Closed · Issue created 1 year ago by GitLab SecurityBot

GraphQL mutations allow Merge Request creator to bypass locked status

HackerOne report #1063420 by `jimeno` on 2020-12-21, assigned to @dcouture:

Report | Attachments | How To Reproduce

Report

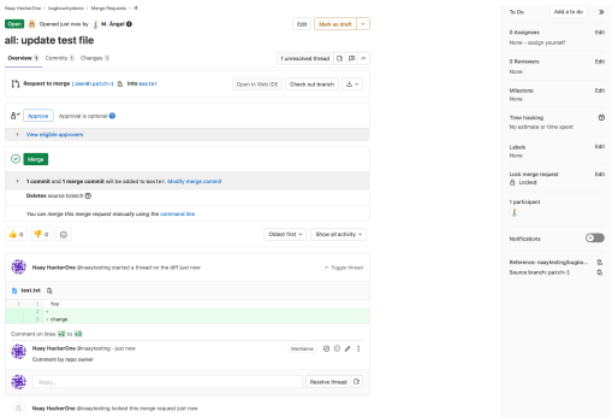
Summary

Hi GitLab,

I'd like to report an Improper Access Control issue where attacker is able to perform state changing actions such as (un-)resolving discussion threads in Merge Requests after the repository owner locks the MR. The mutation used in this example is `discussionToggleResolve`.

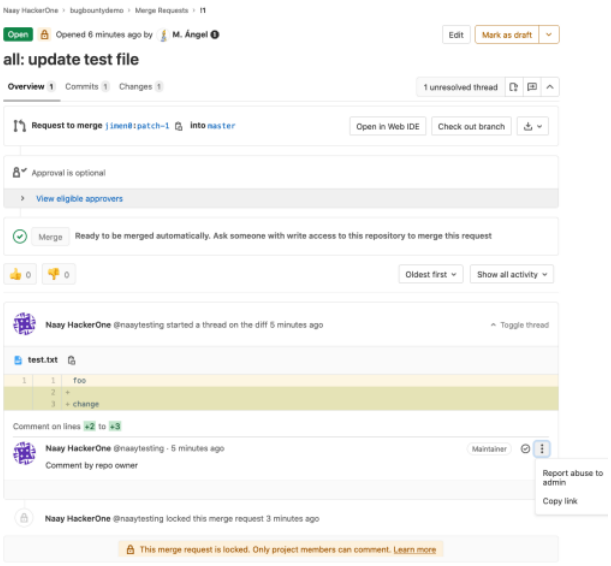
Scenario setup

1. Victim: create a public repository and add a dummy file to it
2. Attacker: fork the project and add a change to it
3. Attacker: submit a Merge Request
4. Any: add a comment so a discussion thread is generated
5. Victim: lock the Merge Request (notice the attacker isn't able to (un-)resolve the existing discussions anymore)



Steps to reproduce

1. Attacker: notice you aren't able to post new comments or resolve the discussion thread via the user interface



2. Attacker: run the following GraphQL query to find the ID of the discussion thread. Please, replace the repo and owner where needed

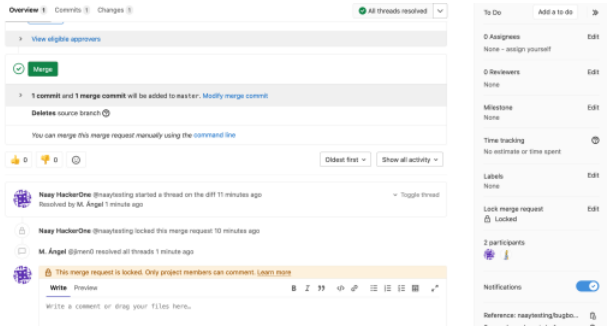
```
{
  project(fullPath: &#34;naaytesting/bugbountydemo&#34;) {
    mergeRequests {
      edges {
        node {
          id
          title
          discussions {
            edges {
              node {
                id
                resolvable
                notes {
                  edges {
                    node {
                      body
                    }
                  }
                }
              }
            }
          }
        }
      }
    }
  }
}
```

3. Attacker: note the discussion ID. In my case it is `gid://gitlab/DiffDiscussion/da18f8200f0934b43fd1d1f62d5ea1a8dec2e1a8`. Notice its `resolvable` attribute is set to `true` while the UI shows the MR is completely locked and doesn't allow to (un-)resolve it.

4. Run the following GraphQL query replacing the ID with yours.

```
mutation test {
  discussionToggleResolve(input: {id: &#34;gid://gitlab/DiffDiscussion/da18f8200f0934b43fd1d1f62d5ea1a8dec2e1a8&#34;,
    errors
  }
}
```

5. Victim: refresh the MR and notice the attacker was able to perform activities against it and the discussion is resolved now.



