# Dedicated Build Namespaces

l

* [Tekton Build Clusters](#DedicatedBuildNamespaces-TektonBuildClu)
* [Setup your own Build Namespace](#DedicatedBuildNamespaces-SetupyourownBu)
* [Permit Tekton to Access your Resources](#DedicatedBuildNamespaces-PermitTektonto)
  + [Bitbucket Access Credentials](#DedicatedBuildNamespaces-BitbucketAcces)
  + [Artifactory Access Credentials](#DedicatedBuildNamespaces-ArtifactoryAcc)
* ['Batch' Build Nodes - Autoscaling, Nodeaffinity](#DedicatedBuildNamespaces-'Batch'BuildNo)

# Tekton Build Clusters

CLEW maintains an OpenShift 4 buildcluster dedicated to the new build system with optimized resources for running Tekton pipelines.  
In order to separate concerns, access control, ressource allocation and transparent billing, each business unit has its own Esta-Tekton build namespace.

The architecture and topology of these dedicated build clusters is to be defined.

# Setup your own Build Namespace

Follow the steps described in the [Getting Started Guide](file:///C:\display\CLEW\Getting+Started#GettingStarted-SetupyourBuildNamespace).

# Permit Tekton to Access your Resources

The ESTA Tekton build system requires access to Bitbucket (where the sources are stored) and Artifactory (where the build artifacts are saved to). In order to follow the principle of least privilege, individual accounts and credentials (i.e. access tokens) should be created for each Tekton namespace. These credentials are used by the Tekton pipeline to read and write your project repositories and no others. The access credentials can be entered and changed via the Tekton Control panel in the System Configuration section.

We suggest to order specific system user accounts for both Bitbucket and Artifactory (suggestion: esta-build-<yourproject>). Create a [CLEW support ticket](https://flow.sbb.ch/servicedesk/customer/portal/8/create/34) (please provide an email address for this user) to request them unless you already have technical user accounts for your project.

## Bitbucket Access Credentials

Once the user account is created you need to do the following:

1. Grant "admin" permissions to the Bitbucket projects/repositories you want to build with Tekton.  
   See [this article](https://confluence.sbb.ch/display/CLEW/Git+-+Default+settings+of+a+Bitbucket+project#GitDefaultsettingsofaBitbucketproject-AuthorizeRepositoryManagersasAdministrators) how to grant admin rights to the Tekton system user (admin rights are required to allow Tekton to register webhooks to Git repositories).
2. Allow the Tekton user to push to the master/main branch of your repositories.  
   Open Project settings > Branch permissions and [edit the permissions](file:///C:\download\attachments\1916375385\bitbucket-edit-branch-permissions.png%3fversion=1&modificationDate=1662127510813&api=v2) on the master branch if configured.
3. Create an Access Token to be used by Tekton (instead of the real password).  
   See [this guide](https://confluence.sbb.ch/display/CLEW/Create+a+Bitbucket+Access+Token) how to do this. Make sure to select "Repository admin" in the Repository permissions drop-down.

## Artifactory Access Credentials

Access to Docker, NPM, Maven and Helm repositories on bin.sbb.ch needs to be explicitly granted to the Tekton build system. Order a new Artifactory user account and then follow these steps:

1. Grant "write" privileges (i.e. "Deploy/Cache" and "Delete/Overwrite") to the repositories where your project artifacts shall be stored.  
   Follow [this guide](https://confluence.sbb.ch/display/CLEW/Manage+and+unlock+Artifactory+7.x+Repo+permissions+from+users) to authorize the Tekton user. Start at <https://bin.sbb.ch/ui/admin/management/permissions>.
2. Create an Identity Token for this user for the use in Tekton. We do not recommend to use the password directly.  
   [Read here](file:///C:\display\CLEW\Create+an+Artifactory+Access+Token) how to create such a token.

# 'Batch' Build Nodes - Autoscaling, Nodeaffinity

Currently we support Build-Namespaces only on clusters aws02t and aws02t. These clusters are configured with specific nodes with the label "batch". All esta-tekton pipelines will run on this nodes. This nodes scale automatically up and also down to zero if not used. If there is no build running for a longer period of time, the Batch nodes are scaled to zero. If there is no batch node available it can take up to several minutes to start the batch node for your build. Once it is started, followup builds start instantly.

For the Tekton Pipelines to run on the 'Batch' nodes we apply automatically to every Pipeline following tolerations and affinity if not set manually in the pipeline.

podTemplate:

tolerations:

- key: node-role.kubernetes.io/batch

operator: Exists

effect: NoSchedule

affinity:

nodeAffinity:

requiredDuringSchedulingIgnoredDuringExecution:

nodeSelectorTerms:

- matchExpressions:

- key: node-role.kubernetes.io/batch

operator: Exists