

Closed

RCA: 2020-02-11 High db insert rate caused by enabling issue templates on projects

RCA: 2020-02-11 High db insert rate caused by enabling issue templates on projects on projects

Incident: [production#1651 \(closed\)](#).

Summary

Enabling instance templates for each project creation due to a bug, caused thousands of services to be created for each project creation (doing URL validation for each service, causing delays) which was causing a high db insert rate and latencies for web.

Additionally, because of copying the same template to the projects, many slack notifications from other projects were sent to out to one single project, which is a security issue: <https://gitlab.com/gitlab-com/gl-security/secops/operations/issues/650>

- Service(s) affected : Service Web Service Postgres
- Team attribution : group ecosystem
- Minutes downtime or degradation : 153m (06:31 - 09:04)

For calculating duration of event, use the [Platform Metrics Dashboard](#) to look at appdex and SLO violations.

Impact & Metrics

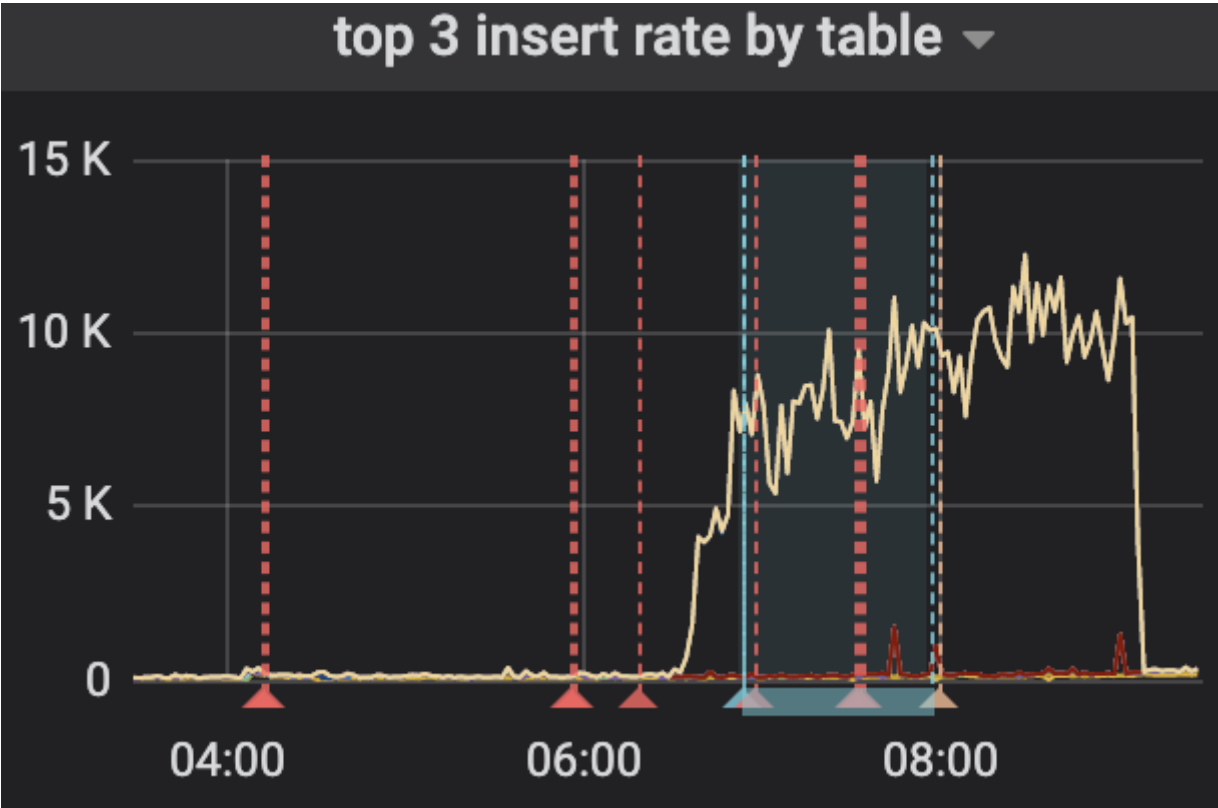
Start with the following:

- What was the impact of the incident? (i.e. service outage, sub-service brown-out, exposure of sensitive data, ...)
- Who was impacted by this incident? (i.e. external customers, internal customers, specific teams, ...)
- How did the incident impact customers? (i.e. preventing them from doing X, incorrect display of Y, ...)
- How many attempts were made to access the impacted service/feature?
- How many customers were affected?
- How many customers tried to access the impacted service/feature?

Include any additional metrics that are of relevance.

