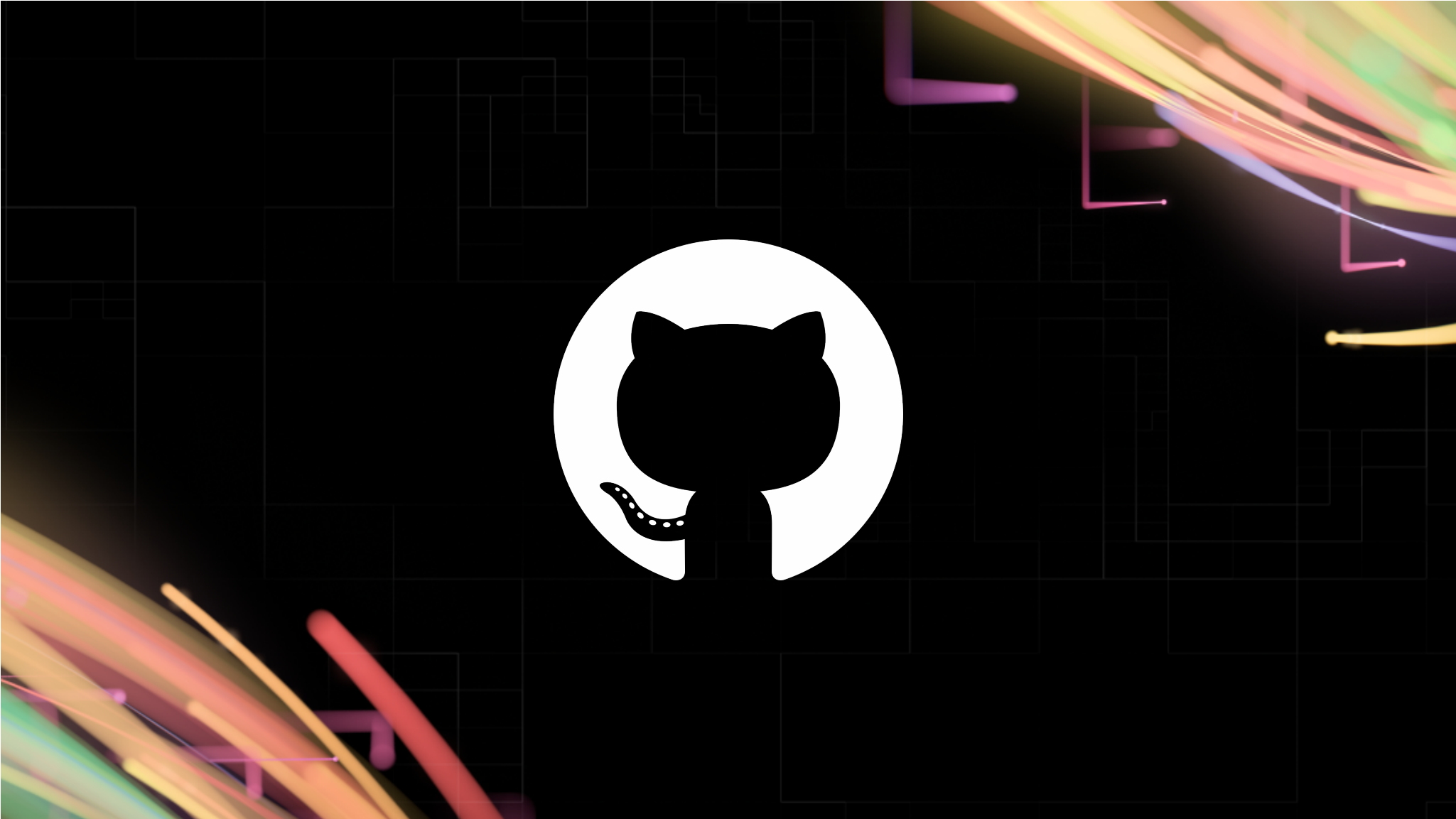
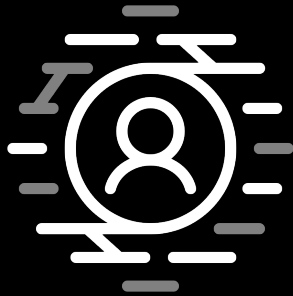


