



Self Service of GitHub Organizations

Thomas Neidhart, Eclipse Foundation

Problem



- Eclipse Foundation hosts > **400** projects with > **2000** source repositories
- How to ensure that certain settings are applied consistently to all repositories?
- How to access the current configuration without escalating privileges or going through HelpDesk?
- How to quickly verify if certain settings are already enabled for an organization or its repositories? E.g. secret scanning

Solution

- Use an IaC approach similar to terraform
- Host the configuration for each organization in a public repo of the organization itself
- Support an approval process for changing the current configuration



Goals



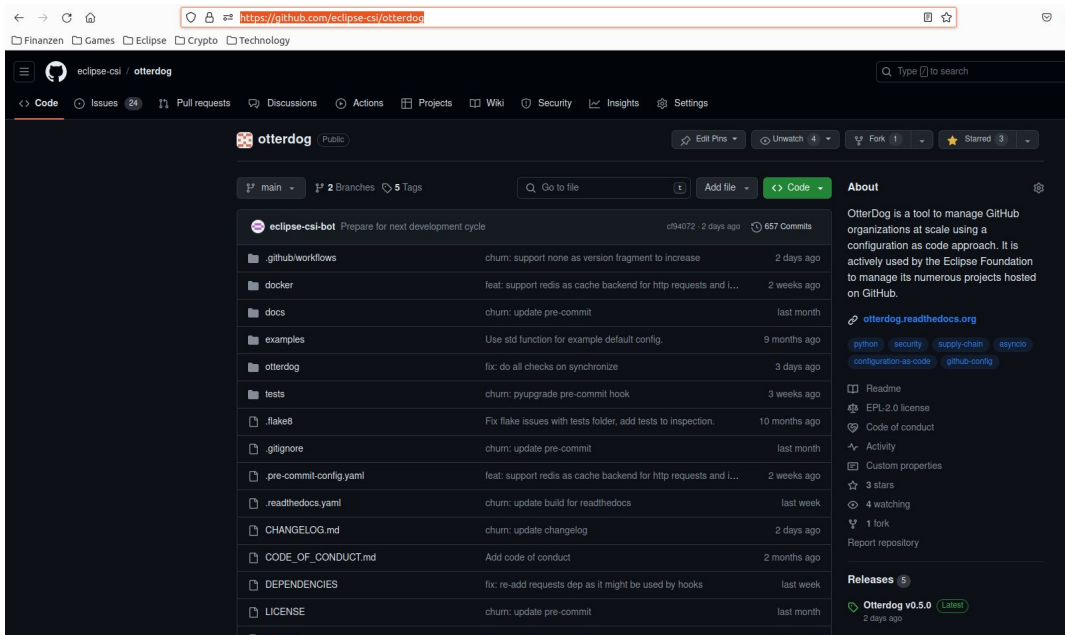
- Support the definition of foundation wide default settings and overriding of those defaults on organization / repo level
- Non disruptive at setup, support for importing the current status quo to quickly bootstrap
- Allow more “self-service” for projects, while not escalating individual privileges
 - Support verification and validation of changes by EF staff and / or project leads
- Provide powerful monitoring and querying capabilities for EF staff to identify areas that need improvement

Benefits



- History of changes to GitHub resources in a single place (**.eclipsefdn** repo) with comments and author
- Support for consistent configuration of resources across multiple repositories
- Simplified verification of the authenticity for requested changes via GitHub handles
- Transparency of current resource configuration to everyone involved in the project
- Ability for everyone to improve the configuration to make the source repos more secure / polished
- Learn from and collaborate with other projects

Open-source tool



<https://github.com/eclipse-csi/otterdog>



GitHub Resources Support

- Organizations
 - Including settings that are not accessible via an API
 - e.g., `members_can_change_repo_visibility`, `members_can_create_teams`, `default_branch_name`
- Organization Secrets / Variables / Webhooks / Workflow settings
- Repositories
 - Branch Protection Rules
 - Rulesets
 - Secrets / Variables / Webhooks / Workflow settings
 - Environments

