

Vulnerability related fields are available to unauthorized users on GraphQL API

Why are we doing this work

The `Pipeline#securityReportFindings` and `Pipeline#securityReportSummary` fields are not restricted to access from unauthorized users! I checked the original MRs introducing these fields([I54104 \(merged\)](#)), and [I31550 \(merged\)](#)) to understand if this is a regression but seems like the permission checks were missing from the beginning.

Relevant links

- I've discovered this after the thread initiated by [@stanhu](#) and the question from [@adamcohen](#) [here](#).

Steps to reproduce

You can use the following curl command to verify that these sensitive fields are available to anyone!

If you run this command, add [your IP](#) and approximate timestamp to this table


Who?	IP Address	Timestamp (date -u)
@bwill	136.49.173.76	Wed Dec 8 20:17:57 UTC 2021 (probably about 15-20 mins before this)
@thiagocsf	59.102.81.249	Wed Dec 8 19:48:00 UTC 2021
@ngeorge1	122.181.40.178	Thu Dec 9 12:02:10 UTC 2021
@quintasan	31.178.237.73	Mon Dec 13 12:53:14 UTC 2021

```
curl 'https://gitlab.com/api/graphql' \  
-H 'authority: gitlab.com' \  
-H 'accept: application/json' \  
-H 'content-type: application/json' \  
--data-raw '{"query":"query {\n  project(fullPath: \"gitlab-org/gitlab\") {\n    id\n    pipeline(  
--compressed
```

Implementation plan

- ☐ [backend](#) Required permissions must be applied for these fields

Edited 11 months ago by [Michał Zając](#)

 Drag your designs here or [click to upload](#).

Tasks 


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

Linked items 


Blocks

 [\[BE\] Add fields to the PipelineSecurityReportFinding GraphQL type](#)



#335372


 14.8  3 

Related merge requests 

 [Add pipeline securityReportSummary to GraphQL](#)

I31550

 13.1  

 [Extend GraphQL Ci::PipelineType to include Security Report Findings](#)

!54104


 13.10  


 [Add fields to the PipelineSecurityReportFinding GraphQL type](#)

!75001



 14.8  

Activity

 **Mehmet Emin INAC** changed milestone to [%14.6](#) 11 months ago

 **Mehmet Emin INAC** added [workflow refinement](#) [group threat insights](#) [priority 1](#) [severity 1](#) [devops secure](#) [section sec](#) scoped labels 11 months ago

 **Mehmet Emin INAC** added [Category Vulnerability Management](#) [security](#) labels 11 months ago

  **GitLab Bot** added [type maintenance](#) scoped label 11 months ago



Mehmet Emin INAC @minac · 11 months ago

Author

Maintainer

[@thiagocsf](#) - This is a really important vulnerability that has to be addressed before anything if you agree.

/cc [@gitlab-org/secure/threat-insights-backend-team](#)



Thiago Figueiró @thiagocsf · 11 months ago

Developer

Absolutely, [@minac](#). This is a drop-everything-else for us. Thank you for following this up - I saw [@adamcohen](#)'s question in the MR and assumed he didn't need to do anything else because we had it covered.

Please [register](#) or [sign in](#) to reply



Stan Hu @stanhu · 11 months ago

Owner

[@minac](#) Should this block [!75001 \(merged\)](#) from being merged then?



Mehmet Emin INAC @minac · 11 months ago

Author

Maintainer

[@stanhu](#) - Yes, we shouldn't expose more information via those fields until we fix this issue. Also by any chance do you know a way to hide that MR from the public or at least the discussion until we release the security patch?



Stan Hu @stanhu · 11 months ago

Owner

I think only way would be to delete the notes or the merge request outright. I did see some recent HackerOne report where someone was able set a note to confidential, though.



Thiago Figueiró @thiagocsf · 11 months ago

Developer

[@adamcohen](#) [@minac](#) [@stanhu](#) given our timeline for releasing critical security vulnerabilities, I think we should delete [@adamcohen](#)'s MR until this is fixed.



Thiago Figueiró @thiagocsf · 11 months ago

Developer

Thinking about this a bit more, maybe deleting the comments related to security is enough? this is the thread hinting at the weakness: [!75001 \(comment 753853999\)](#).

Edited by [Thiago Figueiró](#) 11 months ago



Stan Hu @stanhu · 11 months ago

Owner

Did someone already delete the comments? It look like it to me.




Adam Cohen @adamcohen · 11 months ago

Developer

Did anyone back up these comments somewhere before deleting? There was some useful information in that thread.

Please [register](#) or [sign in](#) to reply

 [GitLab SecurityBot](#) changed due date to December 26, 2021 [11 months ago](#)

 [Thiago Figueiró](#) mentioned in merge request [!75001 \(merged\)](#) [11 months ago](#)



[Thiago Figueiró](#) [@thiagocsf](#) · [11 months ago](#)

Developer

[@ngeorge1](#), can I please enlist your assistance to check for exploits of this in the wild?

