

# Alpaquita Linux

## Getting started with images



Alpaquita Linux  
Revision 1.0  
January 2024

**be//soft**

Copyright © BellSoft Corporation 2018-2024.

BellSoft software contains open source software. Additional information about third party code is available at [https://bell-sw.com/third\\_party\\_licenses](https://bell-sw.com/third_party_licenses). You can also get more information on how to get a copy of source code by contacting [info@bell-sw.com](mailto:info@bell-sw.com).

THIS INFORMATION MAY CHANGE WITHOUT NOTICE. TO THE EXTENT PERMITTED BY APPLICABLE LAW, BELLSOFT PROVIDES THIS DOCUMENTATION “AS IS” WITHOUT WARRANTY OF ANY KIND, INCLUDING WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT. IN NO EVENT WILL BELLSOFT BE LIABLE TO YOU OR ANY THIRD PARTY FOR ANY LOSS OR DAMAGE, DIRECT OR INDIRECT, FROM THE USE OF THIS DOCUMENTATION, INCLUDING WITHOUT LIMITATION, LOST PROFITS, LOST INVESTMENT, BUSINESS INTERRUPTION, GOODWILL, OR LOST DATA, EVEN IF BELLSOFT IS EXPRESSLY ADVISED IN ADVANCE OF THE POSSIBILITY OF SUCH LOSS OR DAMAGE.

The use of any software product referenced in this document is governed by the applicable license agreement, which is not modified in any way by the terms of this notice.

Alpaquita, Liberica and BellSoft are trademarks or registered trademarks of BellSoft Corporation. The registered trademark Linux® is used pursuant to a sublicense from the Linux Foundation, the exclusive licensee of Linus Torvalds, owner of the mark on a worldwide basis. Java and OpenJDK are trademarks or registered trademarks of Oracle and/or its affiliates. Other trademarks are the property of their respective owners and are used only for identification purposes.

# Contents

## 1. Overview 4

---

Docker 4

Podman 4

Liberica JDK 4

## 2. Images 5

---

Available images 5

    alpaquita-linux-base 6

    alpaquita-linux-gcc 6

    alpaquita-linux-python 7

    liberica-runtime-container 7

    liberica-native-image-kit-container 8

Configuring Podman to run containers 9

# 1. Overview

## Docker

Docker is a containerization platform for building, sharing, and running your applications. Docker applications run in containers that can be used on any system: a laptop, desktop systems, servers, or in the cloud. A container is a standard unit of software that packages up code and all its dependencies, so the application runs quickly and reliably from one computing environment to another. Docker provides a set of tools for building and sharing container images, and running containers at any scale.

For more information, visit [Docker website](#).

## Podman

Podman is a daemonless, open-source Linux native tool designed to make it easy to find, run, build, share, and deploy applications using Open Containers Initiative (OCI) Containers and Container Images. Containers under the control of Podman can either be run by root or by a non-privileged user.

For more information, visit [Podman website](#).

## Liberica JDK

Liberica JDK is a TCK-verified 100 % open source progressive Java runtime from a major OpenJDK contributor. Liberica JDK supports an extensive range of system configurations and runs seamlessly in multiple virtual and cloud environments. Quarterly CPU and PSU releases in line with the OpenJDK release schedule help to preserve the runtime stability and security. For more information, visit [Liberica JDK page](#).

## 2. Images

[BellSoft](#) provides a number of Docker images with Alpaquita Linux. They are hosted on the Docker Hub Container Image Library available at <https://hub.docker.com/u/bellsoft>. Typically, downloading publicly available images does not require authentication.



### Note:

This chapter provides examples with the Docker tool. Podman provides a command-line interface similar to the Docker Container Engine. For more information, see [Configuring Podman to run containers on Alpaquita Linux](#).

### Available images

Images are grouped in several repositories, and each repository contains a number of tags. Each tag points to a Docker image.

To create a container from an image with a tag `<image_tag>` stored in a repository `<repository>`, do one of the following.

- Download the image using `docker pull`:

```
docker image pull docker bellsoft/<repository>:<image_tag>
```

and start a container from that image with `docker run`:

```
docker run -it --rm bellsoft/<repository>:<image_tag>
```

- Instead of two consecutive commands, you can combine two operations in a single `docker run` execution:

```
docker container run --rm -it bellsoft/<repository>:<image_tag>
```

For example:

```
docker container run --rm -it bellsoft/alpaquita-linux-base:23-musl
```

If the `:<image_tag>` part is omitted, it is implicitly substituted with `:latest`, that is the command is executed with the `latest` tag.

The description of available repositories and images is provided below.

## alpaquita-linux-base

The repository contains base Alpaquita Linux images.

The following are available tags for images based on musl or glibc Alpaquita Linux version.