<https://otterdog.readthedocs.io/en/latest/reference/organization/>

Configuration as Code: storage location



.eclipsefdn repo





https://github.com/eclipse-cbi/.eclipsefdn/tree/main/otterdog

Finanzen Games Eclipse Crypto Technology

eclipse-cbi / .eclipsefdn

Code Issues Pull requests Actions Security Insights Settings

Files

main + 🔍

Go to file 🔍

- > .github
- > docs
- ▼ otterdog
 - eclipse-cbi.jsonnet
 - README.md
 - mkdocs.yml
 - requirements.txt

.eclipsefdn / otterdog / 📄

eclipse-cbi-bot Deleting file 'otterdog/jsonnetfile.lock.json' with otterdog. ✓

Name	Last commit message
..	
eclipse-cbi.jsonnet	Delete branch after merge automatically (#10)

Configuration as Code: jsonnet definition



<github org id>.jsonnet



Configuration as Code: jsonnet intro

- Jsonnet is a configuration language that adds templating support for json data
- More info at <https://jsonnet.org/>
- The jsonnet configuration contains the differences to the default configuration available at <https://github.com/EclipseFdn/otterdog-defaults/>
- Settings are inherited from the default configuration but can be overwritten
- Extend some settings from the default configuration

```
orgs.newOrg('eclipse-cbi') {
  settings+: {
    billing_email: "webmaster@eclipse.org",
    blog: "https://projects.eclipse.org/projects/technology.cbi",
    default_repository_permission: "none",
    description: "The Eclipse CBI project",
    email: "cbi-dev@eclipse.org",
    location: "Belgium",
    name: "Eclipse CBI",
```

Configuration as Code: jsonnet example

```
1  local orgs = import 'vendor/otterdog-defaults/otterdog-defaults.libsonnet';
2
3  local newBranchProtectionRule(branchName) = orgs.newBranchProtectionRule(branchName) {
4    required_approving_review_count: null,
5    requires_pull_request: false,
6    requires_status_checks: false,
7  };
8
9  orgs.newOrg('eclipse-cbi') {
10   settings+: {
11     billing_email: "webmaster@eclipse.org",
12     blog: "https://projects.eclipse.org/projects/technology.cbi",
13     default_repository_permission: "none",
14     description: "The Eclipse CBI project",
15     email: "cbi-dev@eclipse.org",
16     location: "Belgium",
17     name: "Eclipse CBI",
18     packages_containers_internal: false,
19     readers_can_create_discussions: true,
20     security_managers+: [
21       "technology-cbi-project-leads"
22     ],
23     workflows+: {
24       allow_action_patterns+: [
25         "ludeeus/action-shellcheck@*",
26         "marocchino/sticky-pull-request-comment@*",
27         "release-drafter/release-drafter@*"
28       ],
29       allowed_actions: "selected",
30       default_workflow_permissions: "write",
31     },
32   },
```

Configuration as Code: customization

- Jsonnet offers powerful support for creating customizations as needed
- Some examples from other Eclipse projects:

```
local extractVersion(name) =
  local prefix = "temurin";
  local suffix = "-binaries";
  local versionStart = std.length(prefix);
  local versionEnd = std.length(name) - std.length(suffix);
  if std.startsWith(name, prefix) && std.endsWith(name, suffix) then
    std.substr(name, versionStart, versionEnd - versionStart)
  else
    "unknown";

local newBinaryRepo(repoName) = orgs.newRepo(repoName) {
  description: "Temurin %s binaries" % [extractVersion(repoName)],
  dependabot_alerts_enabled: false,
  dependabot_security_updates_enabled: false,
  has_issues: false,
  has_projects: false,
  has_wiki: false,
  homepage: "https://adoptium.net",
};
```

