



## Red Hat Developer Hub 1.2

在 Red Hat Developer Hub 中配置插件





## 法律通告

Copyright © 2024 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux<sup>®</sup> is the registered trademark of Linus Torvalds in the United States and other countries.

Java<sup>®</sup> is a registered trademark of Oracle and/or its affiliates.

XFS<sup>®</sup> is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL<sup>®</sup> is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js<sup>®</sup> is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack<sup>®</sup> Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

## 摘要

Red Hat Developer Hub 是一个用于构建开发人员门户的开发人员平台。您可以在 Developer Hub 中添加和配置插件，以访问各种软件开发工具。

---

# 目录

|  |    |
|--|----|
| 前言 .....                                     | 3  |
| RED HAT DEVELOPER HUB 支持 .....               | 4  |
| 第 1 章 RED HAT DEVELOPER HUB 中的插件 .....       | 5  |
| 第 2 章 动态插件安装 .....                           | 6  |
| 2.1. 查看已安装的插件 .....                          | 6  |
| 2.2. 预安装的动态插件 .....                          | 6  |
| 2.3. 使用 HELM CHART 安装动态插件 .....              | 24 |
| 2.4. 在 AIR-GAPPED 环境中安装外部插件 .....            | 27 |
| 2.5. 使用自定义 NPM REGISTRY 进行动态插件软件包 .....      | 27 |
| 2.6. 动态插件的基本配置 .....                         | 27 |
| 2.7. 安装和配置 ANSIBLE AUTOMATION PLATFORM ..... | 27 |
| 2.8. 安装和配置 KEYCLOAK .....                    | 30 |
| 2.9. 安装和配置 NEXUS 存储库管理器 .....                | 35 |
| 2.10. 安装和配置 TEKTON .....                     | 37 |



## 前言

Red Hat Developer Hub 是一个企业级、集成的开发人员平台，通过插件扩展，有助于减少开发人员的侵权，同时提高其生产力。

## RED HAT DEVELOPER HUB 支持

如果您在执行本文档所述的某个流程时遇到问题，请访问[红帽客户门户](#)。您可以使用红帽客户门户网站进行以下目的：

- 搜索或浏览红帽知识库，了解有关红帽产品的技术支持文章。
- 为红帽全球支持服务(GSS)创建支持问题单。<https://access.redhat.com/support/cases/#/case/new/get-support?caseCreate=true>要创建支持问题单，请选择 **Red Hat Developer Hub**作为产品，然后选择适当的产品版本。



## 第 1 章 RED HAT DEVELOPER HUB 中的插件

Red Hat Developer Hub 应用程序提供带有各种插件的统一平台。使用 Developer Hub 应用程序中的插件生态系统，您可以访问任何类型的开发基础架构或软件开发工具。

Developer Hub 中的插件通过保持整体用户体验的一致性，从而提高生产力并简化开发 workflow。

## 第 2 章 动态插件安装

动态插件支持基于 backend 插件管理器软件包，这是为动态插件软件包扫描配置的根目录( app 配置中的 **dynamicPlugins.rootDirectory** )的服务，并动态加载它们。

您可以使用 Red Hat Developer Hub 预装的动态插件，或者从公共 NPM registry 安装外部动态插件。

### 2.1. 查看已安装的插件

使用动态插件 Info 前端插件，您可以查看当前在 Red Hat Developer Hub 应用程序中安装的插件。此插件默认为启用。

#### 流程

1. 打开 Developer Hub 应用程序，再点 **Administration**。
2. 转至 **Plugins** 选项卡，以查看已安装的插件和相关信息的列表。

### 2.2. 预安装的动态插件

Red Hat Developer Hub 预安装了特定的动态插件。默认情况下禁用需要自定义配置动态插件。

有关此 Developer Hub 发行版本中预装的动态插件的完整列表，请参阅 [Red Hat Developer Hub 中配置插件中的 动态插件支持列表](#)。

在应用程序启动时，对于默认禁用的每个插件，Developer Hub pod 日志中的 **install-dynamic-plugins init 容器** 会显示类似如下的消息：

```
=====  
Skipping disabled dynamic plugin ./dynamic-plugins/dist/backstage-plugin-catalog-backend-module-github-dynamic
```

要启用此插件，请在 Helm Chart 中添加名称相同的软件包，并将 **disabled** 字段中的值改为 'false'。例如：

```
global:  
  dynamic:  
    includes:  
      - dynamic-plugins.default.yaml  
  plugins:  
    - package: ./dynamic-plugins/dist/backstage-plugin-catalog-backend-module-github-dynamic  
      disabled: false
```



#### 注意

插件的默认配置从 **dynamic-plugins.default.yaml** 文件中提取，但您可以使用 **pluginConfig** 条目来覆盖默认配置。

#### 2.2.1. 预安装的动态插件描述和详情



## 重要

红帽产品服务级别协议 (SLA) 不支持技术预览功能，且其功能可能并不完善，因此红帽不建议在生产环境中使用它们。这些技术预览功能可以使用户提早试用新的功能，并有机会在开发阶段提供反馈意见。

有关红帽技术预览功能的更多信息，请参阅 [技术预览功能范围](#)。

有关红帽如何提供对捆绑社区动态插件的支持的更多详细信息，请参阅 [Red Hat Developer Support Policy](#) 页面。

Red Hat Developer Hub 中提供了 60 个插件。如需更多信息，请参阅下表：

表 2.1. 动态插件支持列表

| Name   | plugin                                     | 角色       | Version | 支持等级   | 路径   | 所需的变量  | default  |
|--------|--|----------|---------|--------|--|--|----------|
| 3scale | @janus-idp/backstage-plugin-3scale-backend | 后端       | 1.5.13  | 红帽技术预览 | ./dynamic - plugins/dist/janus-idp-backstage-plugin-3scale-backend-dynamic | <b>THREESCALE_BASE_URL</b><br><b>THREESCALE_ACCESS_TOKEN</b> | Disabled |
| AAP    | @janus-idp/backstage-plugin-aap-backend    | 后端       | 1.6.13  | 红帽技术预览 | ./dynamic - plugins/dist/janus-idp-backstage-plugin-aap-backend-dynamic    | <b>AAP_BASE_URL</b><br><b>AP_AUTH_TOKEN</b>                  | Disabled |
| ACR    | @janus-idp/backstage-plugin-acr            | frontend | 1.4.11  | 红帽技术预览 | ./dynamic - plugins/dist/janus-idp-backstage-plugin-acr-dynamic            |  | Disabled |

| Name    | plugin   | 角色       | Version | 支持等级       | 路径   | 所需的变量  | default  |
|---------|--|----------|---------|------------|--|--|----------|
| 分析供应商分割 | @janus-idp/backstage-plugin-analytics-provider-segment | frontend | 1.4.7   | Production | ./dynamic-plugins/dist/janus-idp-backstage-plugin-analytics-provider-segment-dynamic | <b>SEGMENT_WRITE_KEY</b><br><b>SEGMENT_TEST_MODE</b> | Enabled  |
| Argo CD | @janus-idp/backstage-plugin-argocd                     | frontend | 1.1.6   | Production | ./dynamic-plugins/dist/janus-idp-backstage-plugin-argocd-dynamic                     |  | Disabled |
| Argo CD | @roadiehq/backstage-plugin-argo-cd                     | frontend | 2.6.5   | Production | ./dynamic-plugins/dist/roadiehq-backstage-plugin-argo-cd-dynamic                     |  | Disabled |

| Name    | plugin   | 角色 | Version | 支持等级       | 路径  | 所需的变量  | default  |
|---------|--|----|---------|------------|---|--|----------|
| Argo CD | @roadiehq<br>/backstage-plugin-<br>argo-cd-<br>backend | 后端 | 3.0.2   | Production | ./dynamic-<br>plugins/di-<br>st/roadie-<br>hq-<br>backstage-<br>-plugin-<br>argo-cd-<br>backend-<br>dynamic | <b>ARGOCD_USER<br/>NAME</b><br><br><b>ARGOCD_PASS<br/>WORD</b><br><br><b>ARGOCD_INST<br/>ANCE1_<br/>URL</b><br><br><b>ARGOCD_AUTH<br/>_TOKEN</b><br><br><b>ARGOCD_INST<br/>ANCE2_<br/>URL</b><br><br><b>ARGOCD_AUTH<br/>_TOKEN<br/>2</b> | Disabled |

| Name         | plugin  | 角色       | Version | 支持等级 | 路径  | 所需的变量  | default  |
|--------------|---|----------|---------|------|---|--|----------|
| Argo CD      | @roadiehq/scaffolder-backend-argocd               | 后端       | 1.1.27  | 社区支持 | ./dynamic-<br>plugins/dist/roadiehq-scaffolder-backend-argocd-dynamic               | <b>ARGOCD_USER_NAME</b><br><b>ARGOCD_PASSWORD</b><br><b>ARGOCD_INSTANCE1_URL</b><br><b>ARGOCD_AUTH_TOKEN</b><br><b>ARGOCD_INSTANCE2_URL</b><br><b>ARGOCD_AUTH_TOKEN2</b> | Disabled |
| Azure        | @backstage/plugin-scaffolder-backend-module-azure | 后端       | 0.1.9   | 社区支持 | ./dynamic-<br>plugins/dist/backstage-plugin-scaffolder-backend-module-azure-dynamic |  | Enabled  |
| Azure Devops | @backstage/plugin-azure-devops                    | frontend | 0.4.4   | 社区支持 | ./dynamic-<br>plugins/dist/backstage-plugin-azure-devops-dynamic                    |  | Disabled |

