

LINUX

Cómo instalar y configurar RabbitMQ en Ubuntu 16.04

• 7 marzo, 2021 💬 0 🔥 151 📖 3 minutos de lectura

RabbitMQ es un software de agente de mensajes de código abierto que implementa el Protocolo de cola de mensajes avanzado (AMQP), el estándar emergente para la mensajería empresarial de alto rendimiento. Es una de las soluciones de intermediarios de mensajes más populares del mercado, que se ofrece con una licencia de código abierto (Mozilla Public License v1.1) como una implementación de AMQP desarrollada utilizando el lenguaje Erlang, que en realidad es relativamente fácil de usar y comenzar. . El servidor RabbitMQ es una implementación robusta y escalable de un corredor AMQP. AMQP es un estándar de código abierto ampliamente aceptado para distribuir y transferir mensajes desde un origen a un destino. Como protocolo y estándar, establece un terreno común para que varias aplicaciones y productos intermedios de intermediarios de mensajería interactúen sin encontrar problemas causados por decisiones de diseño establecidas individualmente.

Conceptos del servidor RabbitMQ:

A continuación se muestran algunos conceptos importantes que debemos definir antes de iniciar la configuración de instalación de RabbitMQ. El host virtual predeterminado, el usuario predeterminado y los permisos predeterminados se utilizan en los ejemplos que siguen, pero es bueno tener una idea de lo que es.

Productor: Aplicación que envía los mensajes.

Consumidor: Aplicación que recibe los mensajes.

Cola: Búfer que almacena mensajes.

Mensaje: Información que se envía del productor al consumidor a través de RabbitMQ.

Conexión: Una conexión es una conexión TCP entre su aplicación y el corredor de RabbitMQ.

Canal: Un canal es una conexión virtual dentro de una conexión. Cuando publica o consume mensajes o se suscribe a una cola, todo se hace a

través de un canal.

Intercambio: Recibe mensajes de los productores y los envía a las colas según las reglas definidas por el tipo de intercambio. Para recibir mensajes, una cola debe estar vinculada a al menos un intercambio.

Vinculante: Un enlace es un enlace entre una cola y un intercambio.

Clave de enrutamiento: La clave de enrutamiento es una clave que el intercambio analiza para decidir cómo enrutar el mensaje a las colas. La clave de enrutamiento es como una dirección para el mensaje.

anfitrión virtual: Un host virtual proporciona una forma de segregar aplicaciones utilizando la misma instancia de RabbitMQ. Diferentes usuarios pueden tener diferentes privilegios de acceso a diferentes vhost y se pueden crear colas e intercambios para que solo existan en un vhost.

Requisitos previos:

Nuestro primer paso es asegurarnos de que todos los paquetes del sistema estén actualizados ejecutando los siguientes comandos apt-get en la terminal de línea de comandos.

```
# apt-get update
```

```
# apt-get upgrade
```

```
root@ubuntu-16i:~# apt-get update
Hit:1 http://security.ubuntu.com/ubuntu xenial-security InRelease
Hit:2 http://gb.archive.ubuntu.com/ubuntu xenial InRelease
Hit:3 http://gb.archive.ubuntu.com/ubuntu xenial-updates InRelease
Hit:4 http://gb.archive.ubuntu.com/ubuntu xenial-backports InRelease
Reading package lists... Done
root@ubuntu-16i:~# apt-get upgrade
Reading package lists... Done
Building dependency tree
Reading state information... Done
Calculating upgrade... Done
The following packages were automatically installed and are no longer required:
  linux-image-3.19.0-15-generic linux-image-extra-3.19.0-15-generic
Use 'sudo apt autoremove' to remove them.
The following packages will be upgraded:
  accounts-service apport bash-completion command-not-found command-not-found-data dh-python dmidecode language-selector-common libaccounts-service0 libexpati
libldap-2.4-2 libpam-systemd libsystemd0 libudev1 linux-headers-4.4.0-22 linux-headers-4.4.0-22-generic linux-image-4.4.0-22-generic
linux-image-extra-4.4.0-22-generic python3-apport python3-commandnotfound python3-problem-report python3-urllib3 systemd systemd-sysv udev
25 to upgrade, 0 to newly install, 0 to remove and 0 not to upgrade.
Need to get 75.3 MB of archives.
After this operation, 25.9 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://gb.archive.ubuntu.com/ubuntu xenial-updates/main amd64 systemd-sysv amd64 229-4ubuntu6 [15.2 kB]
Get:2 http://gb.archive.ubuntu.com/ubuntu xenial-updates/main amd64 libpam-systemd amd64 229-4ubuntu6 [115 kB]
Get:3 http://gb.archive.ubuntu.com/ubuntu xenial-updates/main amd64 libsystemd0 amd64 229-4ubuntu6 [208 kB]
Get:4 http://gb.archive.ubuntu.com/ubuntu xenial-updates/main amd64 systemd amd64 229-4ubuntu6 [3,642 kB]
Get:5 http://gb.archive.ubuntu.com/ubuntu xenial-updates/main amd64 udev amd64 229-4ubuntu6 [390 kB]
Get:6 http://gb.archive.ubuntu.com/ubuntu xenial-updates/main amd64 libudev1 amd64 229-4ubuntu6 [57.9 kB]
Get:7 http://gb.archive.ubuntu.com/ubuntu xenial-updates/main amd64 bash-completion all 1:2.1-4.2ubuntu1.1 [152 kB]
```