Configuration as Code: customization

```
local vertexBranchProtectionRule(branchName) = orgs.newBranchProtectionRule(branchName) {
  required_approving_review_count: null,
  requires_pull_request: false,
  requires_status_checks: false,
  requires_strict_status_checks: true,
};

local newVertexRepo(repoName, default_branch = 'main') = orgs.newRepo(repoName) {
  allow_merge_commit: true,
  allow_update_branch: false,
  default_branch: default_branch,
  delete_branch_on_merge: false,
  homepage: "http://vertx.io",
  web_commit_signoff_required: false,
  branch_protection_rules: [
    vertexBranchProtectionRule($.default_branch) {},
  ],
};
```

Configuration as Code: Default configuration

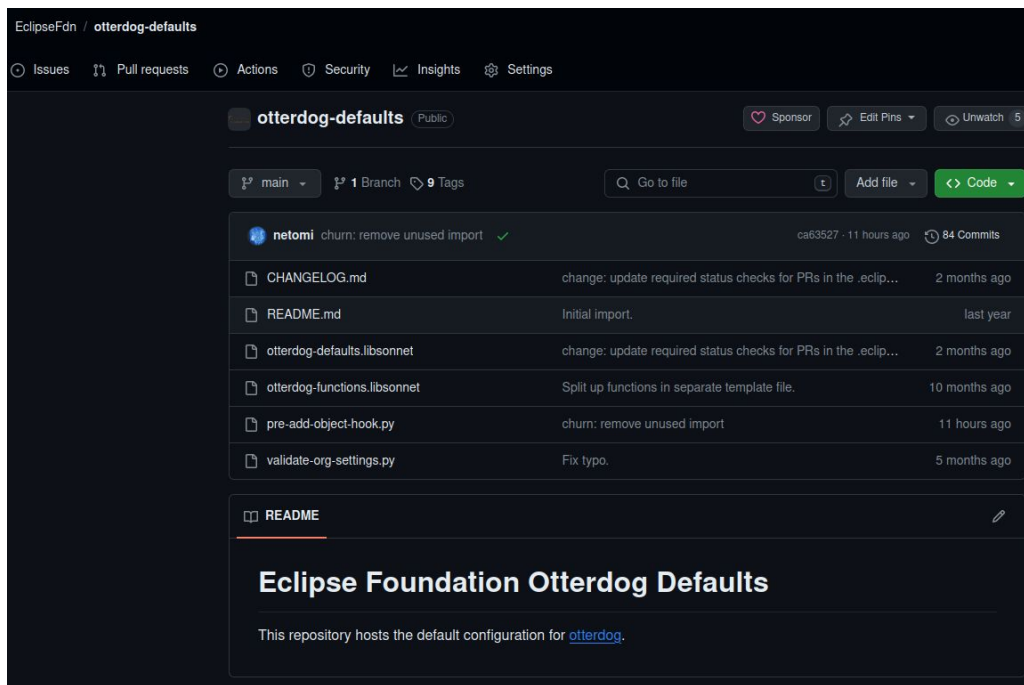


otterdog-defaults.jsonnet



Configuration as Code: default config

- Hosted at <https://github.com/EclipseFdn/otterdog-defaults/>
- Provides template functions for all resources with default settings
- Support for custom hooks
- Versioned



Code Blame 209 lines (169 loc) · 6.5 KB

```
84     reviewDismissalDismissal: false,
85     reviewDismissalAllowances: [],
86     requireLastPushApproval: false
87   };
88
89   # Function to create a new organization with default settings.
90   local newOrg(id) = {
91     github_id: id,
92     settings: {
93       name: null,
94       plan: "free",
95       billing_email: "webmaster@eclipse-foundation.org",
96       company: null,
97       email: null,
98       twitter_username: null,
99       location: null,
100      description: null,
101      blog: null,
102
103      has_organization_projects: true,
104      has_repository_projects: true,
105
106      # Base permissions to the organization's repositories apply to all members and excludes outside collaborators.
107      # Since organization members can have permissions from multiple sources, members and collaborators who have been
108      # granted a higher level of access than the base permissions will retain their higher permission privileges.
109      # Can be one of: read, write, admin, none
110      default_repository_permission: "read",
111
112      # Repository creation
113      members_can_create_private_repositories: false,
114      members_can_create_public_repositories: false,
```



**How does it
work?**



Review the config via the Dashboard

Access it via <https://otterdog.eclipse.org>, e.g.