My first thought is to check production logs for unauthenticated GraphQL requests matching any of the types in the description.



[Philippe Lafoucrière](#) [@plafoucriere](#) · [11 months ago](#)

Maintainer

[@thiagocsf](#) did you ping SIRT on this already?





[Thiago Figueiró](#) [@thiagocsf](#) · [11 months ago](#)

Developer

[@plafoucriere](#) yes <https://gitlab.com/gitlab-com/gl-security/security-operations/sirt/operations/-/issues/1803>

Please [register](#) or [sign in](#) to reply

 [Brian Williams](#) changed the description [11 months ago](#) ·

 [Thiago Figueiró](#) changed the description [11 months ago](#) ·



[Philippe Lafoucrière](#) [@plafoucriere](#) · [11 months ago](#)

Maintainer

[@thiagocsf](#) Does "unauthorized" means the data is also available for private project without any authentication?



[Thiago Figueiró](#) [@thiagocsf](#) · [11 months ago](#)

Developer

[@plafoucriere](#), I believe private projects are not affected. Testing with `gitlab-org/security/gitlab` doesn't return any results.

```
08:05 $ curl 'https://gitlab.com/api/graphql' -H 'authority: gitlab.com' \
-H 'accept: application/json' -H 'content-type: application/json' \
--data-raw '{"query":"query {\n  project(fullPath: \"gitlab-org/security/gitlab\")
```

```
{
  "data": {
    "project": null
  }
}
```

Edited by [Thiago Figueiró](#) [11 months ago](#)



[Brian Williams](#) [@bwill](#) · [11 months ago](#)

Maintainer

[@plafoucriere](#) [@thiagocsf](#) Private projects do require that the user be authenticated and a member of the project, but the vulnerabilities are visible to access levels which normally do not have permission to view them (Guest & Reporter).



[Thiago Figueiró](#) [@thiagocsf](#) · [11 months ago](#)

Developer

Nice one, [@bwill](#). I added this to the [summary](#) as an impact .

Edited by [Thiago Figueiró](#) 11 months ago



Philippe Lafoucrière [@plafoucriere](#) · 11 months ago

Maintainer

Thank you [@bwill](#) and [@thiagocsf](#). This will greatly reduce the scope of this security bug, since anyone can run the same security scans manually and get the results anyway.



Nikhil George [@ngeorge1](#) · 11 months ago

Developer

As Philippe rightly pointed out above the requirement of Guest/Reporter access in private projects to exploit this vulnerability reduces the severity of the issue. As anyone who has access to a project can anyway copy the code and run the scans separately. Based on this [impact](#) it looks a [severity 3](#) (CVSS:3.0/AV:N/AC:L/PR:L/UI:N/S:U/C:L/I:N/A:N) issue instead of [severity 1](#) , adjusting severity accordingly.

cc: [@thiagocsf](#), [@gitlab-com/gl-security/appsec](#), [@plafoucriere](#)



Thiago Figueiró [@thiagocsf](#) · 11 months ago

Developer

[@ngeorge1](#), thank you. That means we don't need to force a release and can wait for the regular schedule.

I assume the same rationale applies to the anonymous access to public projects? i.e. if I have access to the source code, I can run the scanners.

That doesn't account for manually entered vulnerabilities and cluster image scans. The latter requires access to the K8S cluster(s), which the attacker won't have. The former depends where it originally came from.

Edited by [Thiago Figueiró](#) 11 months ago



Nikhil George [@ngeorge1](#) · 11 months ago

Developer

[@thiagocsf](#),

That means we don't need to force a release and can wait for the regular schedule.

Yes

I assume the same rationale applies to the anonymous access to public projects? i.e. if I have access to the source code, I can run the scanners.

Correct

That doesn't account for manually entered vulnerabilities and cluster image scans. The latter requires access to the K8S cluster(s), which the attacker won't have. The former depends where it originally came from.

Thank you for pointing this out, so in projects with custom scanners and cluster image scans, this can have a higher impact on confidentiality. Factoring in this for severity calculation the new CVSS could be CVSS:3.0/AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N which is 6.5 (Medium) so I don't think a severity update is needed.

cc: [@gitlab-com/gl-security/appsec](#)



Brian Williams [@bwill](#) · 11 months ago

Maintainer

[@ngeorge1](#) [@plafoucriere](#) FWIW, Guest members do not have permissions to view the code. (See: [Repository: View project code](#))

I do think that this access level is not very commonly used.

Edited by [Brian Williams](#) 11 months ago



Olivier Gonzalez [@gonzoyumo](#) · 11 months ago

Developer

[@ngeorge1](#) [@thiagocsf](#) [@plafoucriere](#) [@bwill](#) to follow on the idea of "users can run the scans on the project's source code anyway" we also give them access to the CI job artifacts. So even if we prevent access to these API endpoints, they can still get the raw reports. As gitlab is a public project, with [public pipelines enabled](#) I assume, the CI job artifacts are publicly available to anonymous users.

