eipert @leipert · 1 year ago Developer

@toupeira I was able top reproduce with

```
"><iframe xmlns="http://www.w3.org/1999/xhtml" srcdoc="<script src=https://gitlab.com/bugbountyuser1/csp/-/
```

Markus Koller @toupeira · 1 year ago Contributor

@leipert oh I missed the part about having to disable the integration, which also disables the validation 😅 #336614 (comment 632494148)

I can see it now too, nevermind! 👍

I'll give this a weight of 1 since the fix should be very straightforward.

Markus Koller @toupeira · 1 year ago Contributor

We could also just hard-code the link to https://docs.datadoghq.com/account_management/api-app-keys/, and also add it in the alternate label where it's currently missing:

I think this would be useful because I was just trying to set up the Datadog integration and wondering where I get the token, and this is exactly the link I needed 😍

Justin Ho Tuan Duong @justin_ho · 1 year ago Maintainer

I agree that we should definitely be using `v-safe-html` here. I have this small diff that applies that (though I don't think I can do the full MR before going on leave). If @toupeira is already working on this, do you mind including this in your changes? We can maybe verify separately that this works without any backend changes.

```
diff --git a/app/assets/javascripts/integrations/edit/components/dynamic_field.vue b/app/assets/javascripts/
index 3655f94f06f..97058eecf6a 100644
--- a/app/assets/javascripts/integrations/edit/components/dynamic_field.vue
+++ b/app/assets/javascripts/integrations/edit/components/dynamic_field.vue
@@ -1,6 +1,12 @@
 <script>
-/* eslint-disable vue/no-v-html */
-import { GlFormGroup, GlFormCheckbox, GlFormInput, GlFormSelect, GlFormTextarea } from '@gitlab/ui';
+import {
+  GlFormGroup,
+  GlFormCheckbox,
+  GlFormInput,
+  GlFormSelect,
+  GlFormTextarea,
+  GlSafeHtmlDirective as SafeHtml,
+} from '@gitlab/ui';
 import { capitalize, lowerCase, isEmpty } from 'lodash';
 import { mapGetters } from 'vuex';
 import eventHub from '../event_hub';
@@ -14,6 +20,9 @@ export default {
     GlFormSelect,
     GlFormTextarea,
   },
+  directives: {
+    SafeHtml,
+  },
   props: {
     choices: {
       type: Array,
@@ -133,7 +142,7 @@ export default {
     :state="valid"
   >
     <template #description>
-      <span v-html="help"></span>
+      <span v-safe-html="help"></span>
     </template>

     <template v-if="isCheckbox">
```

Markus Koller @toupeira · 1 year ago Contributor

@justin_ho yes I can add that too and check if it's working, thanks! 👍

@leipert I'll go ahead and add this to %14.2, since the fix is small and we should have plenty of time until the security release after August 22.

Lukas 'Eipi' Eipert @leipert · 1 year ago Developer

@justin_ho Thanks for the patch!

@toupeira Feel free to tag me as a frontend reviewer on the MR! With that patch we should check that all the helps are rendered nicely still 😍

Dheeraj Joshi @djadmin · 1 year ago Developer

@toupeira please note that the XSS might not be reproducible anymore with !66391 (merged) changes. I think we would still need to backport the fix.

@justin_ho patch looks good to me, and there's a separate issue for that.

Markus Koller @toupeira · 1 year ago Contributor

@djadmin oh indeed, thanks! 👍

Just tested again and can't reproduce it anymore, the `href` gets HTML-escaped now:

I'll still submit an MR for `master` to hard-code the link since that seems like the cleanest solution, and will backport that to previous releases.

I had a look through the `help:` fields in all integration models and couldn't find any other instances where we inject user-submitted values, so I'll submit a public MR to switch to `v-safe-html` since I don't think we have any other XSS vulnerabilities with this, and that change will cause a bit more churn so it'll be trickier to backport 😉

Dheeraj Joshi @djadmin · 1 year ago — Developer

Sounds like a great plan 🙂

Please register or sign in to reply

Markus Koller changed weight to **1** 1 year ago

GitLab Bot added ~section ~dev scoped label 1 year ago

Markus Koller changed milestone to %14.2 1 year ago

Markus Koller assigned to @toupeira 1 year ago

GitLab SecurityBot removed ~security-set-milestone label 1 year ago

Markus Koller added ~Stretch label 1 year ago

Markus Koller added ~backend label 1 year ago

GitLab Bot @gitlab-bot · 1 year ago — Maintainer

👋, please can you add a type label to this issue to help with issue discovery in issue reports.

GitLab Bot added ~"auto updated" label 1 year ago

Dheeraj Joshi mentioned in issue #241874 (closed) 1 year ago

Arturo Herrero added ~group::integrations scoped label and automatically removed ~group::ecosystem [DEPRECATED] label 1 year ago

Arturo Herrero mentioned in issue #328389 (closed) 1 year ago

Markus Koller added ~workflow::in dev scoped label 1 year ago

Markus Koller added ~workflow::in review scoped label and automatically removed ~workflow::in dev label 1 year ago

Markus Koller mentioned in merge request !67890 (merged) 1 year ago

Markus Koller added ~workflow::verification scoped label and automatically removed ~workflow::in review label 1 year ago

GitLab SecurityBot added ~security-issue-escalated label 1 year ago

GitLab SecurityBot @gitlab-securitybot · 1 year ago — Author, Reporter

@mushakov @arturoherrero @mhenriksen This ~severity::2 ~security issue's milestone has expired.

About this automation: AppSec Escalation Engine

Andrew Kelly @ankelly · 1 year ago — Developer

This went out with the 14.2.2 security release.

Andrew Kelly closed 1 year ago

GitLab SecurityBot removed ~security-issue-escalated label 1 year ago

Arturo Herrero mentioned in issue gitlab-com/www-gitlab-com#12147 (closed) 1 year ago

GitLab Bot added 1 deleted label 1 year ago

GitLab Bot removed ~devops::create label 1 year ago

GitLab SecurityBot @gitlab-securitybot · 1 year 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.

Dominic Couture made the issue visible to everyone 1 year ago

GitLab Bot added ~devops::manage scoped label 3 weeks ago

GitLab Bot removed 1 deleted label 3 weeks ago

Please register or sign in to reply