<https://otterdog.eclipse.org/projects/adoptium>

The screenshot shows the Otterdog dashboard for the 'adoptium' project. The sidebar on the left contains navigation options: Dashboard, Query, and a list of projects including adoptium, automotive, dt, ecd, eclipse, ee4j, iot, modeling, and oniro. The main content area has a search bar and a navigation menu with tabs for Overview, Settings, Workflow Settings, Secrets, Variables, Webhooks, and Repositories. The 'Overview' tab is selected, showing a 'General' section with project details and two donut charts.

Category	Value
Project	adoptium
GitHub Org	adoptium
2FA enforced	True
Default workflow permissions	read
Secrets	4
Variables	0
Webhooks	1
Repositories	68

Secret Scanning

Mode	Count
disabled	1
alert mode	0
protection mode	3

Branch Protections

Status	Count
not protected	3
protected	3

Use the playground to sketch changes

Currently accessible at <https://<org>.github.io/eclipsefdn/playground/>

Playground

You can use the playground below to create and evaluate resource snippets to include in your jsonnet configuration:

playground.jsonnet

```
1 local orgs = import 'otterdog-defaults.libsonnet';
2
3 orgs.newBranchProtectionRule('main') {
4   required_approving_review_count: 1
5 }
```

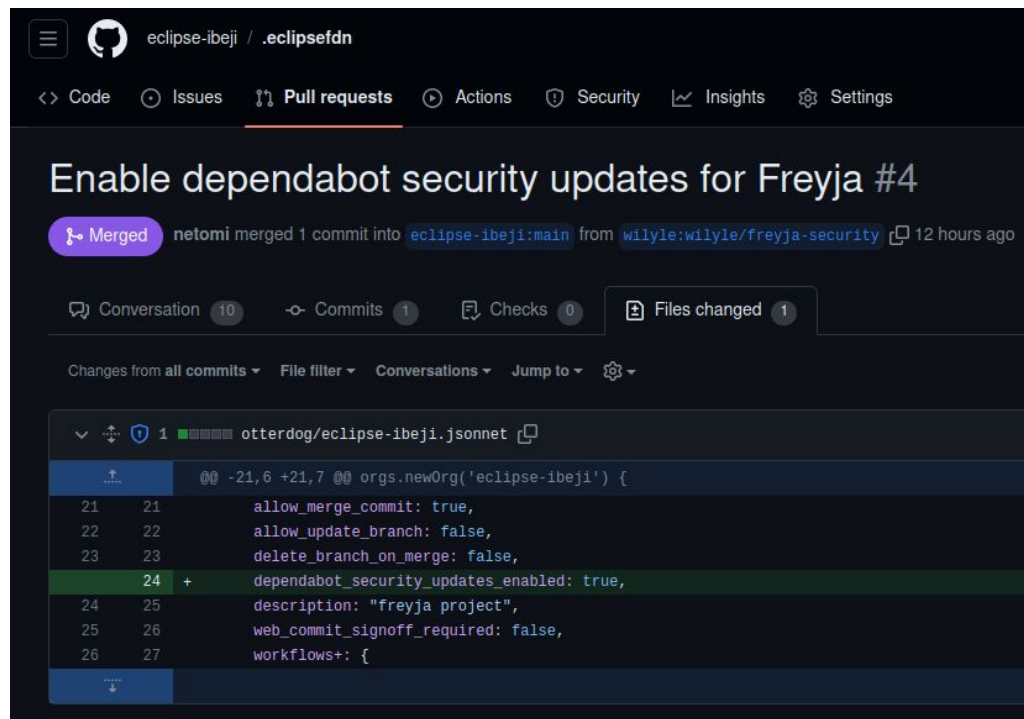


output.json

```
{
  "allows_deletions": false,
  "allows_force_pushes": false,
  "blocks_creations": false,
  "bypass_force_push_allowances": [],
  "bypass_pull_request_allowances": [],
  "dismisses_stale_reviews": false,
  "is_admin_enforced": false,
  "lock_allows_fetch_and_merge": false,
  "lock_branch": false,
  "pattern": "main",
  "push_restrictions": [],
  "require_last_push_approval": false,
  "required_approving_review_count": 1,
  "required_deployment_environments": [],
  "required_status_checks": [
    "eclipse-eca-validation:eclipsefdn/eca"
  ],
  "requires_code_owner_reviews": false,
  "requires_commit_signatures": false,
  "requires_conversation_resolution": false,
  "requires_deployments": false,
  "requires_linear_history": false,
  "requires_pull_request": true,
  "requires_status_checks": true,
  "requires_strict_status_checks": false
}
```

Apply changes via a PR

- Fork <org id>/eclipsefdn repo
- Create branch with changes to the jsonnet configuration file
- Create a PR to the upstream repo

