Summary of Issue Being Investigated

On Thursday June 9 and then again on Friday June 10, Beam's Jenkins instance went down (Infra ticket). Infra diagnosed the source of the issue as the ghprb Jenkins plugin opening 1000s of GitHub requests. Additionally, they provided logging that showed the ghprb plugin logging that it is making all of these requests.

Possible Hypotheses

- Last weekend (June 3-4), the Beam project introduced GitHub issues and imported almost 4,000 new issues. Since the ghprb plugin listens to issue comment events, a bug in how those are handled could cause a problem to arise if it led to a significant increase in requests to GitHub.
- The ghprb plugin has always been close to tipping over the Jenkins server. A combination of a few extra issue events and introducing a new build tipped it over.
- The ghprb plugin is a red herring

Analysis of ghprb Source Code

ghprb does the following things relevant to this investigation:

1. Configures a GitHub webhook to listen to new pull requests and issue comments. Issue comments includes both comments on PRs and Issues. This is configured
2. The webhook triggers the doIndex method which processes the event, extracting the payload
3. doIndex calls handleAction which establishes a connection to GitHub, and then branches on whether it is a pr event or an issue comment event. If it is a pr event, it logs that it is checking the PR and then continues on to handleEvent. If it is a comment event, it checks if it is a comment on a closed PR or an issue. In either case, if it is the plugin logs and early returns. If it is an open PR, it also continues on to handleEvent.
4. handlePr iterates through each configured trigger (one per configured build), and for each trigger processes the PR or issue, before eventually adding the trigger to a thread pool so that its run method gets called eventually. That calls triggerPr or triggerComment which in turn call handlePr and handleComment respectively.
5. handlePr and handleComment each log before calling onPullRequestHook and onIssueCommentHook respectively. These do most of the heavy lifting, including usually calling the GitHub API.

Analysis of the logging provided by Infra

All data included in this section is pulled from queries in the Queries used to get data section. Over the period from June 1-10, 211 distinct PRs are mentioned, with 1,035,400 log lines mentioning a PR number. 330 issue comment events (not including PR comments) were processed, triggering the correct early return. Since issues first appeared on June 3, this accounts for ~50 issue comment events per day, each with at most 1 call (though its not clear any calls are actually being made). 45 comments on closed prs were processed, also triggering a correct early return. These account for ~5 events per day, each also with at most 1 call.

On the other hand, 1,470 pr comment events were correctly processed (~150 per day) and 1,062 pr events were correctly processed (~100 per day). Each of these in turn triggered 100s of configurations to read the prs/comments, calling into GitHub in the process. For example, the beam_PostCommit_Java_Tpcds_Dataflow job checked a PR or PR comment 1318 times between June 3 and June 10 (~170 times per day), each in response to a valid PR or PR comment. That job is only special in that it was added recently - other jobs exhibit similar behavior (and actually have more logs associated with them because they existed earlier).

Conclusions

1. Evidence seems to point to the ghprb plugin handling issue comments correctly and making at most a single call per comment (and potentially no calls).
2. There are not a significant enough number of issue comments to overwhelm the CI server.
3. The proliferation of job triggers has led to a very large number of calls to GitHub. This was most likely close to tipping over before and probably has not drastically changed.

Recommendations

1. Make sure the Jenkins plugin is up to date. All analysis is based on the latest version of the plugin, but I don't have visibility to confirm that is indeed the one being used.
2. Disable some Jenkins comment triggers to reduce load of connections.
3. If that doesn't fix the problem, upgrade from ghprb to a different plugin (MultiBranch Pipelines has been recommended in the ASF slack).
4. In the long term, consider porting to GitHub Actions.

Other options considered:

1. Turn off issues (and revert to Jira) - there's not sufficient evidence to support the idea that this will help.
2. Try to patch the plugin. This would be possible, but not easy. The proper fix would be to introduce shared state or caching to avoid the large number of hits to GitHub. Between how the plugin is structured (multithreaded with different triggers largely unaware of each other) and its dependency structure (calls to GitHub happen via a 3rd party dependency with pretty tight ties to the hook structure), this would be pretty hard to pull off.
3. Emergency port to GitHub Actions - we don't have private runners yet and would quickly exhaust the allotted concurrency from ASF. This is also not a trivial port and would take a significant amount of effort.

Queries used to get data

The following nodejs queries were used to generate the data from the downloaded log file:

```
// Get all numbers of PRs or issues that are being gotten - 211 distinct,  1035400
total, starting June 1
fs.readFileSync('ghprb.txt', {encoding: 'utf-8'}).
   split('\n').
   filter(val => val.indexOf(',') > -1).
   map(l => l.substring(l.indexOf(',')-2, l.indexOf(',')) +
l.substring(l.indexOf(',') + 1, l.indexOf(',')+4)).
   filter((val, index, self) => index == 0 || self[index-1] != val).
   filter((val, index, self) => self.indexOf(val) == index).
   filter(val => !isNaN(val))

// Get the number of times an issue comment is skipped - 330, starting June 3 (~50
per day)
fs.readFileSync('ghprb.txt', {encoding: 'utf-8'}).
   split('\n').
   filter(val =>
val.indexOf('org.jenkinsci.plugins.ghprb.GhprbRootAction.handleAction Skip comment
on Issue') > -1).
   length

// Get the number of times a pr comment is skipped (pr is closed) - 45,  starting
June 1 (~5 per day)
fs.readFileSync('ghprb.txt', {encoding: 'utf-8'}).
   split('\n').
   filter(val =>
val.indexOf('org.jenkinsci.plugins.ghprb.GhprbRootAction.handleAction Skip comment
on closed PR') > -1).
   length

// Get the number of times a comment is processed (pr is open) - 1470, starting
June 1 (~150 per day)
fs.readFileSync('ghprb.txt', {encoding: 'utf-8'}).
   split('\n').
   filter(val =>
val.indexOf('org.jenkinsci.plugins.ghprb.GhprbRootAction.handleAction Checking
issue comment') > -1).
   length

// Get the number of times a pr is processed (pull request event) - 1062, starting
June 1 (~100 per day)
fs.readFileSync('ghprb.txt', {encoding: 'utf-8'}).
   split('\n').
   filter(val =>
val.indexOf('org.jenkinsci.plugins.ghprb.GhprbRootAction.handleAction Checking PR')
> -1).
   length

// Get the number of times the beam_PostCommit_Java_Tpcds_Dataflow job has checked
a pr - 1318, starting June 3 (~170 per day)
fs.readFileSync('ghprb.txt', {encoding: 'utf-8'}).
   split('\n').
   filter(val => (val.indexOf('Checking comment on PR') > -1 ||
val.indexOf('Checking PR') > -1) && val.indexOf('for job
beam_PostCommit_Java_Tpcds_Dataflow') > -1).
   length
```