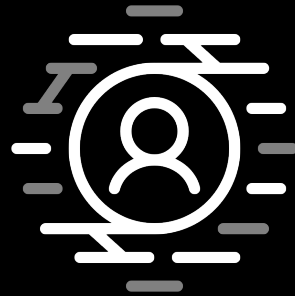
Satellite
INDIA 2021



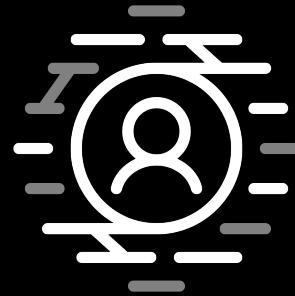
Catchup with GitHub Actions



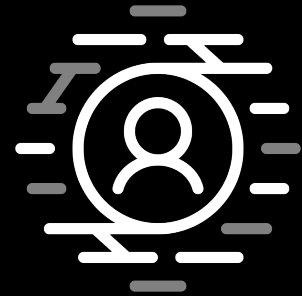
CD



Security



Enterprise
Server



Scale



**GitHub
Actions**

7,000+

**Actions in GitHub
Marketplace**

125m+

Jobs per month

#1

Overall CI on GitHub



Actions GA on GHES 3.0

**Can't I CD with Actions
today?**

Sure, but we think it can be better.

4 (ish) key new capabilities

Visualize

Track progress of complex workflows

The screenshot shows the GitHub Actions interface for the repository `myoctocat`. The `Actions` tab is selected, displaying a workflow run for `request #547 from octocademy/add-octocats-as-code` (Deploy #13). The workflow status is **Success**, triggered via release 33m ago by user `bd0ugie`. The workflow definition in `deploy.yaml` shows a sequence of steps: `Minify & Bundle` (12s) and `Generate Docs` (3s) are grouped together, followed by `Deploy to staging` (2m 5s). The deployment URL is `https://staging.myoctocat.dev`.