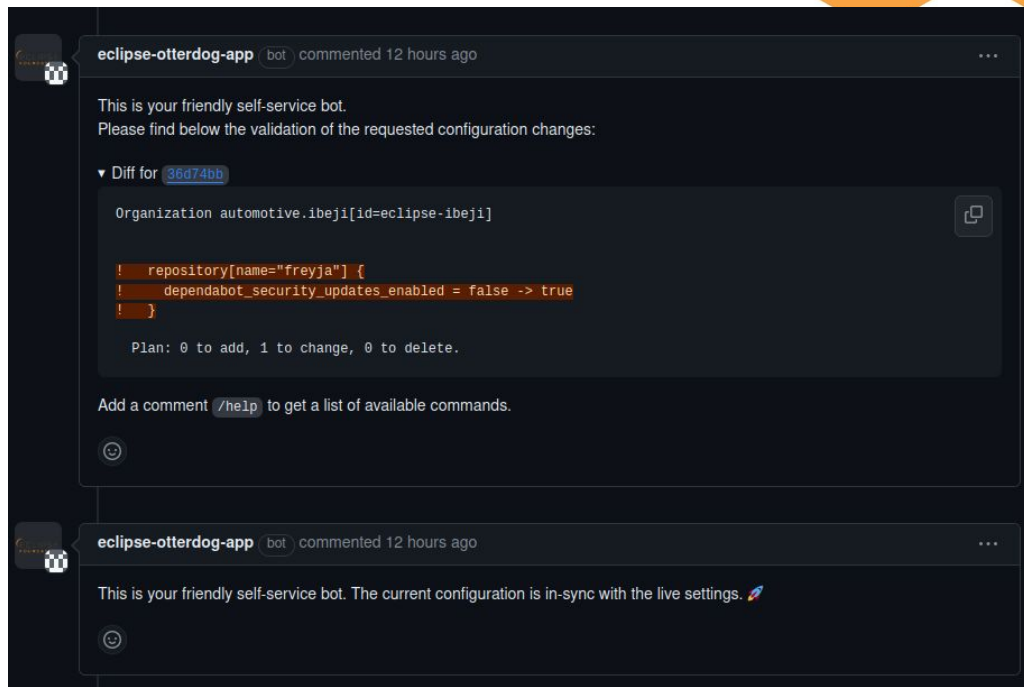


The screenshot shows a GitHub Pull Request interface. At the top, the repository is identified as 'eclipse-ibeji / .eclipsefdn'. The title of the PR is 'Enable dependabot security updates for Freyja #4'. It is marked as 'Merged' and shows that 'netomi' merged 1 commit into 'eclipse-ibeji:main' from 'wilye:wilye/freyja-security' 12 hours ago. Below the title, there are statistics: 10 Conversations, 1 Commit, 0 Checks, and 1 File changed. The main content area shows a diff for the file 'otterdog/eclipse-ibeji.jsonnet'. The diff highlights a change on line 24, where 'dependabot_security_updates_enabled' is set to 'true'. The code snippet is as follows:

```
@@ -21,6 +21,7 @@ orgs.newOrg('eclipse-ibeji') {
 21 21     allow_merge_commit: true,
 22 22     allow_update_branch: false,
 23 23     delete_branch_on_merge: false,
 24 +   dependabot_security_updates_enabled: true,
 24 25     description: "freyja project",
 25 26     web_commit_signoff_required: false,