| Name               | plugin  | 角色 | Version | 支持等级 | 路径  | 所需的变量                                      | default  |
|--------------------|---|----|---------|------|---|--|----------|
| Azure Devops       | @backstage/plugin-azure-devops-backend                            | 后端 | 0.6.5   | 社区支持 | ./dynamic-<br>plugins/dist/backstage-plugin-azure-devops-backend-dynamic                            | <b>AZURE_TOKEN</b><br><br><b>AZURE_ORG</b> | Disabled |
| Azure Repositories | @parfuemerie/douglas-scaffolder-backend-module-azure-repositories | 后端 | 0.2.7   | 社区支持 | ./dynamic-<br>plugins/dist/parfuemerie-douglas-scaffolder-backend-module-azure-repositories-dynamic |  | Disabled |
| Bitbucket 云        | @backstage/plugin-catalog-backend-module-bitbucket-cloud          | 后端 | 0.2.4   | 社区支持 | ./dynamic-<br>plugins/dist/backstage-plugin-catalog-backend-module-bitbucket-cloud-dynamic          | <b>BITBUCKET_WORKSPACE</b>                 | Disabled |

| Name          | plugin   | 角色 | Version | 支持等级 | 路径   | 所需的变量                 | default  |
|---------------|--|----|---------|------|--|-----------------------|----------|
| Bitbucket 云   | @backstage/plugin-scaffolder-backend-module-bitbucket-cloud  | 后端 | 0.1.7   | 社区支持 | ./dynamic-plugins/dist/backstage-plugin-scaffolder-backend-module-bitbucket-cloud-dynamic  |                       | Enabled  |
| Bitbucket 服务器 | @backstage/plugin-catalog-backend-module-bitbucket-server    | 后端 | 0.1.31  | 社区支持 | ./dynamic-plugins/dist/backstage-plugin-catalog-backend-module-bitbucket-server-dynamic    | <b>BITBUCKET_HOST</b> | Disabled |
| Bitbucket 服务器 | @backstage/plugin-scaffolder-backend-module-bitbucket-server | 后端 | 0.1.7   | 社区支持 | ./dynamic-plugins/dist/backstage-plugin-scaffolder-backend-module-bitbucket-server-dynamic |                       | Enabled  |



| Name      | plugin   | 角色       | Version | 支持等级 | 路径   | 所需的变量             | default  |
|-----------|--|----------|---------|------|--|-------------------|----------|
| Datadog   | @roadiehq/<br>backstage-plugin-datadog                     | frontend | 2.2.8   | 社区支持 | ./dynamic-<br>plugins/dist/roadiehq-<br>backstage-plugin-datadog-<br>dynamic                         |                   | Disabled |
| Dynatrace | @backstage/plugin-<br>dynatrace                            | frontend | 10.0.4  | 社区支持 | ./dynamic-<br>plugins/dist/backstage-<br>plugin-dynatrace-<br>dynamic                                |                   | Disabled |
| Gerrit    | @backstage/plugin-<br>scaffolder-backend-<br>module-gerrit | 后端       | 0.1.9   | 社区支持 | ./dynamic-<br>plugins/dist/backstage-<br>plugin-scaffolder-<br>backend-module-<br>gerrit-<br>dynamic |                   | Enabled  |
| GitHub    | @backstage/plugin-<br>catalog-backend-<br>module-github    | 后端       | 0.6.0   | 社区支持 | ./dynamic-<br>plugins/dist/backstage-<br>plugin-catalog-<br>backend-module-<br>github-<br>dynamic    | <b>GITHUB_ORG</b> | Disabled |

| Name            | plugin   | 角色       | Version | 支持等级 | 路径   | 所需的变量 | default  |
|-----------------|--|----------|---------|------|--|-------|----------|
| GitHub          | @backstage/plugin-scaffolder-backend-module-github | 后端       | 0.2.7   | 社区支持 | ./dynamic-plugins/dist/backstage-plugin-scaffolder-backend-module-github-dynamic |       | Enabled  |
| GitHub 操作       | @backstage/plugin-github-actions                   | frontend | 0.6.16  | 社区支持 | ./dynamic-plugins/dist/backstage-plugin-github-actions-dynamic                   |       | Disabled |
| GitHub Insights | @roadiehq/backstage-plugin-github-insights         | frontend | 2.3.29  | 社区支持 | ./dynamic-plugins/dist/roadiehq-backstage-plugin-github-insights-dynamic         |       | Disabled |
| GitHub 问题       | @backstage/plugin-github-issues                    | frontend | 0.4.2   | 社区支持 | ./dynamic-plugins/dist/backstage-plugin-github-issues-dynamic                    |       | Disabled |

| Name        | plugin  | 角色       | Version | 支持等级 | 路径  | 所需的变量                                  | default  |
|-------------|---|----------|---------|------|---|--|----------|
| GitHub 机构   | @backstage/plugin-catalog-backend-module-github-org | 后端       | 0.1.12  | 社区支持 | ./dynamic - plugins/dist/backstage-plugin-catalog-backend-module-github-org-dynamic | <b>GITHUB_URL</b><br><b>GITHUB_ORG</b> | Disabled |
| GitHub 拉取请求 | @roadiehq/backstage-plugin-github-pull-requests     | frontend | 2.5.26  | 社区支持 | ./dynamic - plugins/dist/roadiehq-backstage-plugin-github-pull-requests-dynamic     |  | Disabled |
| GitLab      | @immobiliarelabs/backstage-plugin-gitlab            | frontend | 6.5.0   | 社区支持 | ./dynamic - plugins/dist/immobiliarelabs-backstage-plugin-gitlab-dynamic            |  | Disabled |
| GitLab      | @backstage/plugin-catalog-backend-module-gitlab     | 后端       | 0.3.15  | 社区支持 | ./dynamic - plugins/dist/backstage-plugin-catalog-backend-module-gitlab-dynamic     |  | Disabled |

| Name      | plugin  | 角色 | Version | 支持等级 | 路径  | 所需的变量                                     | default  |
|-----------|---|----|---------|------|---|---|----------|
| GitLab    | @immobiliarelabs/backend-plugin-gitlab-backend      | 后端 | 6.5.0   | 社区支持 | ./dynamic - plugins/dist/immobiliarelabs-backend-plugin-gitlab-backend-dynamic      | <b>GITLAB_HOST</b><br><b>GITLAB_TOKEN</b> | Disabled |
| GitLab    | @backstage/plugin-scaffolder-backend-module-gitlab  | 后端 | 0.3.3   | 社区支持 | ./dynamic - plugins/dist/backstage-plugin-scaffolder-backend-module-gitlab-dynamic  |   | Enabled  |
| GitLab 机构 | @backstage/plugin-catalog-backend-module-gitlab-org | 后端 | 0.3.10  | 社区支持 | ./dynamic - plugins/dist/backstage-plugin-catalog-backend-module-gitlab-org-dynamic |   | Disabled |

| Name              | plugin   | 角色       | Version | 支持等级   | 路径   | 所需的变量   | default  |
|-------------------|--|----------|---------|--------|--|---|----------|
| HTTP 请求           | @roadiehq/<br>scaffolder-backend-module-http-request | 后端       | 4.3.2   | 社区支持   | ./dynamic-<br>plugins/dist/roadiehq-scaffolder-backend-module-http-request-dynamic |   | Enabled  |
| Jenkins           | @backstage/plugin-jenkins                            | frontend | 0.9.10  | 社区支持   | ./dynamic-<br>plugins/dist/backstage-plugin-jenkins-dynamic                        |   | Disabled |
| Jenkins           | @backstage/plugin-jenkins-backend                    | 后端       | 0.4.5   | 社区支持   | ./dynamic-<br>plugins/dist/backstage-plugin-jenkins-backend-dynamic                | <b>JENKINS_URL</b><br><br><b>JENKINS_USERNAME</b><br><br><b>JENKINS_TOKEN</b> | Disabled |
| JFrog Artifactory | @janus-idp/backstage-plugin-jfrog-artifactory        | frontend | 1.4.9   | 红帽技术预览 | ./dynamic-<br>plugins/dist/janus-idp-backstage-plugin-jfrog-artifactory-dynamic    |   | Disabled |

