
Zero-dependency Erlang RPM for RabbitMQ

This is a (virtually) zero dependency Erlang RPM package that provides **just enough to run RabbitMQ**. It may or may not be suitable for running other Erlang-based software or 3rd party RabbitMQ plugins.

x86-64 binaries are provided with every release, while aarch64 (ARM64) binaries will be limited to new major and minor release, and patches when resources allow.

Binary packages can be produced on any host that can run Docker, including aarch64 hosts. See the **Building from Source** section below.

Supported RPM-based Distributions

Binary builds of this package target **modern RPM-based distributions**:

- RHEL 8.4 or later
- CentOS Stream 8
- CentOS Stream 9
- Rocky Linux 8.5 or later
- Fedora 34 or later
- Amazon Linux 2023
- VMware PhotonOS 5
- Oracle Linux 9
- Alma Linux 9

CentOS 7 and derivatives

Team RabbitMQ stopped supporting CentOS 7 in May 2022.

Erlang 26.2.2 and 25.3.2.9 include one-off CentOS 7 packages statically linked against OpenSSL 1.1.x.

Regular CentOS 7 and Amazon Linux 2 builds were produced up to Erlang 23.3.4.18. They are dynamically linked against OpenSSL 1.0.

Packages for Debian-based Distributions

Team RabbitMQ also packages recent Erlang/OTP releases for Debian and a modern Erlang PPA for Ubuntu.

Provided Erlang/OTP Versions

The package targets Erlang/OTP 26.x and 25.x. Both x86-64 and aarch64 versions can be build in containers.

First RabbitMQ release series with Erlang 26 support is 3.12.

RabbitMQ Version Compatibility

See Supported Erlang Versions in RabbitMQ documentation for an up-to-date compatibility matrix.

Erlang 26 First RabbitMQ release series with Erlang 26 support is 3.12.

Erlang 25 Erlang 25 is supported by RabbitMQ starting with 3.10.0.

Erlang 25 depends on OpenSSL 1.1, which is **not available on CentOS 7**. Therefore Erlang 25 packages are **only produced for modern Fedora, Rocky Linux, CentOS Stream, and Amazon Linux 2023**.

Implicit OpenSSL/libcrypto Dependency

This package intentionally **does not include OpenSSL/libcrypto**. It must be provisioned separately. Recent Erlang versions require a modern OpenSSL version, currently this means 1.1.x or later.

Release Artifacts

For direct RPM package downloads, see GitHub releases.

There is also a dnf repository that distributes x64_86 builds of this package. See the repository setup instructions below.

Signing Keys

The package is signed using the standard RabbitMQ signing key:

```
1 ## primary RabbitMQ signing key
2 rpm --import https://github.com/rabbitmq/signing-keys/releases/download/2.0/rabbitmq-release-signing-key.asc
```

To use the Cloudsmith dnf repository, a separate Cloudsmith repository key must be imported:

```
1 ## primary RabbitMQ signing key
2 rpm --import https://github.com/rabbitmq/signing-keys/releases/download
  /2.0/rabbitmq-release-signing-key.asc
3 ## modern Erlang repository
4 rpm --import 'https://dl.cloudsmith.io/public/rabbitmq/rabbitmq-erlang/
  gpg.E495BB49CC4BBE5B.key'
```

Direct Downloads from GitHub

This package is distributed as a single RPM, which makes it convenient to download and install using `dnf install -y /path/to/erlang.rpm`.

Erlang 25 x86-64 and aarch64 releases can be provisioned on RHEL 9, CentOS Stream 9, Amazon Linux 2023, and modern Fedora using a direct download:

```
1 # This is just an example that uses an aarch64 package for Amazon Linux
  2023
2 cd /tmp/
3 curl -sfl -O https://github.com/rabbitmq/erlang-rpm/releases/download/
  v25.3.2.5/erlang-25.3.2.5-1.amzn2023.aarch64.rpm
4 sudo dnf install -y ./erlang-25.3.2.5-1.amzn2023.aarch64.rpm
```

Latest Erlang Version from a Cloudsmith Mirror

This package is distributed via a dnf repository that mirrors from Cloudsmith.

The upstream Cloudsmith repository is **subject to traffic quotas**. When the quota is reached, package installations will only be possible from the mirrors, so it is highly recommended that you use them.

The examples below use a mirror of the Cloudsmith repo. All packages in it are **signed with the same signing key**.