 26 27     workflows+: {
```

Review suggested changes

- A GitHub App will automatically validate the PR and add a comment to the PR with changes in diff format
- PR needs to be approved by an EF staff member and / or a project-lead
- After approval the PR can be merged and the changes will be automatically applied by the GitHub App



```
eclipse-otterdog-app bot commented 12 hours ago

This is your friendly self-service bot.
Please find below the validation of the requested configuration changes:

▼ Diff for 36d74bb

Organization automotive.ibeji[id=eclipse-ibeji]

! repository[name="frejja"] {
!   dependabot_security_updates_enabled = false -> true
! }

Plan: 0 to add, 1 to change, 0 to delete.

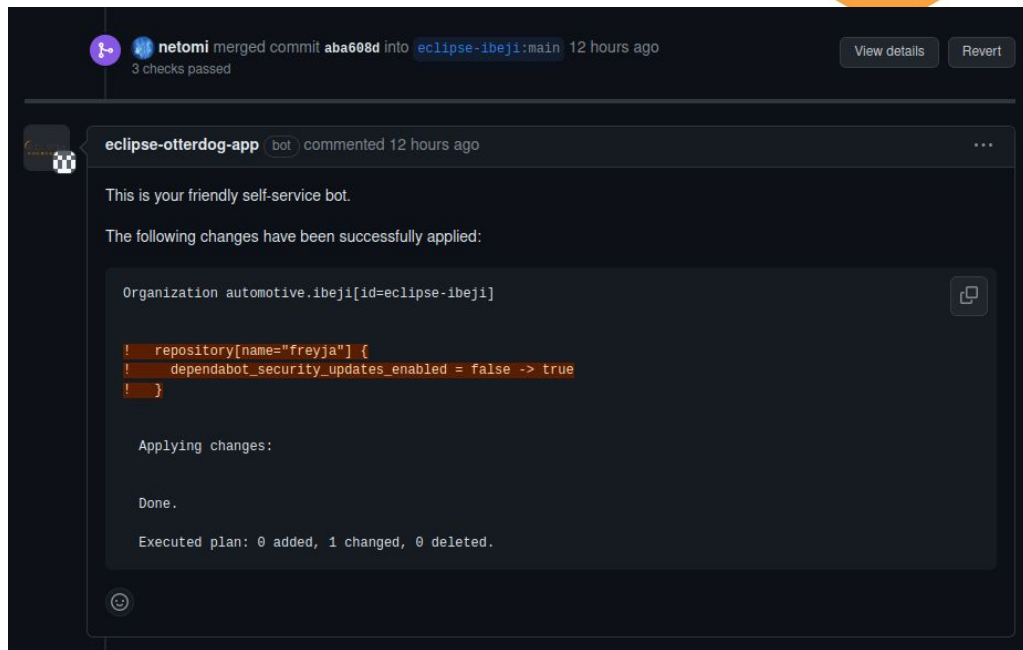
Add a comment /help to get a list of available commands.
```

```
eclipse-otterdog-app bot commented 12 hours ago

This is your friendly self-service bot. The current configuration is in-sync with the live settings.
```