| Name       | plugin                                       | 角色       | Version | 支持等级       | 路径   | 所需的变量  | default  |
|------------|--|----------|---------|------------|--|--|----------|
| jira       | @roadiehq/backstage-plugin-jira              | frontend | 2.5.8   | 社区支持       | ./dynamic-plugins/dist/roadiehq-backstage-plugin-jira-dynamic              |  | Disabled |
| Keycloak   | @janus-idp/backstage-plugin-keycloak-backend | 后端       | 1.9.10  | Production | ./dynamic-plugins/dist/janus-idp-backstage-plugin-keycloak-backend-dynamic | <b>KEYCLOAK_BASE_URL</b><br><br><b>KEYCLOAK_LOGIN_REALM</b><br><br><b>KEYCLOAK_REALM</b><br><br><b>KEYCLOAK_CLIENT_ID</b><br><br><b>KEYCLOAK_CLIENT_SECRET</b> | Disabled |
| Kubernetes | @backstage/plugin-kubernetes                 | frontend | 0.11.9  | 社区支持       | ./dynamic-plugins/dist/backstage-plugin-kubernetes-dynamic                 |  | Disabled |

| Name         | plugin   | 角色       | Version | 支持等级       | 路径   | 所需的变量   | default  |
|--------------|--|----------|---------|------------|--|---|----------|
| Kubernetes   | @backstage/plugin-kubernetes-backend                 | 后端       | 0.17.0  | Production | ./dynamic-plugins/dist/backstage-plugin-kubernetes-backend-dynamic                 | <b>K8S_CLUSTER_NAME</b><br><b>K8S_CLUSTER_URL</b><br><b>K8S_CLUSTER_TOKEN</b> | Disabled |
| lighthouse   | @backstage/plugin-lighthouse                         | frontend | 0.4.20  | 社区支持       | ./dynamic-plugins/dist/backstage-plugin-lighthouse-dynamic                         |   | Disabled |
| Nexus 存储库管理器 | @janus-idp/backstage-plugin-nexus-repository-manager | frontend | 1.6.8   | 红帽技术预览     | ./dynamic-plugins/dist/janus-idp-backstage-plugin-nexus-repository-manager-dynamic |   | Disabled |
| OCM          | @janus-idp/backstage-plugin-ocm                      | frontend | 4.1.6   | Production | ./dynamic-plugins/dist/janus-idp-backstage-plugin-ocm-dynamic                      |   | Disabled |

| Name      | plugin  | 角色       | Version | 支持等级       | 路径  | 所需的变量   | default  |
|-----------|---|----------|---------|------------|---|---|----------|
| OCM       | @janus-idp/backstage-plugin-ocm-backend             | 后端       | 4.0.6   | Production | ./dynamic-plugins/dist/janus-idp-backstage-plugin-ocm-backend-dynamic             | <b>OCM_HUB_NAME</b><br><b>OCM_HUB_URL</b><br><b>moc_infra_token</b> | Disabled |
| PagerDuty | @pagerduty/backstage-plugin                         | frontend | 0.12.0  | 社区支持       | ./dynamic-plugins/dist/pagerduty-backstage-plugin-dynamic                         |   | Disabled |
| Quay      | @janus-idp/backstage-plugin-quay                    | frontend | 1.7.6   | Production | ./dynamic-plugins/dist/janus-idp-backstage-plugin-quay-dynamic                    |   | Disabled |
| Quay      | @janus-idp/backstage-scaffolder-backend-module-quay | 后端       | 1.4.10  | Production | ./dynamic-plugins/dist/janus-idp-backstage-scaffolder-backend-module-quay-dynamic |   | Enabled  |



| Name                          | plugin   | 角色       | Version | 支持等级       | 路径   | 所需的变量 | default  |
|-------------------------------|--|----------|---------|------------|--|-------|----------|
| RBAC                          | @janus-idp/backstage-plugin-rbac   | frontend | 1.20.11 | Production | ./dynamic-plugins/dist/janus-idp-backstage-plugin-rbac-dynamic   |       | Disabled |
| regex                         | @janus-idp/backstage-scaffolder-backend-module-regex                             | 后端       | 1.4.10  | Production | ./dynamic-plugins/dist/janus-idp-backstage-scaffolder-backend-module-regex-dynamic                             |       | Enabled  |
| Scaffolder Relation Processor | @janus-idp/backstage-plugin-catalog-backend-module-scaffolder-relation-processor | 后端       | 1.0.1   | 红帽技术预览     | ./dynamic-plugins/dist/janus-idp-backstage-plugin-catalog-backend-module-scaffolder-relation-processor-dynamic |       | Enabled  |
| 安全 Insights                   | @roadiehq/backstage-plugin-security-insights                                     | frontend | 2.3.17  | 社区支持       | ./dynamic-plugins/dist/roadiehq-backstage-plugin-security-insights-dynamic                                     |       | Disabled |

| Name       | plugin  | 角色       | Version | 支持等级   | 路径   | 所需的变量   | default  |
|------------|---|----------|---------|--------|--|---|----------|
| ServiceNow | @janus-idp/backstage-scaffolder-backend-module-servicenow | 后端       | 1.4.12  | 红帽技术预览 | ./dynamic-plugins/dist/janus-idp-backend-module-scaffolder-backend-module-servicenow-dynamic | <b>SERVICE_NOW_BASE_URL</b><br><br><b>SERVICE_NOW_USERNAME</b><br><br><b>SERVICE_NOW_PASSWORD</b> | Disabled |
| SonarQube  | @backstage/plugin-sonarqube                               | frontend | 0.7.17  | 社区支持   | ./dynamic-plugins/dist/backstage-plugin-sonarqube-dynamic                                    |   | Disabled |
| SonarQube  | @backstage/plugin-sonarqube-backend                       | 后端       | 0.2.20  | 社区支持   | ./dynamic-plugins/dist/backstage-plugin-sonarqube-backend-dynamic                            | <b>SONAR_QUBE_URL</b><br><br><b>SONAR_QUBE_TOKEN</b>  | Disabled |
| SonarQube  | @janus-idp/backstage-scaffolder-backend-module-sonarqube  | 后端       | 1.4.10  | 红帽技术预览 | ./dynamic-plugins/dist/janus-idp-backend-module-scaffolder-backend-module-sonarqube-dynamic  |   | Disabled |

| Name       | plugin                               | 角色       | Version | 支持等级       | 路径   | 所需的变量 | default  |
|------------|--------------------------------------|----------|---------|------------|--|-------|----------|
| TechDocs   | @backstage/plugin-techdocs           | frontend | 1.10.4  | Production | ./dynamic-plugins/dist/backstage-plugin-techdocs-dynamic           |       | Enabled  |
| TechDocs   | @backstage/plugin-techdocs-backend   | 后端       | 1.10.4  | Production | ./dynamic-plugins/dist/backstage-plugin-techdocs-backend-dynamic   |       | Enabled  |
| tech Radar | @backstage/plugin-tech-radar         | frontend | 0.7.4   | 社区支持       | ./dynamic-plugins/dist/backstage-plugin-tech-radar-dynamic         |       | Disabled |
| Tekton     | @janus-idp/backstage-plugin-tekton   | frontend | 3.7.5   | Production | ./dynamic-plugins/dist/janus-idp-backstage-plugin-tekton-dynamic   |       | Disabled |
| Topology   | @janus-idp/backstage-plugin-topology | frontend | 1.21.7  | Production | ./dynamic-plugins/dist/janus-idp-backstage-plugin-topology-dynamic |       | Disabled |

| Name  | plugin                                    | 角色 | Version | 支持等级 | 路径                                     | 所需的变量 | default |
|-------|---|----|---------|------|--|-------|---------|
| utils | @roadiehq/scaffolder-backend-module-utils | 后端 | 1.15.3  | 社区支持 | ./dynamic-backend-module-utils-dynamic |       | Enabled |



### 注意

- 要配置 Keycloak，请参阅 [安装和配置 Keycloak](#)。
- 要配置 Techdocs，请参阅 [参考文档](#)。在试验基本设置后，在为生产环境用例部署 TechDocs 时，使用 CI/CD 生成文档和外部云存储。另请参阅这个 [推荐的部署方法](#)。

## 2.3. 使用 HELM CHART 安装动态插件

您可以使用 Helm Chart 部署 Developer Hub 实例，这是灵活的安装方法。使用 Helm Chart，您可以将动态插件加载到 Developer Hub 实例中，而无需重新编译代码或重建容器。

要使用 Helm 在 Developer Hub 中安装动态插件，请在 Helm Chart 中添加以下 **global.dynamic** 参数：

- **plugins**: 用于安装动态插件列表。默认情况下，列表为空。您可以使用以下字段填充 plugins 列表：
  - **软件包**：您要安装动态插件软件包的软件包规格。您可以使用软件包进行本地或外部动态插件安装。对于本地安装，请使用包含动态插件的本地文件夹的路径。对于外部安装，请使用公共 NPM 存储库中的软件包规格。
  - **完整性**（外部软件包需要）：一个完整性校验和，格式为特定于软件包的 `<alg>-<digest>`。支持的算法包括 **sha256**、**SHA384** 和 **sha512**。
  - **pluginConfig**：特定于插件的 **app-config** YAML 片段。如需更多信息，请参阅插件配置。
  - **disabled**：如果设为 **true**，则禁用动态插件。默认值：**false**。
- **包括**：使用相同语法的 YAML 文件列表。