Erlang 25 on RHEL 9, CentOS Stream 9, modern Fedora, Rocky Linux 9 (x86-64)

Erlang 25 x86-64 releases can be provisioned on RHEL 9, CentOS Stream 9, Rocky Linux, and modern Fedora using a dnf (yum) repository (a Cloudsmith mirror):

```
1 # In /etc/yum.repos.d/modern_erlang.repo
2
3 ##
4 ## Zero dependency Erlang RPM
5 ##
6
```

```
7 [modern-erlang]
8 name=modern-erlang-el8
9 # uses a Cloudsmith mirror @ yum.novemberain.com in addition to its
  Cloudsmith upstream.
10 # Unlike Cloudsmith, the mirror is very unlikely to hit its traffic
   quota
11 baseurl=https://yum1.novemberain.com/erlang/el/8/$basearch
12     https://yum2.novemberain.com/erlang/el/8/$basearch
13 repo_gpgcheck=1
14 enabled=1
15 gpgkey=https://dl.cloudsmith.io/public/rabbitmq/rabbitmq-erlang/gpg.
    E495BB49CC4BBE5B.key
16 gpgcheck=1
17 sslverify=1
18 sslcacert=/etc/pki/tls/certs/ca-bundle.crt
19 metadata_expire=300
20 pkg_gpgcheck=1
21 autorefresh=1
22 type=rpm-md
23
24 [modern-erlang-noarch]
25 name=modern-erlang-el8-noarch
26 # uses a Cloudsmith mirror @ yum.novemberain.com.
27 # Unlike Cloudsmith, the mirror is very unlikely to hit its traffic
   quotas
28 baseurl=https://yum1.novemberain.com/erlang/el/8/noarch
29     https://yum2.novemberain.com/erlang/el/8/noarch
30 repo_gpgcheck=1
31 enabled=1
32 gpgkey=https://dl.cloudsmith.io/public/rabbitmq/rabbitmq-erlang/gpg.
    E495BB49CC4BBE5B.key
33     https://github.com/rabbitmq/signing-keys/releases/download/2.0/
        rabbitmq-release-signing-key.asc
34 gpgcheck=1
35 sslverify=1
36 sslcacert=/etc/pki/tls/certs/ca-bundle.crt
37 metadata_expire=300
38 pkg_gpgcheck=1
39 autorefresh=1
40 type=rpm-md
41
42 [modern-erlang-source]
43 name=modern-erlang-el8-source
44 # uses a Cloudsmith mirror @ yum.novemberain.com.
45 # Unlike Cloudsmith, the mirror is very unlikely to hit its traffic
   quotas
46 baseurl=https://yum1.novemberain.com/erlang/el/8/SRPMS
47     https://yum2.novemberain.com/erlang/el/8/SRPMS
48 repo_gpgcheck=1
49 enabled=1
50 gpgkey=https://dl.cloudsmith.io/public/rabbitmq/rabbitmq-erlang/gpg.
```

```

E495BB49CC4BBE5B.key
51      https://github.com/rabbitmq/signing-keys/releases/download/2.0/
      rabbitmq-release-signing-key.asc
52  gpgcheck=1
53  sslverify=1
54  sslcacert=/etc/pki/tls/certs/ca-bundle.crt
55  metadata_expire=300
56  pkg_gpgcheck=1
57  autorefresh=1
```

To install the package:

```

1  dnf update -y
2  dnf install -y erlang
```

Erlang 25 on RHEL 8, CentOS Stream 8, Rocky Linux 8 (x86-64)