Jump to... Pull requests Issues Codespaces Marketplace Explore

/ myoctocat Private

Issues 17 Pull requests Discussions Actions Projects Wiki Security

request #547 from octocademy/add-octocats-as-code Deploy #13

Triggered via release 33m

bd0ugie published Universe2020 f9edf33

Status Success Total d 2m 5

deploy.yaml

on: release

Minify & Bundle 12s

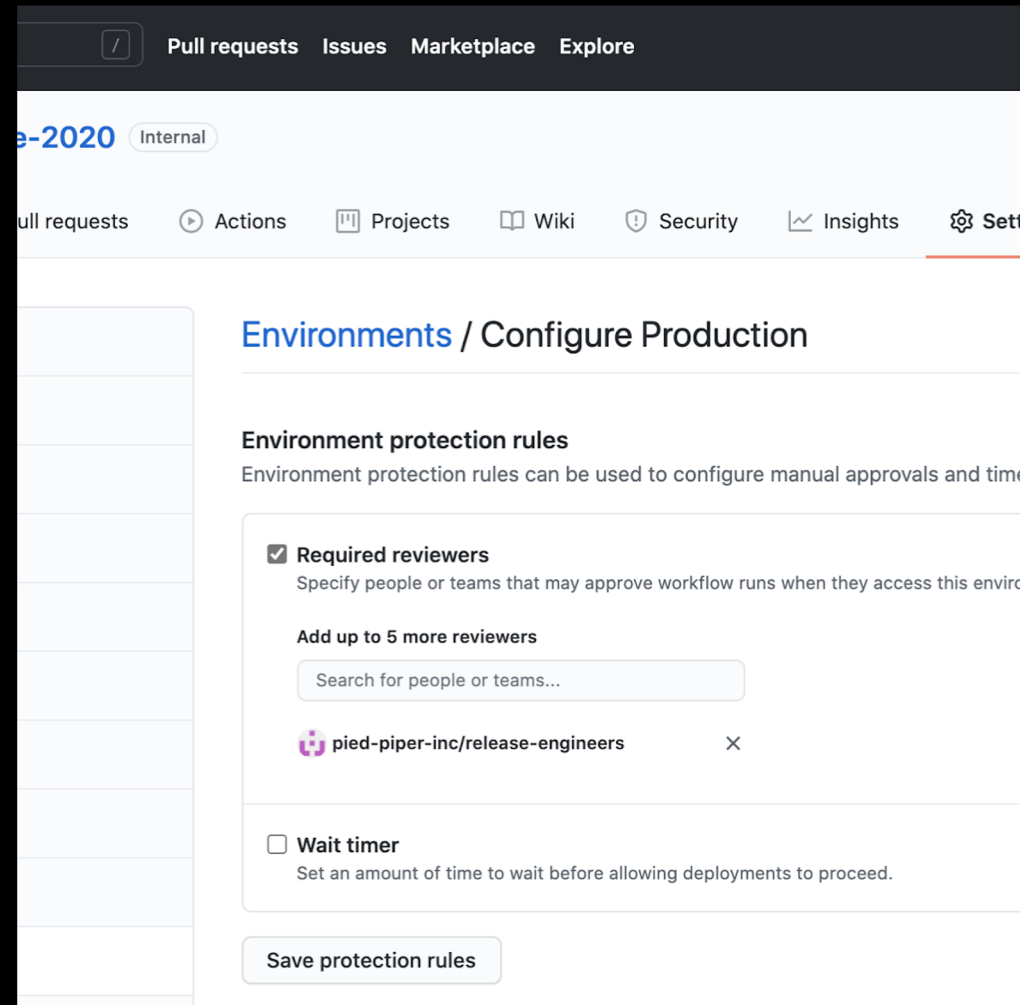
Generate Docs 3s

Deploy to staging

<https://staging.myoctocat.dev>

Protect

Protected Environments prevent unauthorized deployments



The screenshot shows the GitHub web interface for configuring a production environment. The top navigation bar includes links for Pull requests, Issues, Marketplace, and Explore. Below this, the breadcrumb trail shows 'release-2020' with an 'Internal' label. A secondary navigation bar contains links for Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. The main heading is 'Environments / Configure Production'. Under the heading, the section 'Environment protection rules' explains that these rules can be used to configure manual approvals and timeouts. Two rules are visible: 'Required reviewers' (checked) and 'Wait timer' (unchecked). The 'Required reviewers' rule specifies that people or teams must approve workflow runs. A search bar is provided to add reviewers, and one reviewer, 'pied-piper-inc/release-engineers', has been added. The 'Wait timer' rule is currently disabled. A 'Save protection rules' button is at the bottom.

release-2020 Internal

Pull requests Actions Projects Wiki Security Insights Settings

Environments / Configure Production

Environment protection rules

Environment protection rules can be used to configure manual approvals and timeouts.

- ☒ **Required reviewers**
Specify people or teams that may approve workflow runs when they access this environment.
Add up to 5 more reviewers
Search for people or teams...
pied-piper-inc/release-engineers X
- ☐ **Wait timer**
Set an amount of time to wait before allowing deployments to proceed.

Save protection rules

Secrets

Scope secrets to specific Environments

The screenshot shows the GitHub web interface for configuring a production environment. The top navigation bar includes links for Pull requests, Issues, Marketplace, and Explore. Below this, the repository path 'er-inc / universe-2020' is shown with an 'Internal' label. A secondary navigation bar contains icons and labels for Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings (which is highlighted). On the left, a sidebar menu lists various repository features, with 'Environments' and 'Secrets' at the bottom. The main content area is titled 'Environments / Configure Production'. It features a section for 'Environment protection rules' with two checkboxes: 'Required reviewers' and 'Wait timer'. Below these is a 'Save protection rules' button. The 'Environment secrets' section explains that secrets are encrypted environment variables accessible only by GitHub Actions. It displays a single secret named 'DEPLOY_KEY' with a lock icon and an 'Add secret' button with a plus icon. The footer contains copyright information for 2020 GitHub, Inc. and links to Terms, Privacy, Security, Status, Help, Contact GitHub, and Pricing.

h or jump to...

Pull requests Issues Marketplace Explore

er-inc / universe-2020 Internal

Issues Pull requests Actions Projects Wiki Security Insights Settings

Options

Manage access

Security & analysis

Webhooks

Notifications

Integrations

Deploy keys

Autolink references

Actions

Environments

Secrets

Environments / Configure Production

Environment protection rules

Environment protection rules can be used to configure manual approvals and timeouts.

- ☐ **Required reviewers**
Specify people or teams that may approve workflow runs when they access this environment.
- ☐ **Wait timer**
Set an amount of time to wait before allowing deployments to proceed.

Save protection rules

Environment secrets

Secrets are encrypted environment variables. They are accessible only by GitHub Actions in

🔒 DEPLOY_KEY

+ Add secret

© 2020 GitHub, Inc. Terms Privacy Security Status Help Contact GitHub Pricing

Track

Environments track the history of your deployments

The screenshot displays the GitHub Environments interface for a repository named 'niverse-2020'. The top navigation bar includes links for Pull requests, Issues, Codespaces, Marketplace, and Explore. Below this, a secondary navigation bar for the repository shows Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. The main content area is divided into two sections: 'Deployed to production' and 'Deployed to staging'. Each section shows a deployment of the '37cb453' commit by user 'chrispat', with the production deployment being 9 minutes ago and the staging deployment being 10 minutes ago, both marked as 'Active'. Below these sections is an 'Activity log' which provides a detailed history of deployments. It shows a sequence of deployments for both 'production' and 'staging' environments, each associated with the '37cb453' commit and user 'chrispat'. The production deployments are marked as 'Active', while the staging deployments are marked as 'Inactive'.