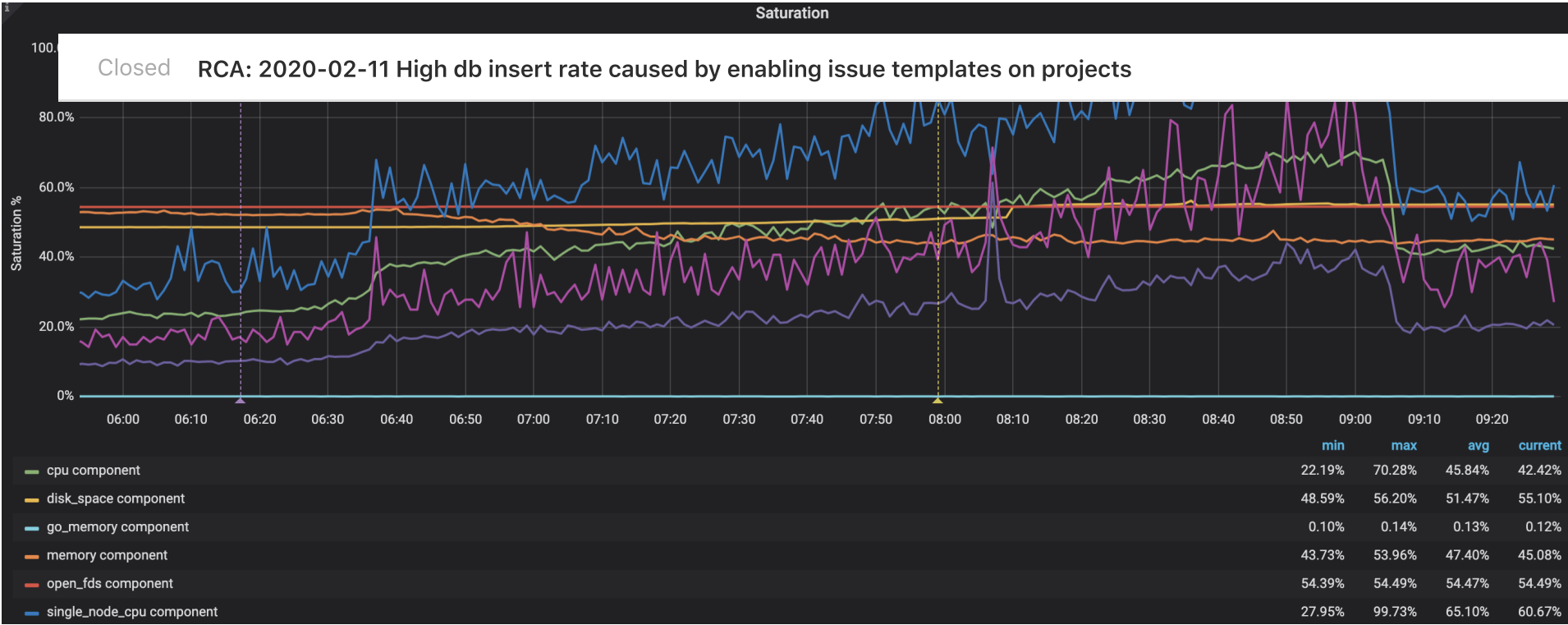
Provide any relevant graphs that could help understand the impact of the incident and its dynamics.

DB inserts into services table:



<https://dashboards.gitlab.net/d/RZmbBr7mk/gitlab-triage?orgId=1&fullscreen&panelId=8&from=1581399974013&to=1581413232942>

Web saturation:



<https://dashboards.gitlab.net/d/general-service/general-service-platform-metrics?orgId=1&fullscreen&panelId=10&from=1581400421276&to=1581413397765>

Detection & Response

Start with the following:

- How was the incident detected?
 - EOC was paged because of db replication lag caused by high insert rate, marquee customer alerts also triggered
- Did alarming work as expected?
 - We did not get SLO violation alerts, maybe the impact wasn't high enough - need to investigate
- How long did it take from the start of the incident to its detection?
 - 45m until [db replication alert](#) fired (06:31 - 07:16)
- How long did it take from detection to remediation?
 - 108m (07:16 - 09:04)
- Were there any issues with the response to the incident? (i.e. bastion host used to access the service was not available, relevant team member wasn't page-able, ...)

Root Cause Analysis

We experienced high web latencies

- Why? - Because we suddenly started to create many services for each project creation, doing url lookups each service in `lib/gitlab/url_blocker.rb:111` , which took a lot of time (during a db transaction inserting into the `services` table).
- Why? - Because the instance creation attribute was set to `true` for each new project.
- Why? - Because a [db migration](#) with a column rename created a trigger to set `instance = template` , which enabled the previously disabled attribute
- Why? - Because we had two versions of the codebase running at the same time, one would set a `template` to false when creating a new service, and the other would set `instance` to false but leave `template` as true (and therefore repopulate `instance` as true based on the trigger)
- Why? - Service code is old and does some things we probably would not accept now like not having unique constraints on `project_id` and service type. We have validations at application level, not at the database level.