Apply accepted changes

- Once the PR got merged, the GitHub App will automatically apply the changes
- Some requested changes might have to be applied manually as they require some access to credentials that the GitHub App currently does not have
- To apply the changes manually, the cli version of otterdog will be used



Apply accepted changes

- Extended example of changes including additions / deletions and require additional manual intervention due to secrets

```
- remove repo_webhook[url="https://readthedocs.org/api/v2/webhook/zenoh-python/135566/", repository="zenoh-py]
- active = true
- content_type = "form"
- events = [
-   "push"
- ],
- insecure_ssl = "0"
- secret = null
- url = "https://readthedocs.org/api/v2/webhook/zenoh-python/135566/"
- }

+ add repo_webhook[url="https://readthedocs.org/api/v2/webhook/zenoh-python/263749/", repository="zenoh-pytho]
+ active = true
+ content_type = "json"
+ events = [
+   "create"
+   "delete"
+   "push"
+   "pull_request"
+ ],
+ insecure_ssl = "0"
+ secret = "pass:bots/iot.zenoh/readthedocs.org/zenoh-python-webhook-secret"
+ url = "https://readthedocs.org/api/v2/webhook/zenoh-python/263749/"
+ }
```

Plan: 4 to add, 0 to change, 2 to delete.

Warnings

- some of requested changes require secrets, need to apply these changes manually

cc @eclipse-zenoh/eclipsefdn-security

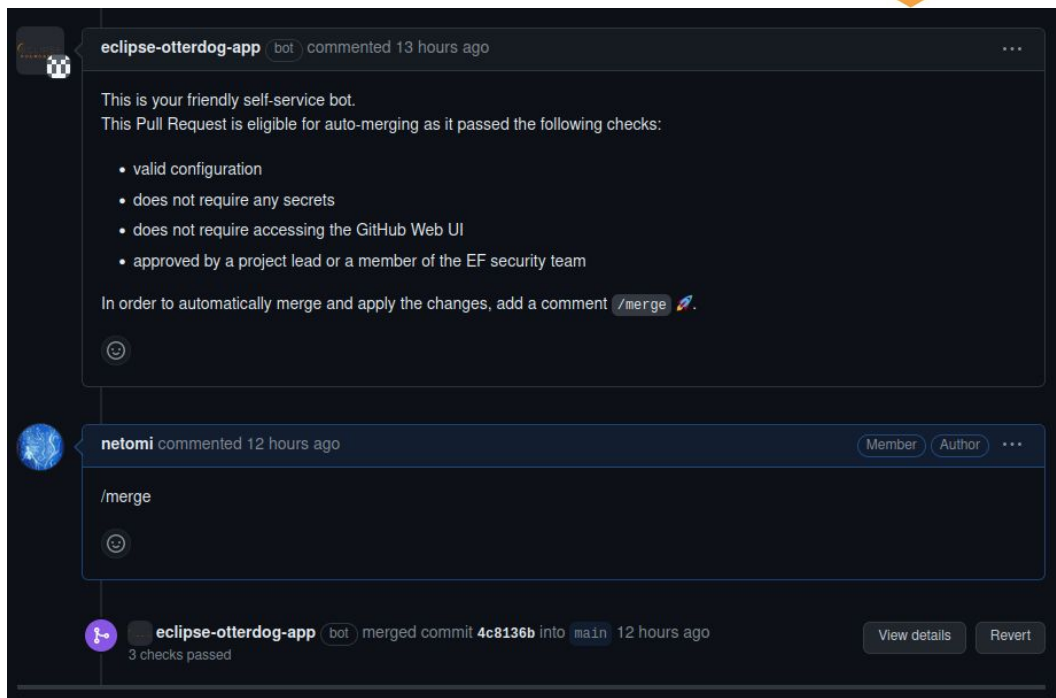
Add a comment `/help` to get a list of available commands.