Images with the latest release in the LTS release series are:

- 23-musl
- 23-glibc

Images with the latest release in the stream release series are:

- stream-musl
- stream-glibc

For example:

```
docker run --rm bellsoft/alpaquita-linux-base:23-musl cat /etc/os-release
```

```
NAME="BellSoft Alpaquita Linux LTS"  
ID=alpaquita  
ID_LIKE=alpine  
VERSION_ID=23.0  
PRETTY_NAME="BellSoft Alpaquita Linux LTS 23.0 (musl)"  
HOME_URL="https://bell-sw.com/"  
BUG_REPORT_URL="https://bell-sw.com/support/"  
ALPAQUITA_LIBC=musl
```

## alpaquita-linux-gcc

The repository contains images with the GCC compiler, tools, and libraries for development in C/C++.

The following are available tags for images based on musl or glibc Alpaquita Linux version.

Image with the latest GCC are:

- 12.2-musl
- 12.2-glibc

For example:

```
docker run --rm bellsoft/alpaquita-linux-gcc:12.2-glibc gcc --version
gcc (Alpaquita 12.2.1_git20220924-r5) 12.2.1 20220924
Copyright (C) 2022 Free Software Foundation, Inc.
This is free software; see the source for copying conditions.  There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
```

## alpaquita-linux-python

The repository contains images with Python and accompanying tools (such as pip3, wheel, setuptools).

The following are available tags for images based on musl or glibc Alpaquita Linux version.

Images with the latest Python 3.11.x version are:

- 3.11-musl
- 3.11-glibc

For example:

```
docker run --rm bellsoft/alpaquita-linux-python:3.11-glibc python3 --version
Python 3.11.4
```

## liberica-runtime-container

The repository contains images with preinstalled Liberica JDK.

Java 8 JRE and Java 8 JDK images contain Liberica JDK. JRE and JDK images with Java 11 and Java 17 contain Liberica JDK Lite. Regular and slim images are available for each such combination.

- bellsoft/liberica-runtime-container:<X>-<Y>-<Z>
- bellsoft/liberica-runtime-container:<X>-<Y>-slim-<Z>

Also, the stream version includes images with prebuilt [Shared Archive](#), such as:

- bellsoftcr/liberica-runtime-container:<X>-<Y>-cds-<Z>
- bellsoftcr/liberica-runtime-container:<X>-<Y>-cds-slim-<Z>

Where <X> is the jre or jdk bundle type; <Y> is the Java version, such as 8, 11, 17; <Z> is either glibc or musl C Runtime Library

The Java version can be specified in three variants:

- Major version, such as 8, 11, or 17
- Release version, such as 11.0.20
- Full version (with build number), such as 11.0.20\_9 that corresponds to 11.0.20+9

Slim images are smaller and do not include an OS package manager. For example, for a regular Java 17 JRE image based on glibc, the full path is as follows:

```
bellsoft/liberica-runtime-container:jre-17-glibc
```

For a slim image with Java 11 based on musl, the full path is:

```
bellsoft/liberica-runtime-container:jdk-11-slim-musl
```

**Note:**

Slim images do not include the apk tools, and the layers in the image are additionally compressed. These modifications reduce the size of the final image in the registry and make it impossible to install additional software in the image using standard OS tools to improve security.

For example:

```
docker run --rm bellsoft/liberica-runtime-container:jre-11-musl java -version
openjdk version "11.0.19" 2023-04-18 LTS
OpenJDK Runtime Environment (build 11.0.19+7-LTS)
OpenJDK 64-Bit Server VM (build 11.0.19+7-LTS, mixed mode)
```

## liberica-native-image-kit-container

The repository contains images that come with Liberica Native Image Kit (NIK) 22.3 software. The images contain JDK 11 or JDK 17 and native-image components.

The following are available tags based on the corresponding Alpaquita Linux version with musl or glibc libraries.

Images with NIK 22.3.2:

- jdk-11-nik-22.3-glibc
- jdk-11-nik-22.3-musl



- `jdk-17-nik-22.3-glibc`
- `jdk-17-nik-22.3-musl`

For example:

```
docker run --rm bellsoft/liberica-native-image-kit-container:jdk-17-nik-22-musl
native-image --version
GraalVM 22.3.2 Java 17 CE (Java Version 17.0.7+7-LTS)
```

## Configuring Podman to run containers

Alpaquita Linux can act as a host system for running containers. Docker and Podman tools are included in the package, which can be installed from the package repository. The setup is different for different tools. The following section describes how to set up the Podman tool. The setup includes the following steps:

1. Run the following command:

```
sudo apk add fuse-overlay podman
```

2. Set `/etc/subuid` and `/etc/subgid` values to specify the range of UIDs for users and allow Podman to run containers in the appropriate user namespace.

3. Since Alpaquita Linux does not use `systemd`, you may need to run the following command:

```
sudo mount --make-rshared /
```

4. Run the following command to set up `cgroups`:

```
sudo rc-update add cgroups; sudo rc-service cgroups start
```

5. To enable `cgroups V2`, edit `/etc/rc.conf` and enable the "unified" option for `rc_cgroup_mode`.

6. Load the modules that are not loaded by default:

```
sudo modprobe tun; sudo modprobe fuse
```

Now you can use Podman to run containers with BellSoft images.



Alpaquita Linux  
Getting started with  
images

**be//soft**