What went well

Start with the following:

- Identify the things that worked well or as expected.
- Any additional call-outs for what went particularly well.

What can be improved

Start with the following:

- Using the root cause analysis, explain what can be improved to prevent this from happening again.
- Is there anything that could have been done to improve the detection or time to detection?
- Is there anything that could have been done to improve the response or time to response?
- Is there an existing issue that would have either prevented this incident or reduced the impact?
- Did we have any indication or beforehand knowledge that this incident might take place?

Corrective actions

- Immediate corrective actions were to revert the change.
 - [gitlab-org/gitlab!24857 \(merged\)](#).
- Closed **RCA: 2020-02-11 High db insert rate caused by enabling issue templates on projects**
- [View incident summary on PagerDuty](#).
- Add suggestion to [renaming columns docs](#) about splitting migration MRs

Guidelines

- [Blameless RCA Guideline](#)
- [5 whys](#)

Edited 4 months ago by [Henri Philipps](#)

Linked issues ⓘ

2

Relates to

⊖

2020-02-11: High insert rate for services table causing higher web latencies

production#1651

○

Run existing performance tests per reference architectures on staging to catch challenging performance regressions

gitlab-com/www-gitlab-... #6556

| Related merge requests ⓘ 11 | |
|---|---|
| <div><div><div></div><div>Service model: Rename template attribute to instance</div></div><div>gitlab-org/gitlab!23595</div></div> | <div><div>🕒 12.8</div><div><div></div><div></div></div><div>⚠️</div></div> |
| <div><div><div></div><div>Backfill LfsObjectsProject records of forks</div></div><div>gitlab-org/gitlab!24767</div></div> | <div><div>🕒 12.8</div><div><div></div><div></div></div><div>⚠️</div></div> |
| <div><div><div></div><div>Revert "Rename services template to instance migration"</div></div><div>gitlab-org/gitlab!24857</div></div> | <div><div>🕒 12.8</div><div><div></div><div></div></div><div>⚠️</div></div> |
| <div><div><div></div><div>Remove column instance on services table</div></div><div>gitlab-org/gitlab!24885</div></div> | <div><div>🕒 12.8</div><div><div></div><div></div></div><div>❌</div></div> |
| <div><div><div></div><div>Re-add template column to services table</div></div><div>gitlab-org/gitlab!24949</div></div> | <div><div>🕒 12.8</div><div><div></div><div></div></div><div>⚠️</div></div> |
| <div><div><div></div><div>Add note to column rename doc</div></div><div>gitlab-org/gitlab!25110</div></div> | <div><div>🕒 12.9</div><div><div></div><div></div></div><div>✅</div></div> |
| <div><div><div></div><div>Concentrates whitelisted and blacklisted tables</div></div><div>gitlab-org/gitlab!25624</div></div> | <div><div>🕒 12.10</div><div><div></div><div></div></div><div>⚠️</div></div> |
| <div><div><div></div><div>Add 'instance' column to services table</div></div><div>gitlab-org/gitlab!25714</div></div> | <div><div>🕒 12.9</div><div><div></div><div></div></div><div>⚠️</div></div> |
| <div><div><div></div><div>Validates only one service template per type</div></div><div>gitlab-org/gitlab!26380</div></div> | <div><div>🕒 12.9</div><div><div></div><div></div></div><div>⚠️</div></div> |
| <div><div><div></div><div>Alert Management alerts Slack Notifications</div></div><div>gitlab-org/gitlab!33017</div></div> | <div><div>🕒 13.1</div><div><div></div><div></div></div><div>⚠️</div></div> |
| <div><div><div></div><div>Link the RCA issue properly since it's public now</div></div><div>gitlab-org/gitlab!35986</div></div> | <div><div>🕒 13.2</div><div><div></div><div></div></div><div>✅</div></div> |

Henri Philipps

@hphilipps

added C1 IncidentReview S1 incident labels 4 months ago

Henri Philipps

@hphilipps

changed weight to 1 4 months ago

Henri Philipps

@hphilipps

made the issue confidential 4 months ago

Henri Philipps

@hphilipps

mentioned in issue [production#1651 \(closed\)](#), 4 months ago

Henri Philipps

@hphilipps

marked this issue as related to [production#1651 \(closed\)](#), 4 months ago


Henri Philipps

@hphilipps

added security label 4 months ago

https://gitlab.com/gitlab-com/gl-infra/infrastructure/-/issues/9176

3/12



Henri Philipps @hphilipps changed the description 4 months ago · [Compare with previous version](#) ▾



Closed

RCA: 2020-02-11 High db insert rate caused by enabling issue templates on projects

So we still don't know how we got the first template, but I think we know how the templates were increasing exponentially.