General Approval rules

- EF staff does some sanity checks for each PR (type of change, author, validity)
- Additional approval might be required from project leads
- Idea to support more customization of approval rules on a project level
- Lets us know what you want / need for your project

Auto merging support - New

- In most cases, PRs are eligible for auto-merging by the author of the PR itself, i.e. if no secrets are involved and no resources are deleted
- Must come from a project lead itself, or approved by a project lead
- If all conditions are met, the author can merge and apply the change him/herself by adding **/merge** as a comment



What's next for the self-service?

- Formalize and automatically enforce approval rules for each project, some examples:
 - 1 approval from a project lead is required (default)
 - certain committers are trusted to make any change
 - custom approval rules depending on the type of change
- Add more monitoring and alerting capabilities
 - Dashboard is a convenient access point to monitor managed organizations
 - Provide help and hints about suggested improvements to make the organization / repo more secure (see next slide)
 - Monitor the presence of various important information on a per-repo basis, e.g. SECURITY.md, DEPENDENCIES, ...
- Add information where artifacts are being published to



Simple things to do right away for your project

- Enforce 2FA for your organization
 - Make sure that all members have 2FA enabled
 - Organization will then be switched to enforce 2FA as well
 - Emails have been sent out to each project, please act accordingly
- Enable secret scanning / push protection for each repo if not yet done (check dashboard to see which repo is missing)
- Set default workflow permissions to “**read**” and grant necessary permissions on a workflow basis
- Pin actions that are used in your workflows
 - Developed a tool to make pinning easier: <https://github.com/TinyGearsOrg/octopin/>
 - Will be moved to the eclipse-csi organization as well
- Enable branch protection rules for main branches to prevent force pushes
- Enable dependabot security alerts for all your repos
- Enable private vulnerability reporting for relevant repos

Browse Dashboard

Browser address bar: <https://otterdog.eclipse.org/index>

Navigation: Home

Dashboard

Home / Dashboard

91 GitHub Organizations

4 Open Pull Requests

479 Merged Pull Requests

1179 Total Repository Count

Organization	2FA enforced	Default workflow permissions	Repositories
adoptium	True	read	68
automotive.ankaaios	True	read	2
automotive.bluechi	True	read	6
automotive.chariott	True	read	4
automotive.ibeji	True	write	4
automotive.kuksa	True	read	18
automotive.leda	True	write	13
automotive.muto	False	write	15

Browse Dashboard

The screenshot shows the Eclipse Otterdog Browse Dashboard for the project automotive.uprotocol. The dashboard is divided into several sections: a left sidebar with navigation options, a top navigation bar, and a main content area with a table of repository status.

Navigation:

- Home
- Overview
- Settings
- Workflow Settings
- Secrets
- Variables
- Webhooks
- Repositories (selected)

Repository Status Table:

Repository	Branch Protections	Secrets	Variables	Webhooks	Secret Scanning	Private Vulnerability Reporting
.eclipsefdn	✓	●	●	●	✓	●
.github	✗	●	●	●	✓	●
manifests	✗	●	●	●	✓	●
up-android-core	✓	●	●	●	✓	●
up-android-discovery	✓	●	●	●	✓	●
up-android-example	✓	●	●	●	✓	●
up-android-helloworld	✗	●	●	●	✓	●
up-client-android-java	✓	●	●	●	✓	●
up-client-android-kotlin	✗	●	●	●	✓	●
up-client-android-rust	✗	●	●	●	✓	●
up-client-azure-java	✗	●	●	●	✓	●
up-client-mqtt5-python	✗	●	●	●	✓	●

Reach out to us

- Open an issue in the HelpDesk
- Open an issue in your own **.eclipsefdn** repo
- Feature requests at <https://github.com/eclipse-csi/otterdog>
- Documentation available at <https://otterdog.readthedocs.io/en/latest/>
- Discussions at <https://github.com/eclipse-csi/otterdog/discussions> (still empty, will be the canonical place to get help)
- Use the chat service <https://chat.eclipse.org>
- Contact us via GitHub handle **@netomi** or **@mbarbero**

Thank you!