### 注意

包含文件中的 **plugins** 列表与主 Helm 值中的 **plugins** 列表合并。如果两个插件列表中提到了 **插件** 软件包，则主 Helm 值中的 **plugins** 字段会覆盖 **includes** 文件中的 **plugins** 字段。默认配置包括 **dynamic-plugins.default.yaml** 文件，其中包含 Developer Hub 中预安装的所有动态插件，无论是默认启用还是禁用。

### 2.3.1. 获取完整性 checksum

要获取完整性 checksum，请输入以下命令：

```
npm view <package name>@<version> dist.integrity
```

### 2.3.2. 动态插件安装的 Helm Chart 配置示例

以下示例演示了如何为特定类型的动态插件安装配置 Helm Chart。

**当外部插件需要特定的 app-config 时，配置本地插件和外部插件**

```
global:
  dynamic:
    plugins:
      - package: <alocal package-spec used by npm pack>
      - package: <external package-spec used by npm pack>
      integrity: sha512-<some hash>
      pluginConfig: ...
```

**从包含的文件禁用插件**

```
global:
  dynamic:
    includes:
      - dynamic-plugins.default.yaml
    plugins:
      - package: <some imported plugins listed in dynamic-plugins.default.yaml>
      disabled: true
```

**从包含的文件启用插件**

```
global:
  dynamic:
    includes:
      - dynamic-plugins.default.yaml
    plugins:
      - package: <some imported plugins listed in dynamic-plugins.custom.yaml>
      disabled: false
```

**启用在包含的文件中禁用的插件**

```
global:
  dynamic:
    includes:
      - dynamic-plugins.default.yaml
    plugins:
      - package: <some imported plugins listed in dynamic-plugins.custom.yaml>
      disabled: false
```

### 2.3.3. 使用 Helm Chart 安装外部动态插件

NPM registry 包含可用于演示目的的外部动态插件。例如，NPMJS 存储库中的 **janus-idp** 组织提供了以下社区插件：

- 通知(frontend 和 backend)
- Kubernetes 操作(scaffolder 操作)

要安装 Notifications 和 Kubernetes 操作插件，请在 **global.dynamic.plugins** 列表中的 Helm Chart 值中包含它们，如下例所示：

```
global:
  dynamic:
    plugins:
      - package: '@janus-idp/plugin-notifications-backend-dynamic@1.3.6'
        # Integrity can be found at https://registry.npmjs.org/@janus-idp/plugin-notifications-backend-dynamic
        integrity: 'sha512-
Qd8pniy1yRx+x7LnwjzQ6k9zP+C1yex24MaCcx7dGDPT/XbTokwoSZr4baSSn8jUA6P45NUUevu1d629
mG4JGQ=='
      - package: '@janus-idp/plugin-notifications@1.1.12'
        # https://registry.npmjs.org/@janus-idp/plugin-notifications

        integrity: 'sha512-
GCdEuHRQek3ay428C8C4wWgxjNpNwCXgldFbUUFGCLLkBFSSaOEw+XaBvWaBGtQ5BLgE3jQEUx
a+422uzSYC5oQ=='
      pluginConfig:
        dynamicPlugins:
          frontend:
            janus-idp.backstage-plugin-notifications:
              applcons:
                - name: notificationsIcon
                  module: NotificationsPlugin
                  importName: NotificationsActiveIcon
            dynamicRoutes:
              - path: /notifications
                importName: NotificationsPage
                module: NotificationsPlugin
              menuItem:
                icon: notificationsIcon
                text: Notifications
              config:
                pollingIntervalMs: 5000
      - package: '@janus-idp/backstage-scaffolder-backend-module-kubernetes-dynamic@1.3.5'
        # https://registry.npmjs.org/@janus-idp/backstage-scaffolder-backend-module-kubernetes-dynamic
        integrity: 'sha512-
19ie+FM3QHxWYPyYzE0uNdI5K8M4vGZ0SPeeTw85XPROY1DrIY7rMm2G0XT85L0ZmntHVwc9qW
+SbHolPg/qRA=='
      proxy:
        endpoints:
          /explore-backend-completed:
            target: 'http://localhost:7017'
      - package: '@dfatwork-pkgs/search-backend-module-explore-wrapped-dynamic@0.1.3-next.1'
        # https://registry.npmjs.org/@dfatwork-pkgs/search-backend-module-explore-wrapped-dynamic
        integrity: 'sha512-
```

```
mv6LS8UOve+eumoMCVypGcd7b/L36IH2z11tGKVrt+m65VzQI4FgAJr9kNCrjUZPMyh36KVGIjYqsu9+
kgzH5A==
- package: '@dfatwork-pkgs/plugin-catalog-backend-module-test-dynamic@0.0.0'
  # https://registry.npmjs.org/@dfatwork-pkgs/plugin-catalog-backend-module-test-dynamic
  integrity: 'sha512-
YsrZMThxJk7cYJU9FtAcsTCx9ICChpytK254TfGb3iMAYQyVcZnr5AA/AU+hezFnXLsr6gj8PP7z/mCZie
uuDA=='
```

## 2.4. 在 AIR-GAPPED 环境中安装外部插件

您可以通过设置自定义 NPM registry，在 air-gapped 环境中安装外部插件。要为动态插件软件包配置 NPM registry URL 和身份验证信息，[请参阅使用自定义 NPM registry 进行动态插件软件包](#)。

## 2.5. 使用自定义 NPM REGISTRY 进行动态插件软件包

您可以使用 Helm Chart 为动态插件软件包配置 NPM registry URL 和身份验证信息。对于通过 **npm pack** 获取的动态插件软件包，您可以使用 **.npmrc** 文件。

使用 Helm Chart，使用以下内容创建名为 **dynamic-plugins-npmrc** 的 secret，将 **.npmrc** 文件添加到 NPM registry 中：

```
apiVersion: v1
kind: Secret
metadata:
  name: dynamic-plugins-npmrc
type: Opaque
stringData:
  .npmrc: |
    registry=<registry-url>
    //<registry-url>:_authToken=<auth-token>
    ...
```

## 2.6. 动态插件的基本配置

有些动态插件需要设置环境变量。如果没有设置强制环境变量，并且插件已启用，则应用程序在启动时可能会失败。

每个插件的必要环境变量列在 [动态插件支持列表中](#)。



### 注意

ZiB-bomb 检测在安装一些包含大型文件的动态插件时，如果安装脚本认为软件包存档成为 Zib-Bomb，安装会失败。

要增加软件包存档中文件的最大允许大小，您可以从默认的 **20000000** 字节增加部署 **install-dynamic-plugins initContainer** 的 **MAX\_ENTRY\_SIZE** 环境值。

## 2.7. 安装和配置 ANSIBLE AUTOMATION PLATFORM

Ansible Automation Platform (AAP) 插件同步可访问的模板，包括来自 AAP 的作业模板和工作流作业模板到 Developer Hub 目录。



## 重要

Ansible Automation Platform 插件只是一个技术预览功能。

红帽产品服务级别协议 (SLA) 不支持技术预览功能，且其功能可能并不完善，因此红帽不建议在生产环境中使用它们。这些技术预览功能可以使用户提早试用新的功能，并有机会在开发阶段提供反馈意见。

有关红帽技术预览功能的更多信息，请参阅 [技术预览功能范围](#)。

有关红帽如何提供对捆绑社区动态插件的支持的更多详细信息，请参阅 [Red Hat Developer Support Policy](#) 页面。

## 2.7.1. 对于管理员

### 2.7.1.1. 安装和配置 AAP 后端插件

AAP 后端插件允许您使用 Developer Hub 中的 **app-config.yaml** 配置文件配置一个或多个供应商。

#### 先决条件

- 您的 Developer Hub 应用程序已安装并运行。
- 您已在 Ansible Automation Platform 中创建帐户。

#### 安装

AAP 后端插件在 Developer Hub 中预加载，带有基本配置属性。要启用它，将 **disabled** 属性设置为 **false**，如下所示：

```
global:
  dynamic:
    includes:
      - dynamic-plugins.default.yaml
    plugins:
      - package: ./dynamic-plugins/dist/janus-idp-backstage-plugin-aap-backend-dynamic
        disabled: false
```

#### 基本配置

要启用 AAP 插件，您必须设置以下环境变量：

- **AAP\_BASE\_URL**: 服务的基本 URL
- **AAP\_AUTH\_TOKEN**: 服务的身份验证令牌

#### 高级配置

1. 您可以使用 **aap** 标记来配置 Developer Hub 的 **app-config.yaml** 文件，如下所示：

```
catalog:
  providers:
    aap:
      dev:
        baseUrl: ${AAP_BASE_URL}
```



```

authorization: 'Bearer ${AAP_AUTH_TOKEN}'
owner: <owner>
system: <system>
schedule: # optional; same options as in TaskScheduleDefinition
# supports cron, ISO duration, "human duration" as used in code
frequency: { minutes: 1 }
# supports ISO duration, "human duration" as used in code
timeout: { minutes: 1 }