When creating projects and copying services from templates we do:

```
def self.build_from_instance(project_id, instance_level_service)
  service = instance_level_service.dup

  if instance_level_service.supports_data_fields?
    data_fields = instance_level_service.data_fields.dup
    data_fields.service = service
  end

  service.instance = false
  service.project_id = project_id
  service.active = false if service.active? && !service.valid?
  service
end
```

So here we set `instance = false` but since we have the trigger from the column rename to set `instance = template`, this gets set to `true` again.

We can see this happening here:

```
[7] pry(main)> s = SlackService.last.dup
SlackService Load (3.2ms)  SELECT "services".* FROM "services" WHERE "services"."type" = 'SlackService' OR
=> #<SlackService:0x00005569a9f5bf50
  id: nil,
  type: "SlackService",
  title: nil,
  project_id: nil,
  created_at: nil,
  updated_at: nil,
  active: false,
  properties: {"notify_only_broken_pipelines"=>true, "branches_to_be_notified"=>"default"},
  push_events: true,
  issues_events: true,
  merge_requests_events: true,
  tag_push_events: true,
  note_events: true,
  category: "chat",
  default: false,
  wiki_page_events: true,
  pipeline_events: true,
  confidential_issues_events: true,
  commit_events: true,
  job_events: true,
  confidential_note_events: true,
  deployment_events: false,
  description: nil,
  comment_on_event_enabled: true,
  instance: true,
  template: true>
[8] pry(main)> s.instance = false
=> false
[9] pry(main)> s.project_id = Project.first.id
[13] pry(main)> s.save
(0.4ms)  BEGIN
SlackService Create (14.4ms)  INSERT INTO "services" ("type", "project_id", "created_at", "updated_at", "pr
(2.2ms)  COMMIT
Project Load (1.3ms)  SELECT "projects".* FROM "projects" WHERE "projects"."id" = 1 LIMIT 1
GeoNode Exists? (0.9ms)  SELECT 1 AS one FROM "geo_nodes" LIMIT 1
Service Exists? (1.0ms)  SELECT 1 AS one FROM "services" WHERE "services"."project_id" = 1 AND "services"."
Project Update (4.7ms)  UPDATE "projects" SET "has_external_wiki" = FALSE WHERE "projects"."id" = 1
Service Exists? (1.7ms)  SELECT 1 AS one FROM "services" WHERE "services"."project_id" = 1 AND "services"."
Project Update (3.2ms)  UPDATE "projects" SET "has_external_issue_tracker" = FALSE WHERE "projects"."id" =
=> true
[14] pry(main)> s.reload
SlackService Load (1.8ms)  SELECT "services".* FROM "services" WHERE "services"."type" = 'SlackService' AND
=> #<SlackService:0x00005569a9f5bf50
  id: 485,
```

```
type: "SlackService",
title: nil,
```

Closed RCA: 2020-02-11 High db insert rate caused by enabling issue templates on projects

```
updated_at: Tue, 11 Feb 2020 10:34:59 UTC +00:00,
active: false,
properties: {"notify_only_broken_pipelines"=>true, "branches_to_be_notified"=>"default"},
push_events: true,
issues_events: true,
merge_requests_events: true,
tag_push_events: true,
note_events: true,
category: "chat",
default: false,
wiki_page_events: true,
pipeline_events: true,
confidential_issues_events: true,
commit_events: true,
job_events: true,
confidential_note_events: true,
deployment_events: false,
description: nil,
comment_on_event_enabled: true,
instance: true,
template: true>
```

▼ Collapse replies



Heinrich Lee Yu @engwan · 4 months ago

Developer

So we still don't know how we got the first template, but I think we know how the templates were increasing exponentially.

When creating projects and copying services from templates we do:

As discussed on the call, this is actually the root cause because when creating new services, we use the same method and copy from the instance template.

This is the relevant code in `app/models/project.rb` :

```
def find_or_initialize_service(name)
  return if disabled_services.include?(name)

  service = find_service(services, name)
  return service if service

  # We should check if an instance-level service exists
  instance_level_service = find_service(instance_level_services, name)

  if instance_level_service
    Service.build_from_instance(id, instance_level_service)
  else
    # If no instance-level service exists, we should create a new service. Ex `build_gitlab_ci_service`
    public_send("build_#{name}_service") # rubocop:disable GitlabSecurity/PublicSend
  end
end
```

And we have a single template that existed way before (which is normal).

