# Jenkins Master for Private Lab

If a blueprint is testing in a private lab, one option is to use their own Jenkins master.  There are many ways to setup Jenkins and create the projects in Jenkins.  This is only one example.  The primary requirements are that the Jenkins jobs be run multiple times over time (not just a single successful run) and the log results are are pushed to the Akraino Nexus server.

## Setting up the Jenkins Master

You can find documentation on how to install Jenkins on the Jenkins project web site:  https://www.jenkins.io/doc/book/installing/

- There are several install methods available, but installing using Docker on Linux is the recommended method.
- The instructions map the `/var/jenkins_home` directory in the container to the Docker volume with the name `jenkins-data`. Instead of mapping the `/var/jenkins_home` directory to a Docker volume, it is recommended that you map this directory to one on your machine's local file system. For example, specifying the option `--volume /jenkins:/var/jenkins_home` would map the container's `/var/jenkins_home` directory to the `/jenkins` directory on your local machine.  This will allow easy access to the files and scripts used by the Jenkins container.

## Setting up the Jenkins Freestyle Project

Once Jenkins is installed and running, then projects need to be created to deploy and test the blueprint.  There are multiple ways to create a project, but here is one example:  https://java2blog.com/freestyle-projects-jenkins/

- The project should use the Source Code Management to pull the appropriate code from the Akraino gerrit
- The project should be triggered on a regular schedule or completion of another project if you break the testing into multiple projects.
- The build step is often an shell script located in the directory mounted as the jenkins home directory in the container (as described above).
- There should be a post build action to upload the logs to the Akraino Nexus (see How to: Push Logs to Nexus ).