Deployed to production

37cb453 was deployed by chrispat 9 minutes ago Active

Deployed to staging

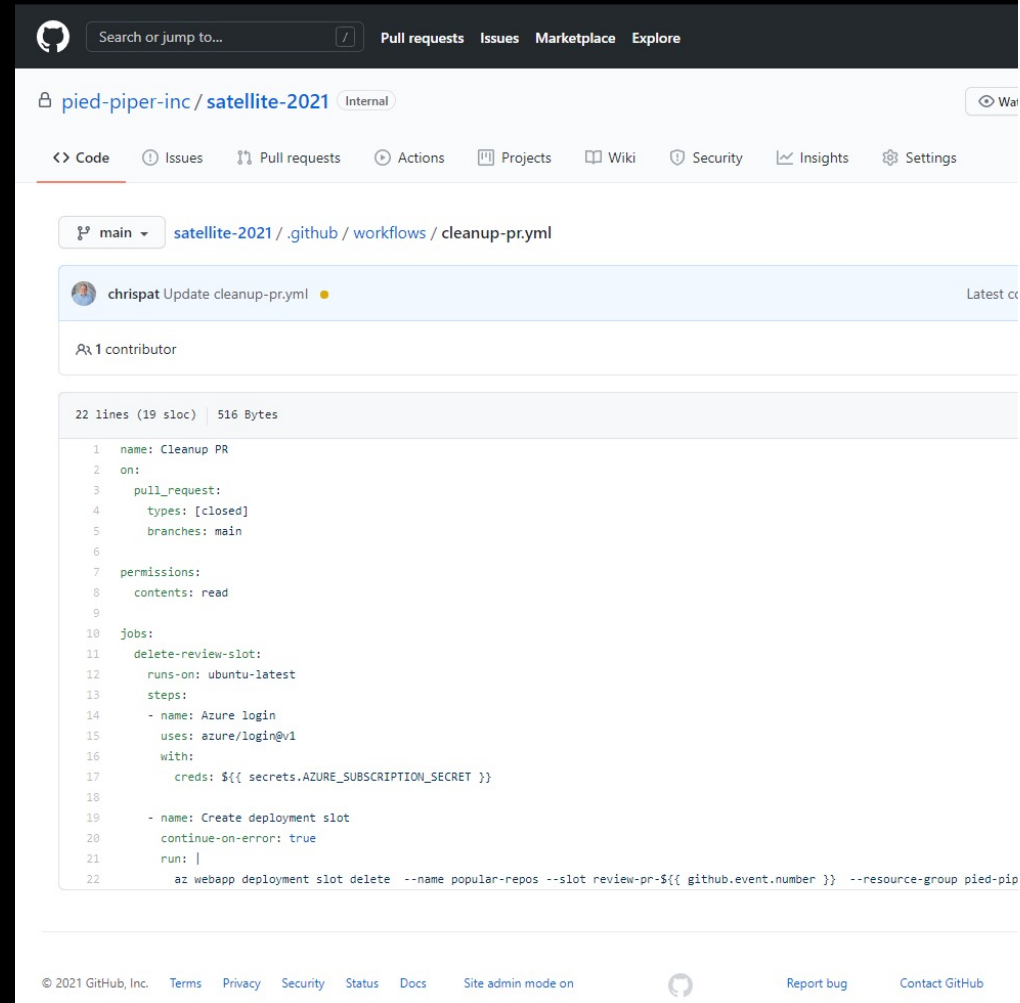
37cb453 was deployed by chrispat 10 minutes ago Active

Activity log

- production at 37cb453
Deployed by chrispat 9 minutes ago Active
- staging at 37cb453
Deployed by chrispat 10 minutes ago Active
- production at 37cb453
Deployed by chrispat 11 minutes ago Inactive
- staging at 37cb453
Deployed by chrispat 12 minutes ago Inactive
- production at 37cb453
Deployed by chrispat 16 minutes ago Inactive

Secure

Secure your workflows



The screenshot displays the GitHub interface for the repository `pied-piper-inc / satellite-2021`. The page shows the `cleanup-pr.yml` workflow file located at `.github / workflows / cleanup-pr.yml`. The workflow is authored by `chrispat` and has 1 contributor. The workflow file content is as follows:

```
1 name: Cleanup PR
2 on:
3   pull_request:
4     types: [closed]
5     branches: main
6
7 permissions:
8   contents: read
9
10 jobs:
11   delete-review-slot:
12     runs-on: ubuntu-latest
13     steps:
14       - name: Azure login
15         uses: azure/login@v1
16         with:
17           creds: ${ secrets.AZURE_SUBSCRIPTION_SECRET }
18
19       - name: Create deployment slot
20         continue-on-error: true
21         run: |
22           az webapp deployment slot delete --name popular-repos --slot review-pr-${ github.event.number } --resource-group pied-pip
```

The footer of the page includes the copyright notice `© 2021 GitHub, Inc.` and links to `Terms`, `Privacy`, `Security`, `Status`, `Docs`, `Site admin mode on`, `Report bug`, and `Contact GitHub`.



Demo