Edited by Heinrich Lee Yu 4 months ago



Stan Hu @stanhu · 4 months ago

Maintainer

In our delayed replica, we had this inactive `SlackService` :

```
gitlabhq_production=# select * from services where template = 't' AND type = 'SlackService';
-[ RECORD 1 ]-----+-----
id              | 40371579
type            | SlackService
title           |
project_id      |
created_at      | 2017-07-21 18:48:14.542172+00
updated_at      | 2017-07-21 18:48:14.542172+00
active          | f
```



| | | |
|------------|--|--|
| properties | | <code>{"notify_only_broken_pipelines":true,"notify_only_default_branch":true}</code> |
| template | | t |

Closed **RCA: 2020-02-11 High db insert rate caused by enabling issue templates on projects**

| | | |
|----------------------------|--|------|
| merge_requests_events | | t |
| tag_push_events | | t |
| note_events | | t |
| category | | chat |
| default | | f |
| wiki_page_events | | t |
| pipeline_events | | t |
| confidential_issues_events | | t |
| commit_events | | t |
| job_events | | t |
| confidential_note_events | | t |
| deployment_events | | f |
| description | | |
| comment_on_event_enabled | | t |
| instance | | t |

```
gitlabhq_production=# select count(*) from services where template = 't' and active = 'f';
-[ RECORD 1 ]
count | 35


gitlabhq_production=# select count(*) from services where template = 't' and active = 't';
-[ RECORD 1 ]
count | 0
```

**Stan Hu** @stanhu · 4 months ago

Maintainer

As discussed in the call, @Andysoiron is going to:


1. Revert [gitlab-org/gitlab!23595 \(diffs\)](#)
2. Add a new migration to drop the trigger and the new instance column

**Stan Hu** @stanhu · 4 months ago

Maintainer

Ideas to avoid this in the future:

1. Add database constraints to ensure (project ID, type) is unique (e.g. disallow duplicate SlackService items)
2. Add database constraints to ensure (template, type) is unique
3. Don't copy instance templates if they aren't active in <https://gitlab.com/gitlab-org/gitlab/blob/4fb5926f3bb143834c72e899e8ae4c7009e18db8/app/models/project.rb#L1226>
4. Consider putting instance templates in a different table (instead of relying on a boolean column). Add database constraints to ensure only type values aren't duplicated.

**Sean McGivern** @smcgivern · 4 months ago

Maintainer

Great find @engwan!

@stanhu I think that your first two items would indeed cover this:

1. Add database constraints to ensure (project ID, type) is unique (e.g. disallow duplicate SlackService items)

2. Add database constraints to ensure (template, type) is unique

If we also added a disabled template for each service on GitLab.com. We already had a bunch from 2017 (I'm not sure why), but we appear to have made those non-templates even though they have no project ID (which should also be invalid!). This means we'd want to refine those a little:

1. project_id, type is unique where template = 'f'.
2. project_id is null where template = 't'.
3. type is unique where template = 't'.

(The last two can't be combined, I don't think, because null isn't equal to null.)

These were the former templates that were inactive anyway:

```
gitlabhq_production=> SELECT created_at, type FROM services WHERE project_id IS NULL AND active != 't';
      created_at      |      type
-----+-----
2019-12-13 19:13:30.566086+00 | AlertsService
```

2017-07-21 18:48:13.972186+00 | AsanaService

2017-07-21 18:48:13.991688+00 | AssemblaService

Closed

RCA: 2020-02-11 High db insert rate caused by enabling issue templates on projects

2017-07-21 18:48:14.035665+00 | BuildkiteService

2017-07-21 18:48:14.053291+00 | CampfireService

2017-07-21 18:48:14.071443+00 | CustomIssueTrackerService

2017-07-21 18:48:14.093299+00 | DroneCiService

2017-07-21 18:48:14.111552+00 | EmailsOnPushService

2017-07-21 18:48:14.129146+00 | ExternalWikiService

2017-07-21 18:48:14.149001+00 | FlowdockService

2017-07-21 18:48:14.192648+00 | GitlabSlackApplicationService

2017-07-21 18:48:14.222023+00 | IrkerService

2017-07-21 18:48:14.241304+00 | JenkinsService

2017-07-21 18:48:14.259266+00 | JenkinsDeprecatedService

2017-07-21 18:48:14.270927+00 | JiraService

2017-07-21 18:48:14.31222+00 | MattermostService

2017-07-21 18:48:14.330182+00 | MattermostSlashCommandsService

2017-07-21 18:48:14.426645+00 | MicrosoftTeamsService

2017-07-21 18:48:14.446769+00 | PipelinesEmailService

2017-07-21 18:48:14.482695+00 | PrometheusService

2017-07-21 18:48:14.505058+00 | PushoverService

2017-07-21 18:48:14.523186+00 | RedmineService

2017-07-21 18:48:14.542172+00 | SlackService

2017-07-21 18:48:14.552865+00 | SlackSlashCommandsService

2017-07-21 18:48:14.578765+00 | TeamcityService

2017-12-12 14:41:35.346956+00 | PackagistService

2019-04-30 21:11:13.663384+00 | DiscordService

2019-04-30 21:11:13.694105+00 | HangoutsChatService

2019-04-30 21:11:13.725864+00 | HipchatService

2019-04-30 21:11:13.827216+00 | YoutrackService


2017-07-21 18:48:14.464925+00 | PivotaltrackerService

2019-12-13 19:13:30.608975+00 | GithubService

2019-12-13 19:13:30.679222+00 | UnifyCircuitService

(35 rows)

Note that there are currently no duplicates there. We should probably set those back to being templates?