```

### 2.7.1.2. AAP 后端插件的日志行故障排除

启动 Developer Hub 应用程序时，您可以看到以下日志行：

```

[1] 2023-02-13T15:26:09.356Z catalog info Discovered ResourceEntity API type=plugin
target=AapResourceEntityProvider:dev
[1] 2023-02-13T15:26:09.423Z catalog info Discovered ResourceEntity Red Hat Event (DEV, v1.2.0)
type=plugin target=AapResourceEntityProvider:dev
[1] 2023-02-13T15:26:09.620Z catalog info Discovered ResourceEntity Red Hat Event (TEST, v1.1.1)
type=plugin target=AapResourceEntityProvider:dev
[1] 2023-02-13T15:26:09.819Z catalog info Discovered ResourceEntity Red Hat Event (PROD,
v1.1.1) type=plugin target=AapResourceEntityProvider:dev
[1] 2023-02-13T15:26:09.819Z catalog info Applying the mutation with 3 entities type=plugin
target=AapResourceEntityProvider:dev

```

## 2.7.2. 对于用户

### 2.7.2.1. 从 Developer Hub 中的 AAP 访问模板

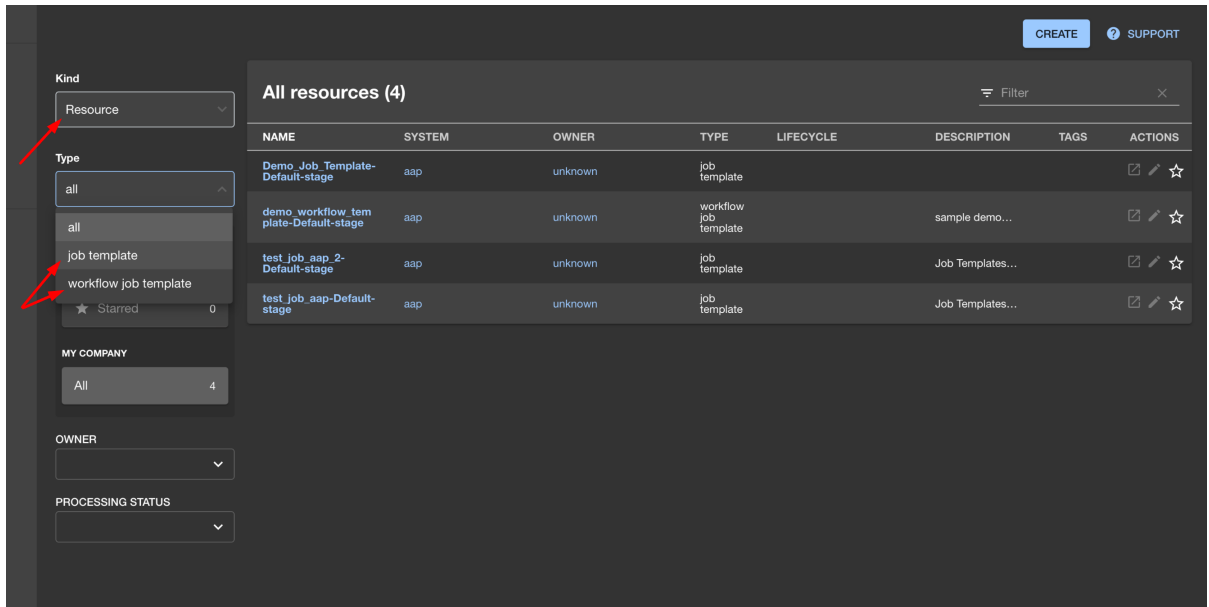
当您成功配置了 AAP 后端插件时，它会同步模板，包括 AAP 中的作业模板和工作流作业模板，并将其显示在 Developer Hub Catalog 页面中作为 Resources。

#### 先决条件

- 您的 Developer Hub 应用程序已安装并运行。
- 已安装 AAP 后端插件。有关安装和配置说明，请参阅 [第 2.7.1.1 节“安装和配置 AAP 后端插件”](#)。

#### 流程

1. 打开 Developer Hub 应用程序，再打开 **Catalog** 页面。
2. 从页面左侧的 **Type** 下拉菜单中选择 **Kind** 下拉菜单和 **作业模板** 或 **工作流作业模板** 的资源。

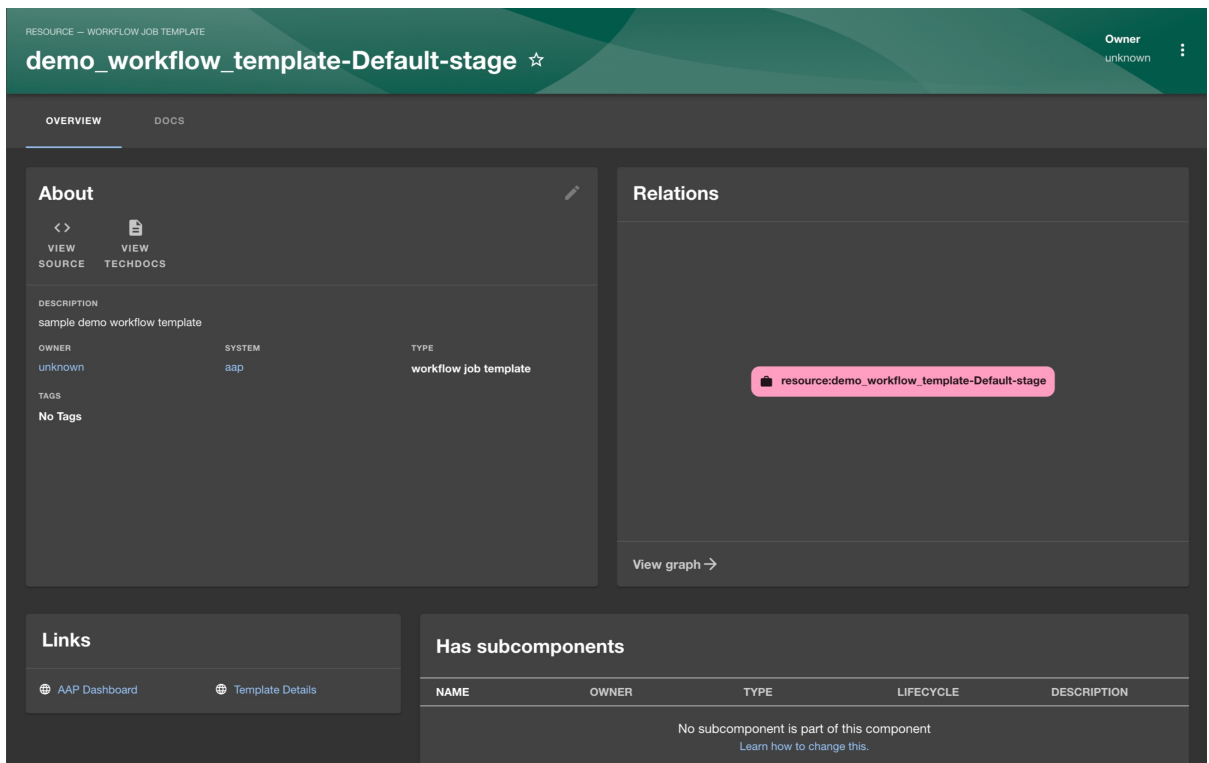


页面中会出现来自 AAP 的所有可用模板的列表。

3. 从列表选择一个模板。

**OVERVIEW** 选项卡会出现包含不同卡，例如：

- **关于**：提供有关模板的详细信息。
- **关系**：显示模板和相关的各个方面的视觉表示。
- **链接**：包含到 AAP 仪表板的链接和模板的详细信息页面。
- **具有子组件**：显示关联的子组件列表。



## 2.8. 安装和配置 KEYCLOAK

将 Keycloak 后端插件集成到 Developer Hub 中，具有以下功能：

- 同步域中的 Keycloak 用户。
- 在域中同步 Keycloak 组及其用户。

## 2.8.1. 对于管理员

### 2.8.1.1. 安装

Keycloak 插件在 Developer Hub 中预加载，带有基本配置属性。要启用它，将 **disabled** 属性设置为 **false**，如下所示：

```
global:
  dynamic:
    includes:
      - dynamic-plugins.default.yaml
    plugins:
      - package: ./dynamic-plugins/dist/janus-idp-backstage-plugin-keycloak-backend-dynamic
        disabled: false
```

### 2.8.1.2. 基本配置

要启用 Keycloak 插件，您必须设置以下环境变量：

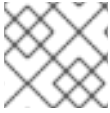
- **KEYCLOAK\_BASE\_URL**
- **KEYCLOAK\_LOGIN\_REALM**
- **KEYCLOAK\_REALM**
- **KEYCLOAK\_CLIENT\_ID**
- **KEYCLOAK\_CLIENT\_SECRET**

### 2.8.1.3. 高级配置

#### 调度配置

您可以在 **app-config.yaml** 文件中配置调度，如下所示：

```
catalog:
  providers:
    keycloakOrg:
      default:
        # ...
        # highlight-add-start
        schedule: # optional; same options as in TaskScheduleDefinition
          # supports cron, ISO duration, "human duration" as used in code
        frequency: { minutes: 1 }
          # supports ISO duration, "human duration" as used in code
        timeout: { minutes: 1 }
        initialDelay: { seconds: 15 }
        # highlight-add-end
```



