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(!54104 (merged), and !31550 (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 <u>@stanhu</u> and the question from <u>@adamcohen</u> <u>here</u>.

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

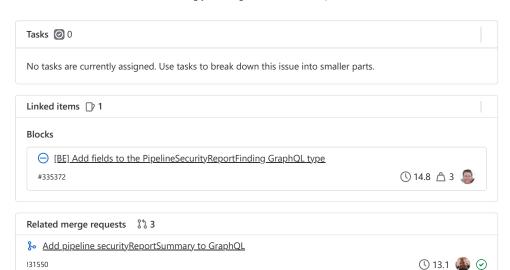
Who?	IP Address	Timestamp (date -u)
@bwill	136.49.173.76	Wed Dec 8 20:17:57 UTC 2021 (probably about 15-20 mins before this)
@thiagocsf	59.102.81.249	Wed Dec 8 19:48:00 UTC 2021
@ngeorge1	122.181.40.178	Thu Dec 9 12:02:10 UTC 2021
@quintasan	31.178.237.73	Mon Dec 13 12:53:14 UTC 2021

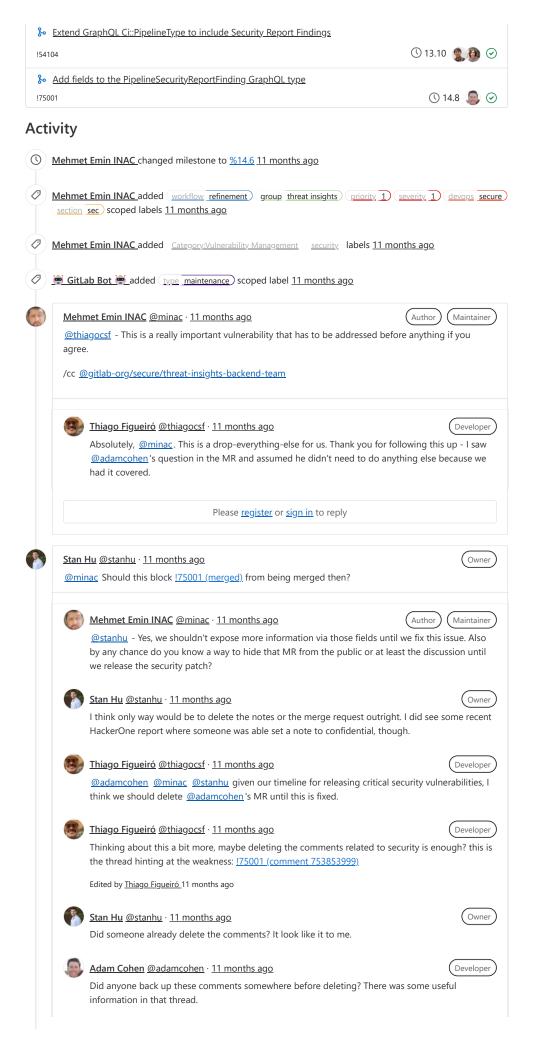
Implementation plan

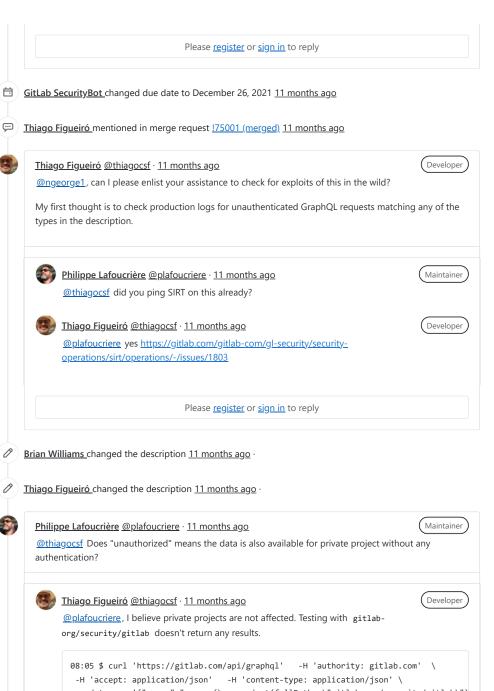
backend Required permissions must be applied for these fields

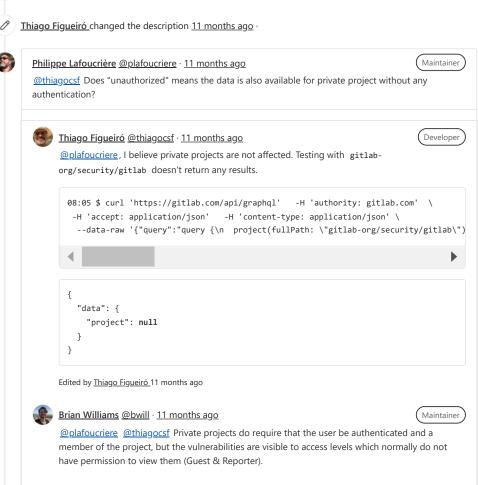
Edited 11 months ago by Michał Zając

 ⚠ Drag your designs here or <u>click to upload</u>.









Developer

Thiago Figueiró @thiagocsf · 11 months ago

Nice one, <a>@bwill. I added this to the <a>summary as an impact .



Philippe Lafoucrière @plafoucriere · 11 months ago

Maintainer

Thank you <u>@bwill</u> and <u>@thiagocsf</u>. 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 (CVSS:3.0/AV:N/AC:L/PR:L/UI:N/S:U/C:L/I:N/A:N) issue instead of severity adjusting severity accordingly.

cc: @thiagocsf, @gitlab-com/gl-security/appsec, @plafoucriere



Thiago Figueiró @thiagocsf · 11 months ago

Developer

<u>@ngeorge1</u>, 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

<u>@ngeorge1</u> <u>@plafoucriere</u> FWIW, Guest members do not have permissions to view the code. (See: <u>Repository; View project code</u>)

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 <a>@bwill, this is a very good point. <a>@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 <u>public pipelines enabled</u> I assume, the CI job artifacts are publicly available to anonymous users. Artifacts are only available $\underline{\text{If Public pipelines is enabled in Project Settings}} > \underline{\text{CI/CD.}}$, 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 7 <u>Thiago Figueiró</u> added <u>workflow in dev</u> <u>type</u> <u>bug</u> scoped labels and automatically removed type maintenance workflow refinement labels 11 months ago 🛆) <u>Thiago Figueiró</u> assigned to <u>@minac</u>, <u>@subashis</u>, and <u>@jschafer</u> <u>11 months ago</u> 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