Erlang 25 x86-64 releases can be provisioned on RHEL 8, CentOS Stream 8, Rocky Linux, and modern Fedora using a dnf (yum) repository (a Cloudsmith mirror):

```

1  # In /etc/yum.repos.d/modern_erlang.repo
2  [modern-erlang]
3  name=modern-erlang-el8
4  # uses a Cloudsmith mirror @ yum1.novemberain.com.
5  # Unlike Cloudsmith, it does not have traffic quotas
6  baseurl=https://yum1.novemberain.com/erlang/el/8/$basearch
7      https://yum1.novemberain.com/erlang/el/8/$basearch
8  repo_gpgcheck=1
9  enabled=1
10 gpgkey=https://dl.cloudsmith.io/public/rabbitmq/rabbitmq-erlang/gpg.
    E495BB49CC4BBE5B.key
11 gpgcheck=1
12 sslverify=1
13 sslcacert=/etc/pki/tls/certs/ca-bundle.crt
14 metadata_expire=300
15 pkg_gpgcheck=1
16 autorefresh=1
17 type=rpm-md
18
19 [modern-erlang-noarch]
20 name=modern-erlang-el8-noarch
21 # uses a Cloudsmith mirror @ yum1.novemberain.com.
22 # Unlike Cloudsmith, the mirror is very unlikely to hit its traffic
    quotas
23 baseurl=https://yum1.novemberain.com/erlang/el/8/noarch
24      https://yum2.novemberain.com/erlang/el/8/noarch
25 repo_gpgcheck=1
26 enabled=1
```

```
27  gpgkey=https://dl.cloudsmith.io/public/rabbitmq/rabbitmq-erlang/gpg.  
    E495BB49CC4BBE5B.key  
28      https://github.com/rabbitmq/signing-keys/releases/download/2.0/  
        rabbitmq-release-signing-key.asc  
29  gpgcheck=1  
30  sslverify=1  
31  sslcacert=/etc/pki/tls/certs/ca-bundle.crt  
32  metadata_expire=300  
33  pkg_gpgcheck=1  
34  autorefresh=1  
35  type=rpm-md  
36  
37  [modern-erlang-source]  
38  name=modern-erlang-el8-source  
39  # uses a Cloudsmith mirror @ yum1.novemberain.com.  
40  # Unlike Cloudsmith, the mirror is very unlikely to hit its traffic  
    quotas  
41  baseurl=https://yum1.novemberain.com/erlang/el/8/SRPMS  
42      https://yum2.novemberain.com/erlang/el/8/SRPMS  
43  repo_gpgcheck=1  
44  enabled=1  
45  gpgkey=https://dl.cloudsmith.io/public/rabbitmq/rabbitmq-erlang/gpg.  
    E495BB49CC4BBE5B.key  
46      https://github.com/rabbitmq/signing-keys/releases/download/2.0/  
        rabbitmq-release-signing-key.asc  
47  gpgcheck=1  
48  sslverify=1  
49  sslcacert=/etc/pki/tls/certs/ca-bundle.crt  
50  metadata_expire=300  
51  pkg_gpgcheck=1  
52  autorefresh=1
```

To install the package:

```
1  dnf update -y  
2  dnf install -y erlang
```

Available Erlang Applications

This package removes certain Erlang/OTP components that are non-essential for running RabbitMQ but can greatly complicate installation process by bringing in complicated dependencies.

Only the following OTP applications are provided:

- asn1
- compiler
- crypto
- eldap

-
- erl_interface
 - erts
 - inets
 - kernel
 - mnesia
 - os_mon
 - public_key
 - reltool
 - runtime_tools
 - sasl
 - snmp
 - ssl
 - stdlib
 - syntax_tools
 - xmerl

Building from Source

With Docker

This repository provides scripts that can build an image with the `rpmbuild` toolchain and build the package in it. They can be found under the `./docker` directory.

The recommended host environment for these builds is Fedora 38+. Due to Docker-specific volume sharing permissions intricacies, performing such builds on macOS will require modifications to the build scripts.

```
1 cd docker
2
3 #
4 # Use build-image-and-rpm.sh to execute all scripts:
5 # build an image and build the RPM in it
6 #
7 # Supported distribution aliases:
8 # * stream9 for CentOS Stream 9
9 # * stream8 for CentOS Stream 8
10 # * al2023 for Amazon Linux 2023
11 # * f38 for Fedora 38
12 #
13 ./build-image-and-rpm.sh stream9 --no-cache
14
15 #
16 # To only build an image with the necessary toolchain,
```

```
17 # use ./build-docker-image.sh.
18 #
19 # Supported distribution aliases:
20 # * stream9 for CentOS Stream 9
21 # * stream8 for CentOS Stream 8
22 # * al2023 for Amazon Linux 2023
23 # * f38 for Fedora 38
24 ./build-docker-image.sh stream9 --no-cache
25
26 #
27 # To only build the RPM using an already built and available image,
28 # use ./build-rpm-in-docker.sh
29 #
30 # Supported distribution aliases:
31 # * stream9 for CentOS Stream 9
32 # * stream8 for CentOS Stream 8
33 # * al2023 for Amazon Linux 2023
34 # * f38 for Fedora 38
35 #
36 ./build-rpm-in-docker.sh stream9
```

Built packages can be found under `docker/pkg-build-dir/RPMS/{architecture}/`.

Without Docker

On an RPM-based distro (CentOS Stream 9, modern Fedora or Amazon Linux 2023), the package can be built without containers.

```
1 # Note: this will use sudo to drive `dnf update -y` and install a few
   packages
2 make
```

and see `RPMS/{architecture}/`. Note that all artifacts created this way may be owned by root due to the use of `sudo`.

Scope of Patching

The patches apply to the build system only (Makefile, Makefile.in and similar files). No source code is patched.

Older Versions

The directory `versions` contains the patch files used for the older versions. Git repository history and release archive can be useful as well.

Copyright and License

Copyright VMware, Inc and its affiliates, 2011-2023. All Rights Reserved.

Released under the Apache Software License 2.0, same as Erlang/OTP starting with 18.0.