## 注意

如果您在 **app-config.yaml** 文件中对调度进行了任何更改，则重启以应用更改。

## Keycloak 查询参数

您可以覆盖 **app-config.yaml** 文件中的默认 Keycloak 查询参数，如下所示：

```
catalog:
  providers:
    keycloakOrg:
      default:
        # ...
        # highlight-add-start
        userQuerySize: 500 # Optional
        groupQuerySize: 250 # Optional
        # highlight-add-end
```

使用 Keycloak API 启用 Developer Hub 和 Keycloak 之间的通信。用户名和密码或客户端凭证是支持的身份验证方法。

下表描述了您可以配置在 **app-config.yaml** 文件中的 **catalog.providers.keycloakOrg.<ENVIRONMENT\_NAME>** 对象下启用插件的参数：

| Name                 | 描述  | 默认值           | 必填                      |
|----------------------|---|---------------|-------------------------|
| <b>baseUrl</b>       | Keycloak 服务器的位置，如 <b>https://localhost:8443/auth</b> 。请注意，Keycloak 的较新版本省略了 <b>/auth</b> 上下文路径。 | ""            | 是                       |
| <b>realm</b>         | 要同步的域   | <b>master</b> | 否                       |
| <b>loginRealm</b>    | 用于验证的域  | <b>master</b> | 否                       |
| <b>username</b>      | 进行身份验证的用户名  | ""            | 如果使用基于密码的身份验证，则为        |
| <b>password</b>      | 进行身份验证的密码   | ""            | 如果使用基于密码的身份验证，则为        |
| <b>clientId</b>      | 要进行身份验证的客户端 ID  | ""            | 如果使用基于客户端凭证的身份验证，则为 yes |
| <b>clientSecret</b>  | 要进行身份验证的客户端 Secret  | ""            | 如果使用基于客户端凭证的身份验证，则为 yes |
| <b>userQuerySize</b> | 一次查询的用户数  | <b>100</b>    | 否                       |

| Name                  | 描述      | 默认值        | 必填 |
|-----------------------|---------|------------|----|
| <b>groupQuerySize</b> | 一次查询的组数 | <b>100</b> | 否  |