Impact

Attacker is able to do state-changing actions against a locked MR. In this particular case, (un-)resolve discussion threads.

Examples

GitLab.com users can perform state-changing actions in their Merge Requests after the repository owners have locked them.

What is the current *bug* behavior?

Attacker is able to do state-changing actions against a locked MR. In this particular case, (un-)resolve discussion threads.

What is the expected *correct* behavior?

Attacker isn't able to do state-changing actions after the repo owner locks their MR.

Relevant logs and/or screenshots

Please, refer to each section.

Output of checks

This bug happens on GitLab.com

Results of GitLab environment info

N/A

Impact

Attacker is able to do state-changing actions against a locked MR. In this particular case, (un-)resolve discussion threads.

If this is the expected behavior, please let me know and I'll be happy to self-close this report as N/A.

Attachments

Warning: Attachments received through HackerOne, please exercise caution!

- attacker_opens_locked_mr.png
- attacker_activity_in_pr_after_locking.png
- setup_final_status.png

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.

Activity

GitLab SecurityBot changed due date to March 22, 2021 1 year ago

GitLab SecurityBot added HackerOne security labels 1 year ago

GitLab SecurityBot added Weakness CWE-284 priority 3 severity 3 scoped labels 1 year ago

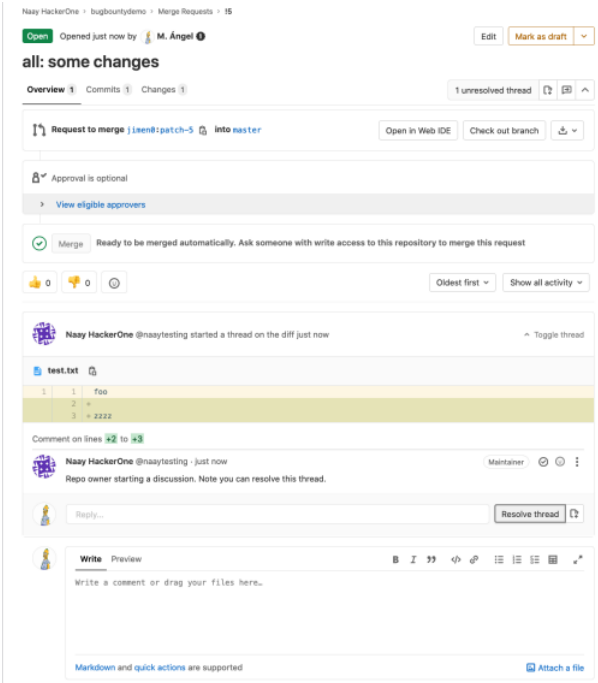
GitLab SecurityBot @gitlab-securitybot · 1 year ago Author Reporter

HackerOne comment by jimeno :

Some minor notes I forgot:

- Official documentation on how to start a discussion thread: https://docs.gitlab.com/ee/user/discussions/index.html#threaded-discussions
- Official documentation on what a locked discussion is and how to lock/unlock it: https://docs.gitlab.com/ee/user/discussions/index.html#lock-discussions

Here is a screenshot of the status of the MR viewed from the attacker (jimen0) profile **before** the repo owner (victim, naaytesting) locks the MR. As you can see, it's possible to resolve the discussion thread. It isn't possible to do it after the owner locks the MR.



Policy/Scope: checked Impact: checked Reproducible: checked

Please, let me know if anything else is needed from my side.

Attachments

Warning: Attachments received through HackerOne, please exercise caution!

- attacker_status_before_owner_locks_mr.png

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

HackerOne comment by dcouture :

Hi [@]jimeno,

Thank you for your report, it's well written and simple to reproduce!

I could reproduce the issue you describe however I wonder if the bug actually is that the author of the MR should retain their write access on a locked MR even if they're not a member of the project. I'm going to discuss with the product team and update you as soon as possible. However do note that many people are taking time off during this holiday season so there might be a small delay in my reply.

Best regards, Dominic GitLab Security Team

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

Dominic Couture @dcouture · 1 year ago — Developer

Hello @phikai @danielgruesso ! I'm not completely sure which one of your groups is better suited for this.

We received a report on HackerOne for the following behavior:

- Assume a non-project-member opens an MR in a public project from a fork
- The MR is then locked for whatever reason
- The author of the MR, who is not a member and should be locked out, can still resolve discussions through the GraphQL API and also apply any suggestions through the UI (the Apply Suggestion button is still active)

Before accepting that report as a vulnerability, I was wondering if the bug wasn't actually on the other way around? Should an MR author retain their write access to a locked MR even if it's locked? I could see it go both ways and I wanted your input on that.

