Author	DK Benjamin Ramya Ragupathy
Report Date	Oct 31, 2023
Incident Date(s)	Oct 11 - 17
Incident	Tasking Manager 4.6.2 Hotfix

Description:

An incident is an unintended event/happening/occurrence that disturbs and disrupts normal operation. This Incident Post-Mortem provides a framework to reflect on why an incident occurred, its impact, what actions were taken to mitigate and resolve it, and what should be done to prevent it from happening again. This template is not a tool for 'pointing fingers' or assigning blame...incident post-mortems should be blameless activity, allowing the people involved to account for their actions, impact, what they knew and when, etc. without fear of punishment or retribution. This process helps to uncover systemic gaps, fosters teamwork, improves departmental coordination, and provides actionable insights.

Executive Summary

In 2-5 sentences, summarize the report below.

After the V4.6.2 deployment users reported several frontend issues affecting multiple workflows across tools. Upon initial investigation, these stem from a refactoring done on the frontend that was part of the V4.6.2 release. We missed catching these issues owing to lack of sufficient test cases for the frontend codebase.

Incident Report

1. ## Incident - What failed, and why?

   In 1-2 paragraphs, describe the list of events that led up to the incident, the incident itself and its internal/external impact.

   Release 4.6.2
   One issue that was reported in #tasking-manager on slack:
   - *Authorised users were unable to access private and draft projects*
     https://github.com/hotosm/tasking-manager/issues/6088

2. ## Response - How did those involved react?

   In 1 paragraph, describe what individual/team first detected the issue, who responded, and how the issue was resolved.

   DK responded on Slack to multiple users reporting issues with private and draft projects. DK also flagged this with the development team to investigate further. Aditya was able to figure out the issue and applied a fix which was tested by Royall. Later this was pushed to development setup. Initial testing was done by the team and DK pushed this as a hotfix on the same day as it was critical to the workflow of several organisations.

3. ## Timeline

   Briefly layout the full timeline of events described in sections 1 and 2.

   18 Oct 12AM UTC - V4.6.2 deployment
   18 Oct 16 UTC - Patrik reported, Anthaas confirmed
   19 Oct 16 UTC - Nicole reported and Jorieke confirmed

20 Oct 6AM UTC - Fix made available on development instance

20 Oct 9AM UTC - Testing done by team

20 Oct 17 UTC - Hotfix published for the workflow

4. Root Cause

*Without assigning blame, describe the root cause of the incident. Conduct the '5 whys analysis' (explanations **here** and **here** and example **here**) to help with this process if needed.*

Problem statement: private and draft projects on TM were inaccessible to authorized users.

Why? 1 month ago PR refactoring the react code. Auth token was missing from the API request for accessing private and draft projects.

Why wasn't this flagged?
- *Code changes after review approval - were not peer reviewed*
- *PM was walked through changes for functionality but not for testing*
- *Tests passed despite regression*
- *Transition in teams*
- *Not enough eyes on staging setup*

5. Related Incidents
Detail any previous related incidents that occurred, including any mitigation efforts that were attempted.

- Inconsistent user level labelling https://github.com/hotosm/tasking-manager/issues/6113
- Username tagging not working in Project Q&A section: https://github.com/hotosm/tasking-manager/issues/6114
- Incomplete contributor list in task comments: https://github.com/hotosm/tasking-manager/issues/6115

- Missing pagination links in My Tasks section:
  https://github.com/hotosm/tasking-manager/issues/6116

6. Lessons Learned

Detail the top 3-5 lessons learned, including what went well and what can be improved.

- On response: good to have multiple people in timezones for responding and addressing users
- We should have thorough tests (smoke tests) to catch issues like this
- When code is refactored: check for every change that code affects- more precisely done
- Coordination was good - pretty timely on hotfix

7. Follow-up Tasks

Note follow-up tasks, who is responsible, and what is the timeline.

- ☐ Playbook for UAT on staging Ramya Ragupathy   manjita.naxa@gmail.com
- ☐ Fix other endpoints that were touched as part of the refactoring @Naxa team
- ☐ Another hotfix as early as possible DK Benjamin