在使用客户端凭据时，必须将访问类型设为 **机密**，并且必须启用服务帐户。您还必须从 **realm-management** 客户端角色中添加以下角色：

- **query-groups**
- **query-users**
- **view-users**

#### 2.8.1.4. 限制

如果您有自签名或企业证书问题，您可以在启动 Developer Hub 前设置以下环境变量：

**NODE\_TLS\_REJECT\_UNAUTHORIZED=0**



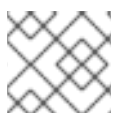
#### 注意

不建议设置环境变量的解决方案。

### 2.8.2. 对于用户

#### 2.8.2.1. 使用 Keycloak 插件在 Developer Hub 中导入用户和组

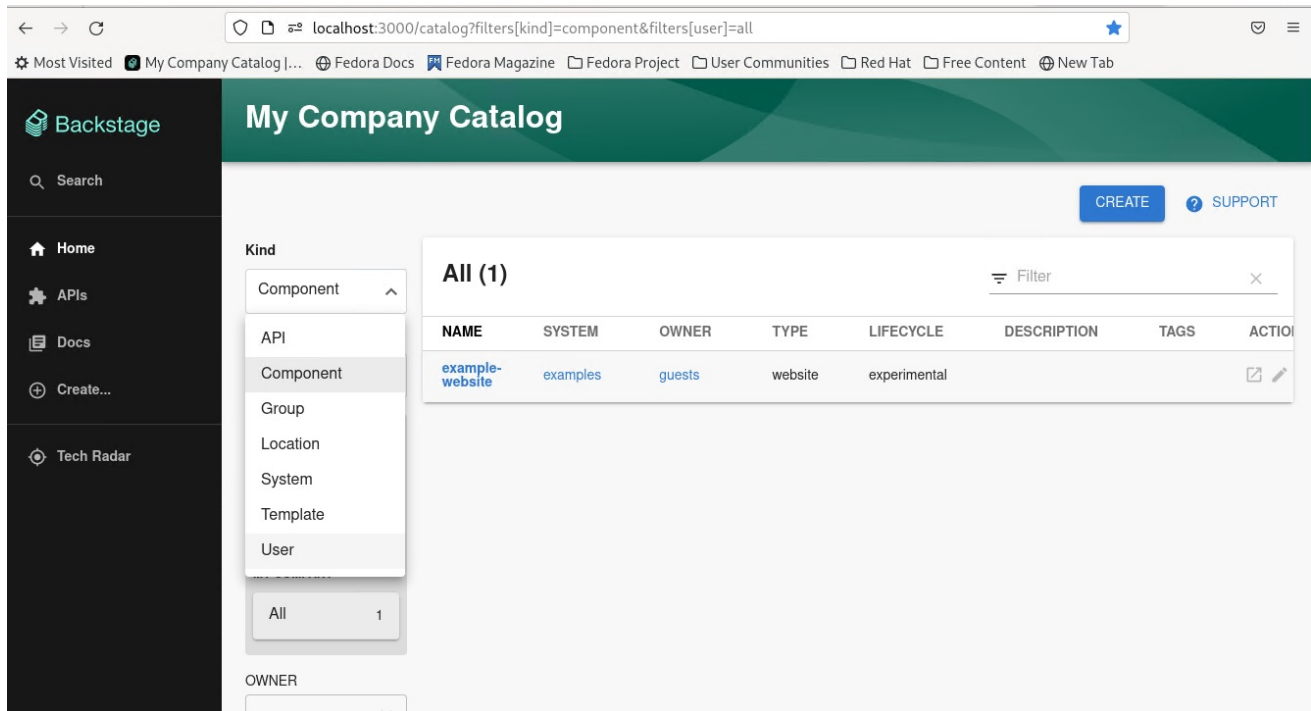
成功配置插件后，插件会在每次启动时都会导入用户和组。



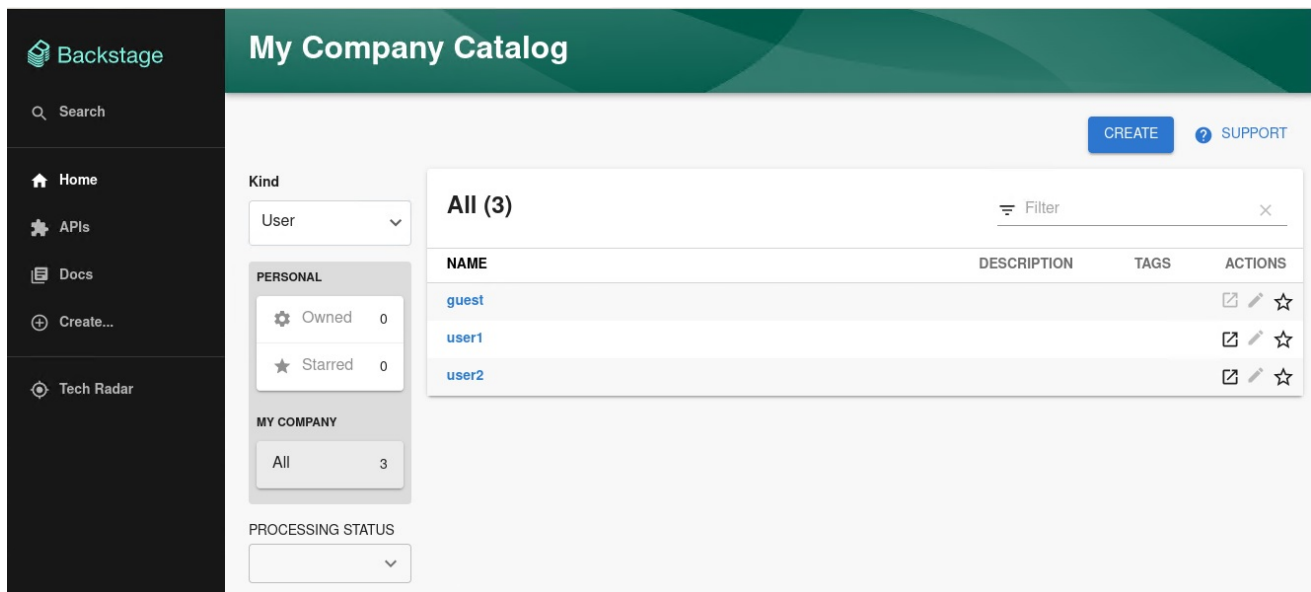
#### 注意

如果您设置了调度，则用户和组也会被导入。

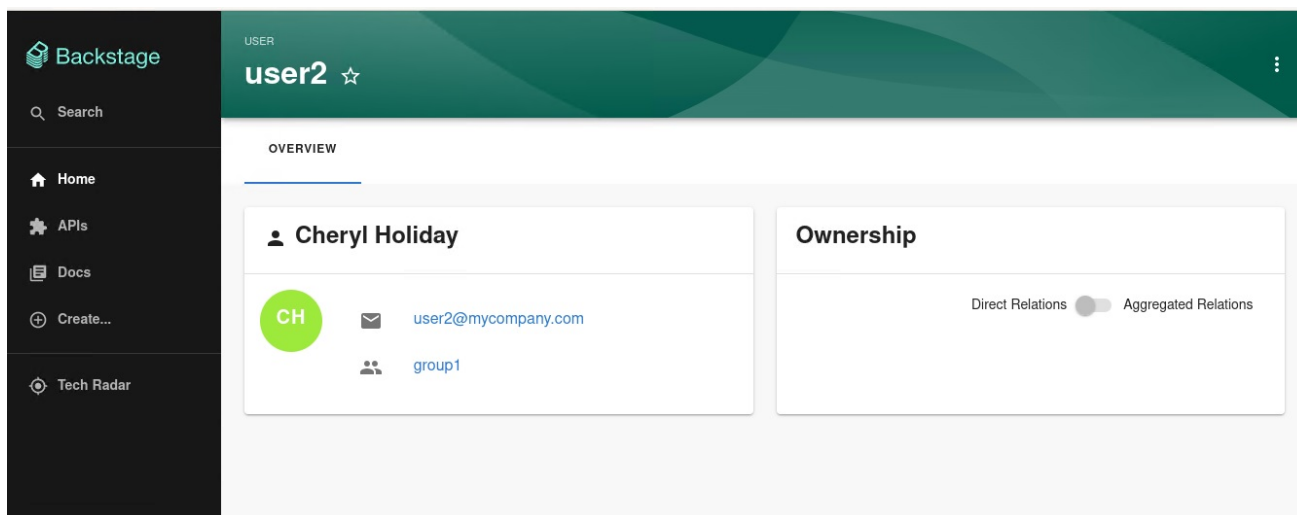
第一次导入完成后，您可以从目录页面中选择 **User** 来列出用户：



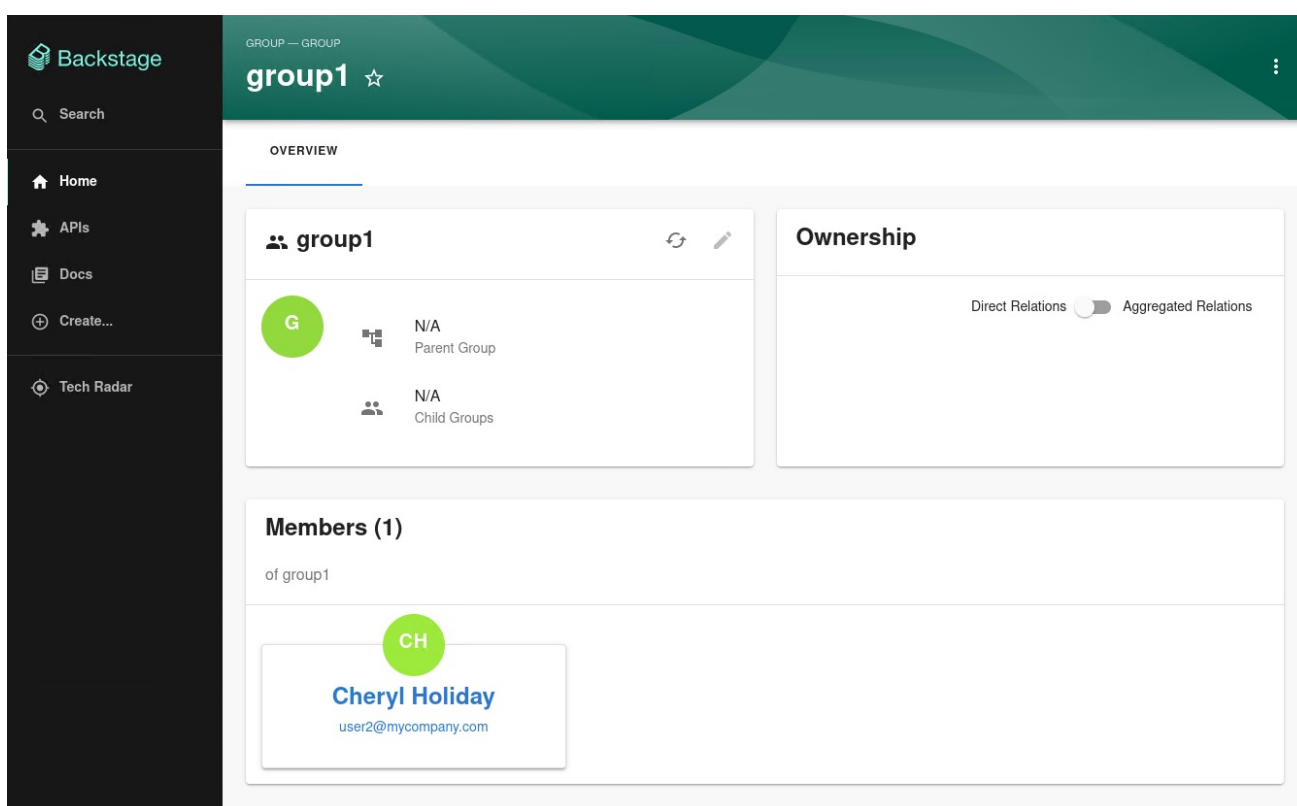
您可以在页面中查看用户列表：



当您选择一个用户时，您可以看到从 Keycloak 导入的信息：



您还可以选择一个组，查看列表，并为组选择或查看从 Keycloak 导入的信息：



## 2.9. 安装和配置 NEXUS 存储库管理器

Nexus Repository Manager 插件显示有关 Developer Hub 应用中构建工件的信息。构建工件在 Nexus Repository Manager 中提供。



## 重要

Nexus Repository Manager 插件只是一个技术预览功能。

红帽产品服务级别协议 (SLA) 不支持技术预览功能，且其功能可能并不完善，因此红帽不建议在生产环境中使用它们。这些技术预览功能可以使用户提早试用新的功能，并有机会在开发阶段提供反馈意见。

有关红帽技术预览功能的更多信息，请参阅 [技术预览功能范围](#)。

有关红帽如何提供对捆绑社区动态插件的支持的更多详细信息，请参阅 [Red Hat Developer Support Policy](#) 页面。

## 2.9.1. 对于管理员

### 2.9.1.1. 安装和配置 Nexus Repository Manager 插件

#### 安装

Nexus Repository Manager 插件在 Developer Hub 中预加载，带有基本配置属性。要启用它，将 `disabled` 属性设置为 `false`，如下所示：

```
global:
  dynamic:
    includes:
      - dynamic-plugins.default.yaml
    plugins:
      - package: ./dynamic-plugins/dist/janus-idp-backstage-plugin-nexus-repository-manager
        disabled: false
```

#### 配置

1. 在 `app-config.yaml` 文件中将代理设置为所需的 Nexus Repository Manager 服务器，如下所示：

```
proxy:
  '/nexus-repository-manager':
    target: 'https://<NEXUS_REPOSITORY_MANAGER_URL>'
    headers:
      X-Requested-With: 'XMLHttpRequest'
      # Uncomment the following line to access a private Nexus Repository Manager using a
      # token
      # Authorization: 'Bearer <YOUR TOKEN>'
    changeOrigin: true
    # Change to "false" in case of using self hosted Nexus Repository Manager instance with a
    # self-signed certificate
    secure: true
```

2. 可选：更改 Nexus Repository Manager 代理的基本 URL，如下所示：

```
nexusRepositoryManager:
  # default path is `nexus-repository-manager`
  proxyPath: /custom-path
```



3. 可选：启用以下实验性注解：

```
nexusRepositoryManager:
  experimentalAnnotations: true
```

4. 使用以下注解给实体添加注解：

```
metadata:
  annotations:
    # insert the chosen annotations here
    # example
    nexus-repository-manager/docker.image-name: '<ORGANIZATION>/<REPOSITORY>',
```

## 2.9.2. 对于用户

### 2.9.2.1. 在 Developer Hub 中使用 Nexus Repository Manager 插件

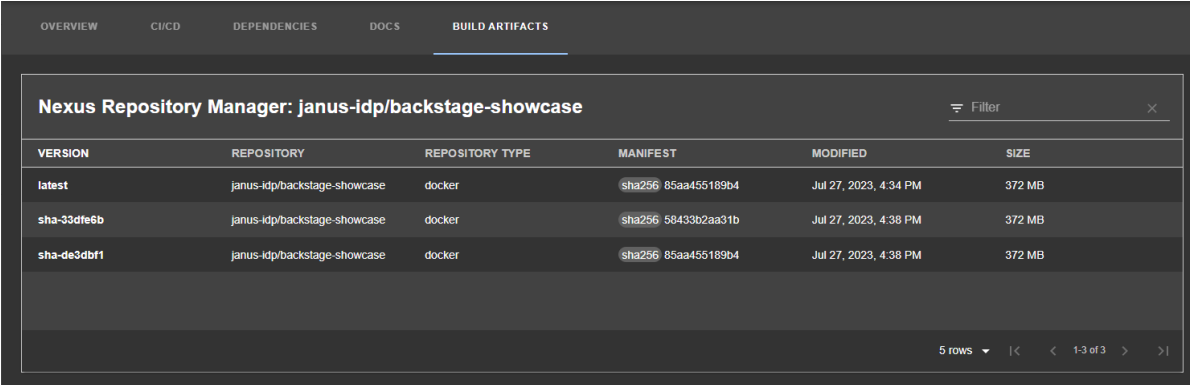
Nexus Repository Manager 是一个前端插件，可让您查看有关构建工件的信息。