Heinrich Lee Yu


@engwan · 4 months ago

Developer

Note that there are currently no duplicates there. We should probably set those back to being templates?

Yeah, I think we should. If we don't, we'd create a new set of these when we access the service templates page in the admin UI.

Having these extra rows won't really cause any problems though.



Sean McGivern

@smcgivern · 4 months ago


Maintainer

@hphilipps I think someone should fix these [#9176 \(comment 286198091\)](#).


UPDATE services SET template = 't' WHERE project_id IS NULL AND active != 't';

Should do it. When do we want to do that?

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


- 

Henri Philipps

@hphilipps changed the description 4 months ago · [Compare with previous version](#)
- 

Henri Philipps

@hphilipps assigned to [@hphilipps](#) 4 months ago



Sean McGivern

@smcgivern · 4 months ago

Maintainer

<https://log.gitlab.net/goto/034e27ed19fc35ce3bec423c82d530ab> has the non-GET requests made today by the user '[REDACTED]', identified in <https://gitlab.com/gitlab-com/gl-security/secops/operations/issues/650> as the potential creator of the first problem webhook.

The `webhook` param is filtered (as it should be), but the times seem like they line up. It's also notable that they were using canary, but beyond that, I still don't understand why this got set as a template.

Edited by [Henri Philipps](#) 2 days ago

▼ Collapse replies



Closed

RCA: 2020-02-11 High db insert rate caused by enabling issue templates on projects

Here are the service updates (web UI only) using canary today: <https://log.gitlab.net/goto/67ea68cb87ff1893b30249aee585dbe6>

There aren't many, but there are some from before the problem started that also update Slack services.



Heinrich Lee Yu @engwan · 4 months ago

Developer

That's because of the `PUT [REDACTED]_studio_app/-/services/slack`

This is a new service so we do `project.find_or_initialize_service(params[:id])` . Which initializes it from a template. So it's the same problem described above.

Edited by [Henri Philipps](#) 3 days ago



Sean McGivern @smcgivern · 4 months ago

Maintainer

Right, I see you explained this above, I just didn't understand it first time round 😊

Just rephrasing as I was already doing so ... the reason this project caused the issue is that:

1. The user was using canary.
2. The project didn't already have a Slack service (i.e. nobody had visited the integration settings page - we create these when viewing that page).
3. They then configured the service.

So as you said in [#9176 \(comment 286141635\)](#), we use the instance-level template and unset `instance` , but don't unset `template` , and then we have two problems:

1. Non-canary users will be using code that checks `template` , not `instance` .
2. The trigger will propagate `template` to `instance` anyway.

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



Henri Philipps @hphilipps · 4 months ago

Owner

MR for reverting the migration: gitlab-org/gitlab!24857 (merged).



David Smith 🌱 @dawsmith · 4 months ago

Owner

[@hphilipps](#) , [@marin](#) , [@lyle](#) - for Paging CMOC, I have changed the last line of escalation policies for Production and CMOC services to directly escalate by name (instead of schedule) to [@ansdval](#) and me. As a side note, I did not get a call/page related to this, but could have been available if needed. This should 'fix' that.

▼ Collapse replies



Lyle Kozloff 🛠️ @lyle · 4 months ago

Developer

Thanks [@dawsmith](#) - working on getting the CMOC rotation staffed this quarter in gitlab-com/support/support-team-meta#2098 (closed).



Henri Philipps @hphilipps · 4 months ago

Owner

Thanks [@dawsmith](#) ! And the page wasn't going out to you: <https://gitlab.pagerduty.com/incidents/P5PS2EL/timeline>. And things looked under control soon after the page so I didn't bother reaching out to you in this case.



David Smith 🌱 @dawsmith · 4 months ago

Owner

Makes sense, I just wanted to make sure the fall through ultimately went to people just in case.


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



[Henri Philipps](#) [@hphilipps](#) mentioned in issue [on-call-handovers#324 \(closed\)](#) 4 months ago

Mek Stittri @meks · 4 months ago

Maintainer



It seems the root cause is [gitlab-org/gitlab!23595 \(merged\)](#) ?, we need to pull in the engineers from there for the RCA as well.

▼

Closed

RCA: 2020-02-11 High db insert rate caused by enabling issue templates on projects




Marin Jankovski [@marin](#) · 4 months ago

Owner

Yes, that was the root cause.