Después de la actualización del sistema, necesitamos obtener las dependencias principales de RabbitMQ, como Erlang. Usemos el siguiente comando para obtener Erlang en nuestro servidor Ubuntu 16.04.

```
root@ubuntu-16:~#
root@ubuntu-16:~# apt-get install rabbitmq-server
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  erlang-nox
Suggested packages:
  erlang-manpages erlang-doc
The following NEW packages will be installed:
  erlang-nox rabbitmq-server
0 to upgrade, 2 to newly install, 0 to remove and 0 not to upgrade.
Need to get 4,265 kB of archives.
After this operation, 5,292 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://gb.archive.ubuntu.com/ubuntu xenial/main amd64 erlang-nox all 1:18.3-dfsg-lubuntu3 [14.0 kB]
Get:2 http://gb.archive.ubuntu.com/ubuntu xenial/main amd64 rabbitmq-server all 3.5.7-1 [4,251 kB]
Fetched 4,265 kB in 0s (9,237 kB/s)
Selecting previously unselected package erlang-nox.
(Reading database ... 89040 files and directories currently installed.)
Preparing to unpack .../erlang-nox_1:18.3-dfsg-lubuntu3_all.deb ...
Unpacking erlang-nox (1:18.3-dfsg-lubuntu3) ...
Selecting previously unselected package rabbitmq-server.
Preparing to unpack .../rabbitmq-server_3.5.7-1_all.deb ...
Unpacking rabbitmq-server (3.5.7-1) ...
Processing triggers for man-db (2.7.5-1) ...
Processing triggers for systemd (229-4ubuntu6) ...
Processing triggers for ureadahead (0.100.0-19) ...
Setting up erlang-nox (1:18.3-dfsg-lubuntu3) ...
Setting up rabbitmq-server (3.5.7-1) ...
Adding group `rabbitmq' (GID 122) ...
Done.
Adding system user `rabbitmq' (UID 112) ...
Adding new user `rabbitmq' (UID 112) with group `rabbitmq' ...
Not creating home directory `/var/lib/rabbitmq'.
Processing triggers for systemd (229-4ubuntu6) ...
Processing triggers for ureadahead (0.100.0-19) ...
```

Inicio de los servicios de RabbixMQ:

El servidor RabbitMQ se ha instalado en Ubuntu 16.04, ahora ejecute los siguientes comandos para iniciar y verificar el estado del servidor RabbitMQ y permitir que sus servicios se inicien automáticamente después de cada reinicio.

```
# systemctl enable rabbitmq-server
```

```
# systemctl start rabbitmq-server
```

```
# systemctl status rabbitmq-server
```

Habilitación de la consola de administración de RabbitMQ

El servidor RabbitMQ está en funcionamiento, ahora le mostraremos cómo puede configurar su consola de administración web mediante el complemento de administración de rabbitmq. El complemento de administración de Rabbitmq le permite administrar y monitorear su servidor RabbitMQ de varias maneras, como enumerar y eliminar intercambios, colas, enlaces y muchos más.

Ejecutemos el siguiente comando para instalar este complemento en su servidor Ubuntu 16.04.

```
# rabbitmq-plugins enable rabbitmq_management
```

El complemento `rabbitmq_management` es una combinación de los siguientes complementos que se habilitarán después de ejecutar el comando anterior.

- `mochiweb`
- `máquina web`
- `rabbitmq_web_dispatch`
- `amqp_client`
- `rabbitmq_management_agent`
- `rabbitmq_management`

Ahora podemos acceder a la consola de administración de RabbitMQ desde nuestro navegador web, disponible en el puerto HTTP **15672** por defecto. También puede crear un nuevo usuario administrador usando los siguientes comandos.

```
# rabbitmqctl add_user radmin radmin
# rabbitmqctl set_user_tags radmin administrator
# rabbitmqctl set_permissions -p / radmin ".*" ".*" ".*"
```

Ahora abra la URL debajo junto con el puerto predeterminado e inicie sesión con su usuario y contraseña recién creados. También puede utilizar el nombre de usuario 'invitado' predeterminado y la contraseña de 'invitado' para iniciar sesión.

```
http://your_servers_ip:15672/
```

Usando la consola web RabbitMQ:

Bienvenido a la consola de administración web de RabbitMQ, después de proporcionar los detalles de inicio de sesión de usuario correctos, puede administrar su servidor RabbitMQ desde su web.

Conclusión:

La ejecución de rabbitmq-server en primer plano muestra un mensaje de banner e informa sobre el progreso en la secuencia de inicio, que concluye con el mensaje «corredor en ejecución», lo que indica que el corredor de RabbitMQ se ha iniciado correctamente. RabbitMQ es una pila de aplicaciones completamente desarrollada (es decir, un intermediario de mensajes) que le brinda todas las herramientas con las que necesita trabajar, en lugar de actuar como un marco para que usted implemente el suyo. Espero que este artículo le resulte muy útil e

interesante. No te olvides de compartir con tus amigos.