E.g. <https://gitlab.com/gitlab-org/gitlab/-/jobs/1872137012/artifacts/browse> can be seen even when not signed in.

It might be time to revisit this though and maybe we should look into having private artifacts? And what if the job logs output the results too? This is obviously out of scope but just throwing it there.



Philippe Lafoucrière @plafoucriere · 11 months ago

Maintainer

FWIW, Guest members do not have permissions to view the code. (See: [Repository: View project code](#))

Thanks for pointing that out @bwill, this is a very good point. @ngeorge1 I don't it would change the CVSS vector, as the user has to be a member (at least guest) of the project. Also, the Confidentiality remains low to me, WDYT?

So even if we prevent access to these API endpoints, they can still get the raw reports. As gitlab is a public project, with [public pipelines enabled](#) I assume, the CI job artifacts are publicly available to anonymous users.

Artifacts are only available [If Public pipelines is enabled in Project Settings > CI/CD](#), so it makes sense to me to keep this setting as it. Having access to the pipelines but not the artifacts could confuse users. But there's indeed a gap for Reporters who can download these artifacts, but can't access vulnerability reports.



Nikhil George @ngeorge1 · 11 months ago

Developer

I don't it would change the CVSS vector, as the user has to be a member (at least guest) of the project. Also, the Confidentiality remains low to me, WDYT?

My rationale was there is a complete loss of confidentiality w.r.t vulnerability info as all the project members in case of the private project could access the vulnerabilities. Even if the guest could not access the source code or when pipeline's access is set to private.

Please [register](#) or [sign in](#) to reply



Thiago Figueiró added [workflow](#) [in dev](#) [type](#) [bug](#) scoped labels and automatically removed [type](#) [maintenance](#) [workflow](#) [refinement](#) labels 11 months ago



Thiago Figueiró assigned to [@minac](#), [@subashis](#), and [@jschafer](#) 11 months ago



GitLab Bot added [Accepting merge requests](#) label 11 months ago



GitLab SecurityBot @gitlab-securitybot · 11 months ago

Reporter

[@matt wilson](#) [@thiagocsf](#) [@ngeorge1](#) This issue is ready for triage as per [HackerOne process](#).

About this automation: [AppSec Escalation Engine](#)



Thiago Figueiró @thiagocsf · 11 months ago

Developer

But this didn't come from H1 🤔



Nikhil George @ngeorge1 · 11 months ago

Developer

Good catch, thanks for pointing out. I will create an issue for this.

Please [register](#) or [sign in](#) to reply



GitLab Bot removed [Accepting merge requests](#) label 11 months ago




Thiago Figueiró mentioned in issue [#335372 \(closed\)](#) 11 months ago




Thiago Figueiró marked this issue as related to [#335372 \(closed\)](#) 11 months ago



Nikhil George added [severity](#) [3](#) scoped label and automatically removed [severity](#) [1](#) label 11 months ago

 **Nikhil George** changed the description 11 months ago ·

 **Michał Zając** changed the description 11 months ago ·

 **GitLab SecurityBot** added `security-issue-escalated` label 10 months ago



GitLab SecurityBot @gitlab-securitybot · 10 months ago

Reporter

@matt wilson @thiagocsf @ngeorge1 This severity 3 security issue's milestone has expired.

About this automation: [AppSec Escalation Engine](#)



Thiago Figueiró @thiagocsf · 10 months ago

Developer

@ngeorge1, is there anything else for group threat insights to do here? @pshutsin has assigned the MRs to the bot (example).



Nikhil George @ngeorge1 · 10 months ago

Developer

@thiagocsf, I do not see any pending tasks from group threat insights. Main MR is approved by maintainer and appsec, milestones are set, backports in place and approved, all pipelines green, looks all good to me.

Edited by Nikhil George 10 months ago



Thiago Figueiró @thiagocsf · 10 months ago


Developer

Excellent! thanks, @ngeorge1. I see the security tracking release issue <https://gitlab.com/gitlab-org/gitlab/-/issues/347281> is due 10/Jan, so I'll check back here after that.

FYI @subashis @minac

Edited by Thiago Figueiró 10 months ago

Please [register](#) or [sign in](#) to reply

 **Mehmet Emin INAC** changed milestone to %14.7 10 months ago

 **Mehmet Emin INAC** added `workflow verification` scoped label and automatically removed `workflow in dev` label 10 months ago



Vitor Meireles De Sousa @vdesousa · 10 months ago

Developer

This is fixed in 14.6.2. Closing.



Subashis Chakraborty @subashis · 10 months ago

Developer

Thanks Everyone.

Please [register](#) or [sign in](#) to reply

 **Vitor Meireles De Sousa** closed 10 months ago

 **GitLab SecurityBot** removed `security-issue-escalated` label 10 months ago

 **Thiago Figueiró** made the issue visible to everyone 10 months ago



Thiago Figueiró @thiagocsf · 10 months ago

Developer

Making this public so we can reference it in the release post for %14.7.

Please [register](#) or [sign in](#) to reply