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




Christopher Lefelhocz [@clefelhocz1](#) · 4 months ago

Developer

[@cdu1](#) can you get someone to run the RCA in enablement. Please check with infrastructure on the correct form to use.


▼ Collapse replies



Chun Du [@cdu1](#) · 4 months ago

Developer


[@nhxnguyen](#) Please run an RCA with the team and circle back the learnings to this thread. It may help by inviting an infrastructure team representative and learning our deployment process, which may help us prevent missteps like this in the future, given that all team members are new to GitLab. /cc [@meks](#) [@marin](#)



Nick Nguyen [@nhxnguyen](#) · 4 months ago

Developer


We are scheduling the team RCA for tomorrow and will share findings. Inviting [@smcgivern](#) as an infrastructure rep, but feel free to suggest someone else.



Sean McGivern [@smcgivern](#) · 4 months ago

Maintainer


Do we normally do team-specific RCAs? This was reviewed by people in other teams and took some time to find the bug, so I don't think that this is a problem with the Ecosystem team. Having the meeting for a specific team, most of whom didn't work on this, seems like we'll be limiting the value we get from it.



Mek Stittri [@meks](#) · 4 months ago

Maintainer


[@cdu](#), feel free to use <https://docs.google.com/document/d/1zoCIB7nVrbldvrUykbTna8Yd8aWHFiRkbH6oBgC7O54/edit#heading=h.hus829973byt> which is the format we have been using for the recent RCA meetings.



Chun Du [@cdu1](#) · 4 months ago

Developer

[@meks](#) Thanks for the pointer of template. [@smcgivern](#) The intention of RCA by Ecosystem team is actually about learning because everyone on the team is quite new to GitLab. On the other hand, an RCA by all participants is certainly more comprehensive to identify opportunities and prevent something similar from happening again.



Nick Nguyen [@nhxnguyen](#) · 4 months ago

Developer

We still found it useful to conduct a team-specific RCA to understand and learn from what happened. Thank you everyone for a great discussion and special thanks to [@smcgivern](#) for joining in and contributing your experience and insights.

Agenda: https://docs.google.com/document/d/1YM-ylcETiwj9QnVTICqH9hEH_cBATBvb278IEe4ywlk

Key Takeaways / Action Items:

- Added two additional Whys (issue description updated)
- Separating out the post-deploy migration to a separate MR could have helped detect issue during testing
- We will add database constraints as suggested in [#9176 \(comment 286149490\)](#).
- Add suggestion to [renaming columns docs](#) about splitting migration MRs

[@gitlab-org/ecosystem-team](#) Please add any other takeaways/actions I've missed



Heinrich Lee Yu [@engwan](#) · 4 months ago

Developer

Just wanted to add this after reading the doc:

Nick: Could we split up the MR?

Closed

RCA: 2020-02-11 High db insert rate caused by enabling issue templates on projects

Nailia: Would we have been able to spot on staging?

Andy: Might not have caught since we don't have canary/prod

Sean: What would have been the ideal way to split?

Andy: Put post-migration in MR

It's not just splitting into another MR, but it should also be in a separate deploy. If the post-migration cleanup was in a separate deploy, there's a chance we could have caught it in staging.

Even without a canary / prod split, the problem would show up because the DB trigger would still be there.

This wasn't a problem with the code split but rather a problem when the new code is running without the post-deploy migration cleanup.

The old code was actually working fine because we're still setting `template = false` and the trigger would propagate the change. Problem started when a service was created from canary with the DB trigger still around.

Edited by [Heinrich Lee Yu](#) 4 months ago



Mayra Cabrera ⚡ [@mayra-cabrera](#) · 4 months ago

Maintainer

[@nhxnguyen](#)

Separating out the post-deploy migration to a separate MR could have helped detect issue during testing

I'm not sure if this would have helped to detect the issue earlier, this was a very tricky one.

In retrospect, we were renaming a column (from `template` to `instance`) just to match a new terminology [gitlab-org/gitlab#196789 \(closed\)](#), while I agree renaming the code is important, perhaps the column renaming was not necessary.

`services` is one of our largest tables with ~3.5 million records, so instead of modifying their schema a simpler solution could've been to use an `alias_method` on `service.rb`, and leave the schema as-is.

cc [@Andysoiron](#)

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



Mek Stittri [@meks](#) changed the description 4 months ago · [Compare with previous version](#) ▾



Mek Stittri [@meks](#) · 4 months ago

Maintainer

Added the immediate corrective actions we took to the issue description as SSoT.

- Immediate corrective actions were to revert the change.
 - [gitlab-org/gitlab!24857 \(merged\)](#).
 - [gitlab-org/gitlab!24885 \(merged\)](#).