#### 先决条件

- 您的 Developer Hub 应用程序已安装并运行。
- 已安装 Nexus Repository Manager 插件。有关安装过程，请参阅 [第 2.9.1.1 节“安装和配置 Nexus Repository Manager 插件”](#)。

#### 流程

1. 打开 Developer Hub 应用程序，从 **Catalog** 页面中选择一个组件。
2. 转至 **BUILD ARTIFACTS** 选项卡。  
**BUILD ARTIFACTS** 选项卡包含构建工件和相关信息的列表，如 **VERSION**、**REPOSITORY TYPE**、**REPOSITORY TYPE**、**MANIFEST**、**MODIFIED** 和 **SIZE**。



| VERSION     | REPOSITORY                   | REPOSITORY TYPE | MANIFEST            | MODIFIED              | SIZE   |
|-------------|------------------------------|-----------------|---------------------|-----------------------|--------|
| latest      | janus-idp/backstage-showcase | docker          | sha256 85aa455189b4 | Jul 27, 2023, 4:34 PM | 372 MB |
| sha-33dfe6b | janus-idp/backstage-showcase | docker          | sha256 58433b2aa31b | Jul 27, 2023, 4:38 PM | 372 MB |
| sha-de3dbf1 | janus-idp/backstage-showcase | docker          | sha256 85aa455189b4 | Jul 27, 2023, 4:38 PM | 372 MB |

## 2.10. 安装和配置 TEKTON

您可以使用 Tekton 插件来可视化 CI/CD 管道在 Kubernetes 或 OpenShift 集群上运行的结果。该插件允许用户可视化查看管道中所有相关任务的高级别状态。

### 2.10.1. 对于管理员

### 2.10.1.1. 安装

#### 先决条件

- 您已安装并配置了 `@backstage/plugin-kubernetes` 和 `@backstage/plugin-kubernetes-backend` 动态插件。
- 您已将 Kubernetes 插件配置为使用 **ServiceAccount** 连接到集群。
- 对于访问集群的 **ServiceAccount**，必须为自定义资源(PipelineRuns 和 TaskRuns)授予 **ClusterRole**。



#### 注意

如果您配置了 RHDH Kubernetes 插件，则必须授予 **ClusterRole**。

- 要查看 pod 日志，您已为 `pod/log` 授予权限。
- 您可以使用以下代码为自定义资源和 pod 日志授予 **ClusterRole**：

```
kubernetes:
  ...
  customResources:
    - group: 'tekton.dev'
      apiVersion: 'v1'
      plural: 'pipelineruns'
    - group: 'tekton.dev'
      apiVersion: 'v1'

  ...
  apiVersion: rbac.authorization.k8s.io/v1
  kind: ClusterRole
  metadata:
    name: backstage-read-only
  rules:
    - apiGroups:
      - ""
      resources:
        - pods/log
      verbs:
        - get
        - list
        - watch
    - apiGroups:
      - tekton.dev
      resources:
        - pipelineruns
        - taskruns
      verbs:
        - get
        - list
```

您可以将准备的清单用于只读 **ClusterRole**，它为 Kubernetes 插件和 Tekton 插件提供访问权限。

- 在实体的 **catalog-info.yaml** 文件中添加以下注解，以识别实体是否包含 Kubernetes 资源：

```

annotations:
  ...

  backstage.io/kubernetes-id: <BACKSTAGE_ENTITY_NAME>

```

- 您还可以添加 **backstage.io/kubernetes-namespace** 注解来使用定义的命名空间标识 Kubernetes 资源。

```

annotations:
  ...

  backstage.io/kubernetes-namespace: <RESOURCE_NS>

```

- 将以下注解添加到实体的 **catalog-info.yaml** 文件中，以便在 RHDH 中启用 Tekton 相关功能。注解值标识 RHDH 实体的名称：

```

annotations:
  ...

  janus-idp.io/tekton : <BACKSTAGE_ENTITY_NAME>

```

- 添加自定义标签选择器，它 RHDH 用来查找 Kubernetes 资源。标签选择器优先于 ID 注解。

```

annotations:
  ...

  backstage.io/kubernetes-label-selector: 'app=my-app,component=front-end'

```

- 在资源中添加以下标签，以便 Kubernetes 插件从请求的实体获取 Kubernetes 资源：

```

labels:
  ...

  backstage.io/kubernetes-id: <BACKSTAGE_ENTITY_NAME>

```



### 注意

使用标签选择器时，资源上必须存在上述标签。

## 流程

- Tekton 插件在 RHDH 中预加载，带有基本配置属性。要启用它，将 `disabled` 属性设置为 `false`，如下所示：

```

global:
  dynamic:
    includes:
      - dynamic-plugins.default.yaml

```

plugins:

- package: ../dynamic-plugins/dist/janus-idp-backstage-plugin-tekton
- disabled: false

## 2.10.2. 对于用户

### 2.10.2.1. 在 RHDH 中使用 Tekton 插件

您可以使用 Tekton 前端插件来查看 **PipelineRun** 资源。

#### 先决条件

- 已安装 Red Hat Developer Hub (RHDH)。
- 已安装 Tekton 插件。对于安装过程，请参阅 [安装和配置 Tekton 插件](#)。

#### 流程

1. 打开 RHDH 应用程序，然后从 **Catalog** 页面中选择一个组件。
2. 转至 **CI** 选项卡。

**CI** 选项卡显示与 Kubernetes 集群关联的 PipelineRun 资源列表。列表中包含管道运行详情，如 **NAME**、**VULNERABILITIES**、**STAT**、**STATSTATUS**、**STARTED** 和 **DURATION**。

| Pipeline Runs                                     |                    |             |                                  |                      |                      |         | Search |
|---|--------------------|-------------|----------------------------------|----------------------|----------------------|---------|--------|
| NAME  | VULNERABILITIES    | STATUS      | TASK STATUS                      | STARTED              | DURATION             | ACTIONS |        |
| > <b>PLR</b> 4ac96b4d-6cfa-4154-b2bc-27e27b4df977 | -                  | ✔ Succeeded | <div style="width: 100%;"></div> | 3/4/2024, 5:45:07 PM | 48 seconds           | 📄 🔗 📄   |        |
| > <b>PLR</b> 3f83247d-9e3b-4987-b69b-abf9d59c5302 | -                  | ✔ Succeeded | <div style="width: 100%;"></div> | 3/4/2024, 5:43:13 PM | 1 minute 14 seconds  | 📄 🔗 📄   |        |
| > <b>PLR</b> 01ab329a-94e8-4860-9cb9-94d471a598e1 | 🔥 3 📈 30 🟡 97 📉 81 | ✔ Succeeded | <div style="width: 100%;"></div> | 3/4/2024, 4:01:00 PM | 5 minutes 28 seconds | 📄 🔗 📄   |        |

5 rows 1-3 of 3

3. 点列表中的 PipelineRun 名称旁的展开行按钮来查看 PipelineRun visualization。管道运行资源包括要完成的任务。当您将鼠标指针悬停在任务卡上时，您可以查看完成该特定任务的步骤。

| Pipeline Runs                                     |                    |             |                                  |                      |                      |         | Search |
|---|--------------------|-------------|----------------------------------|----------------------|----------------------|---------|--------|
| NAME  | VULNERABILITIES    | STATUS      | TASK STATUS                      | STARTED              | DURATION             | ACTIONS |        |
| > <b>PLR</b> 4ac96b4d-6cfa-4154-b2bc-27e27b4df977 | -                  | ✔ Succeeded | <div style="width: 100%;"></div> | 3/4/2024, 5:45:07 PM | 48 seconds           | 📄 🔗 📄   |        |
| > <b>PLR</b> 3f83247d-9e3b-4987-b69b-abf9d59c5302 | -                  | ✔ Succeeded | <div style="width: 100%;"></div> | 3/4/2024, 5:43:13 PM | 1 minute 14 seconds  | 📄 🔗 📄   |        |
| ∨ <b>PLR</b> 01ab329a-94e8-4860-9cb9-94d471a598e1 | 🔥 3 📈 30 🟡 97 📉 81 | ✔ Succeeded | <div style="width: 100%;"></div> | 3/4/2024, 4:01:00 PM | 5 minutes 28 seconds | 📄 🔗 📄   |        |

```

graph LR
    A[git-clone 1/1] --> B[package 2/2]
    B --> C[scan-source 1/1]
    C --> D[build-sign-image 8/8]
    D --> E[acs-deploy-check 4/4]
    D --> F[acs-image-check 2/2]
    D --> G[acs-image-scan 3/3]
    E --> H[scan-export-sbom 3/3]
    F --> H
    G --> I[query-ekor-signed-provenance 2/2]
    H --> J[resync 1/1]
    I --> J
    
```

5 rows 1-3 of 3