Either we lock down those few actions that are still possible for the non-member author, or we open up everything and consider the author as a project member for locked MRs.

Edited by Dominic Couture 1 year ago

Kai Armstrong @phikai · 1 year ago — Developer

Thoughts @m_gill?

Michelle Gill @m_gill · 1 year ago — Developer

@phikai my personal *opinion* is that we should enforce permissions in the GraphQL API just the same as they are today in the UI and that this a vulnerability. It's not a strong opinion. Also I'm relabeling this to Code Review!

Kai Armstrong @phikai · 1 year ago — Developer

@m_gill Thanks - I think that makes sense. Although I'm not sure this is a vulnerability and mostly just a bug given the impact.

Dominic Couture @dcouture · 1 year ago — Developer

Certainly a very low severity one, but as a permission security ~bug we can use that word. :) Thanks for your feedback!

Please register or sign in to reply

Dominic Couture added priority 4 severity 4 group source code devops create scoped labels and automatically removed priority 3 severity 3 labels 1 year ago

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

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

Setting label(s) Category:Source Code Management section dev based on group source code.

GitLab Bot added Category:Source Code Management label 1 year ago

GitLab Bot added section dev scoped label 1 year ago

Michelle Gill added group code review scoped label and automatically removed group source code label 1 year ago

Michelle Gill changed milestone to %13.10 1 year ago

GitLab Bot added Accepting merge requests label 1 year ago

Kai Armstrong changed milestone to %13.11 1 year ago

Michelle Gill changed milestone to %13.10 1 year ago

Michelle Gill mentioned in issue create-stage#12791 (closed) 1 year ago

Michelle Gill added backend label 1 year ago

Michelle Gill changed weight to 3 1 year ago

Michelle Gill added workflow ready for development scoped label 1 year ago

GitLab Bot mentioned in issue gitlab-org/quality/triage-reports#1993 (closed) 1 year ago

Michelle Gill added Deliverable label 1 year ago

Michelle Gill mentioned in issue gitlab-com/www-gitlab-com#10501 (closed) 1 year ago

GitLab Bot mentioned in issue gitlab-org/quality/triage-reports#2151 (closed) 1 year ago

Tomislav Nikić added quad-planning complete-action scoped label 1 year ago

Tomislav Nikić @tomi · 1 year ago Contributor

This could actually be part of an E2E test in combination with something else. No need for an explicit test though 🤔

Michelle Gill @m_gill · 1 year ago Developer

This is no longer a Deliverable due to a P1/S1 we received https://gitlab.com/gitlab-org/gitlab/-/issues/323198

Michelle Gill removed Deliverable label 1 year ago

Michelle Gill changed milestone to %13.11 1 year ago

Michelle Gill changed milestone to %13.12 1 year ago

Michelle Gill changed milestone to %14.0 1 year ago

Matt Nohr mentioned in issue create-stage#12823 (closed) 1 year ago

Kai Armstrong changed milestone to %14.1 1 year ago

Matt Nohr mentioned in issue create-stage#12865 (closed) 1 year ago

Kai Armstrong changed milestone to %14.2 1 year ago

Matt Nohr mentioned in issue create-stage#12871 (closed) 1 year ago

Kai Armstrong changed milestone to %14.3 1 year ago

Matt Nohr @mnohr · 1 year ago Developer

We will not have the capacity for this in %14.3

Matt Nohr @mnohr · 1 year ago Developer

Moving this to %14.5 as we have higher priority security issues planned for %14.4

Please register or sign in to reply

Matt Nohr changed milestone to %14.4 1 year ago

Matt Nohr changed milestone to %14.5 1 year ago

Matt Nohr mentioned in issue create-stage#12912 (closed) 1 year ago

Matt Nohr changed due date to April 21, 2021 1 year ago

Patrick Bajao assigned to @patrickbajao 1 year ago

Patrick Bajao added workflow in dev scoped label and automatically removed workflow ready for development label 1 year ago

GitLab Bot removed Accepting merge requests label 1 year ago

Nikhil George @ngeorge1 · 1 year ago Developer

Closing this issue as it is fixed in 14.4.1 security release CVE-ID: CVE-2021-39904

cc: @dcouture

Nikhil George closed 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.

Nikhil George @ngeorge1 · 1 year ago Developer

Switching issue to public

Please register or sign in to reply

Nikhil George made the issue visible to everyone 1 year ago

Please register or sign in to reply