▼ Collapse replies



Mayra Cabrera ⚡ [@mayra-cabrera](#) · 4 months ago

Maintainer

There was another corrective action [gitlab-org/gitlab!24949 \(merged\)](#)/. All merge requests are now on production 🎉

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



Tiger Watson [@tigerwnz](#) mentioned in merge request [gitlab-org/gitlab!24767 \(merged\)](#) 4 months ago



Andy Soiron 🟡 [@Andysoiron](#) mentioned in issue [gitlab-org/gitlab#204801 \(closed\)](#) 4 months ago



Yorick Peterse [@yorickpeterse](#) mentioned in issue [gitlab-org/gitlab#205207](#) 4 months ago



Nick Nguyen @nhxnnguyen changed the description 4 months ago · [Compare with previous version](#) ▾

Closed **RCA: 2020-02-11 High db insert rate caused by enabling issue templates on projects**

~~**Nick Nguyen** @nhxnnguyen changed the description 4 months ago · [Compare with previous version](#) ▾~~

GitLab SecurityBot @gitlab-securitybot changed due date to March 13, 2020 4 months ago

Henri Philipps @hphilipps changed the description 4 months ago · [Compare with previous version](#) ▾

Andy Soiron 🍷 @Andysoiron mentioned in merge request [gitlab-org/gitlab!25110 \(merged\)](#) 4 months ago

Andy Soiron 🍷 @Andysoiron mentioned in issue [gitlab-org/gitlab#205419 \(closed\)](#) 4 months ago



Andy Soiron 🍷 @Andysoiron · 4 months ago

Developer

I added [gitlab-org/gitlab#205419 \(closed\)](#) to add database constraints to the services table



Mek Stittri @meks mentioned in issue [gitlab-com/www-gitlab-com#6556](#) 4 months ago



Mek Stittri @meks marked this issue as related to [gitlab-com/www-gitlab-com#6556](#) 4 months ago



Henri Philipps @hphilipps · 4 months ago

Owner

Closing, as we have issues for all corrective actions.



Henri Philipps @hphilipps closed 4 months ago



Mayra Cabrera ⚡ @mayra-cabrera mentioned in merge request [gitlab-org/gitlab!25624 \(merged\)](#) 4 months ago



Arturo Herrero @arturoherrero mentioned in merge request [gitlab-org/gitlab!26380 \(merged\)](#) 4 months ago



Nick Nguyen @nhxnnguyen mentioned in issue [gl-retrospectives/ecosystem#5 \(closed\)](#) 4 months ago



Nick Nguyen @nhxnnguyen mentioned in issue [gitlab-com/www-gitlab-com#6757 \(closed\)](#) 4 months ago



Arturo Herrero @arturoherrero mentioned in issue [gitlab-org/gitlab#208925 \(closed\)](#) 4 months ago



Mayra Cabrera ⚡ @mayra-cabrera mentioned in issue [gitlab-com/www-gitlab-com#5186 \(closed\)](#) 3 months ago



Mayra Cabrera ⚡ @mayra-cabrera mentioned in merge request [gitlab-org/gitlab!25714 \(merged\)](#) 3 months ago



David Smith 🌴 @dawsmith mentioned in issue [#9797 \(closed\)](#) 2 months ago



Mayra Cabrera ⚡ @mayra-cabrera mentioned in merge request [gitlab-org/gitlab!33017 \(merged\)](#) 1 month ago



Peter Leitzen @splattael · 3 days ago

Developer

[@hphilipps](#) 🙌

Coming from https://docs.gitlab.com/ee/development/multi_version_compatibility.html#project-service-templates-incorrectly-copied

For more information, see this confidential issue [#9176 \(closed\)](#).

Are there any plans to make this issue public? 🤔 Which confidential data (project name/ids?) should be redacted from the comments?

▼ Collapse replies



Henri Philipps @hphilipps · 3 days ago

Owner

[@splattael](#) I redacted one project path [here](#). Can't see much else that shouldn't be public, but can you give it another look please? Then I'd be fine to make it public.



Peter Leitzen @splattael · 2 days ago

Developer

[@hphilipps](#) Thanks! Do we have to redact project ids (e.g. [#9176 \(comment 286143098\)](#))?

If no, we are good to make it public from my side 🙌



Henri Philipps @hphilipps · 2 days ago


Closed

RCA: 2020-02-11 High db insert rate caused by enabling issue templates on projects

Owner

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

- 

Henri Philipps @hphilipps made the issue visible to everyone [2 days ago](#)
- 

Peter Leitzen @splattaeI mentioned in commit [gitlab-org/gitlab@d2a3dfba](#) [2 days ago](#)
- 

Peter Leitzen @splattaeI mentioned in merge request [gitlab-org/gitlab!35986](#) [2 days ago](#)

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