

Cómo agregar manualmente el cluster de Rabbitmq

Contenido

[Introducción](#)

[prerrequisitos](#)

[Requisitos](#)

[Componentes Utilizados](#)

[Problema](#)

[Verifique si RabbitMQ está fuera de cluster](#)

[Solución](#)

Introducción

Este documento describe cómo agregar manualmente RabbitMQ a un cluster si el cluster está quebrado.

Prerrequisitos

Requisitos

No hay requisitos específicos para este documento.

Componentes Utilizados

La información que contiene este documento se basa en las siguientes versiones de software y hardware.

- Servidor de RabbitMQ del mínimo 2
- Un balanceador de la carga

Problema

Este artículo le dirige en cómo verificar el cluster de RabbitMQ y agregar manualmente éstos caso al cluster. CloudCenter proporciona a un Asisitente para configurar la Alta disponibilidad (HA) para RabbitMQ sin embargo, en muy algunos caso que dice que el HA está configurado con éxito después de que salga al Asisitente pero el cluster de RabbitMQ no se forma correctamente.

Verifique si RabbitMQ está fuera de cluster

Paso 1. Inicie sesión a todo el servidor de RabbitMQ con el uso de la consola CLI.

Paso 2. Verifique si el servidor de RabbitMQ se ejecuta en todos los casos.

```
#ps -ef | grep rabbit
```

Output:

```
rabbitmq 1677      1  0 14:47 ?          00:00:00 /usr/lib/erlang/erts-6.4/bin/epmd -daemon
root     1973      1  0 14:47 ?          00:00:00 /bin/sh /etc/rc.d/init.d/rabbitmq-server start
root     2000    1973  0 14:47 ?          00:00:00 /bin/bash -c ulimit -S -c 0 >/dev/null 2>&1 ;
/usr/sbin/rabbitmq-server
root     2003    2000  0 14:47 ?          00:00:00 su rabbitmq -s /bin/sh -c
/usr/lib/rabbitmq/bin/rabbitmq-server
rabbitmq 2026    2003  0 14:47 ?          00:00:05 /usr/lib/erlang/erts-6.4/bin/beam -W w -K true -
A30 -P 1048576 -- -root /usr/lib/erlang -prognose erl -- -home /var/lib/rabbitmq -- -pa
/usr/lib/rabbitmq/lib/rabbitmq_server-3.5.1/sbin/./ebin -noshell -noinput -s rabbit boot -sname
rabbit@ip-172-31-32-101 -boot start_sasl -config /etc/rabbitmq/rabbitmq -kernel
inet_default_connect_options [{nodelay,true}] -sasl errlog_type error -sasl sasl_error_logger
false -rabbit error_logger {file,"/var/log/rabbitmq/rabbit@ip-172-31-32-101.log"} -rabbit
sasl_error_logger {file,"/var/log/rabbitmq/rabbit@ip-172-31-32-101-sasl.log"} -rabbit
enabled_plugins_file "/etc/rabbitmq/enabled_plugins" -rabbit plugins_dir
"/usr/lib/rabbitmq/lib/rabbitmq_server-3.5.1/sbin/./plugins" -rabbit plugins_expand_dir
"/var/lib/rabbitmq/mnesia/rabbit@ip-172-31-32-101-plugins-expand" -os_mon start_cpu_sup false -
os_mon start_disksup false -os_mon start_memsup false -mnesia dir
"/var/lib/rabbitmq/mnesia/rabbit@ip-172-31-32-101" -kernel inet_dist_listen_min 25672 -kernel
inet_dist_listen_max 25672
rabbitmq 2242    2026  0 14:47 ?          00:00:00 inet_gethost 4
rabbitmq 2243    2242  0 14:47 ?          00:00:00 inet_gethost 4
root     2602    2588  0 15:04 pts/0      00:00:00 grep --color=auto rabbit
OR
#/sbin/service rabbitmq-server status
```

```
Status of node 'rabbit@ip-172-31-32-101' ...
[{pid,2026},
 {running_applications,
  [{rabbitmq_management,"RabbitMQ Management Console","3.5.1"},
   {rabbitmq_web_dispatch,"RabbitMQ Web Dispatcher","3.5.1"},
   {webmachine,"webmachine","1.10.3-rmq3.5.1-gite9359c7"},
   {mochiweb,"MochiMedia Web Server","2.7.0-rmq3.5.1-git680dba8"},
   {rabbitmq_management_agent,"RabbitMQ Management Agent","3.5.1"},
   {rabbit,"RabbitMQ","3.5.1"}],
  ]}
```

----- Text omitted for brevity

Paso 3. Verifique el estatus del cluster de todo el caso con estos comandos:

```
[root@ip-172-31-32-101 ~]# rabbitmqctl cluster_status
Cluster status of node 'rabbit@ip-172-31-32-101' ...
[{nodes, [{disc, ['rabbit@ip-172-31-32-101']}]},
 {running_nodes, ['rabbit@ip-172-31-32-101']},
 {cluster_name, <<"rabbit@ip-172-31-32-101.us-east-2.compute.internal">>},
 {partitions, []}]
```

En esta salida, usted puede identificar que hay solamente un nodo que se ejecuta en el cluster.

Solución

En este escenario, usted agrega **rabbit@ip-172-31-32-101** a su cluster **rabbit@ip-172-31-45-110.us-east-2.compute.internal**

Paso 1. Para conectar el segundo servidor de RabbitMQ, usted necesita asegurarse de que la aplicación esté parada y se una al cluster.

Paso 2. El Switch al servidor RabbitMQ2 y para la aplicación.

```
[root@ip-172-31-32-101 ~]# rabbitmqctl cluster_status
Cluster status of node 'rabbit@ip-172-31-32-101' ...
[{nodes, [{disc, ['rabbit@ip-172-31-32-101']}]},
 {running_nodes, ['rabbit@ip-172-31-32-101']},
 {cluster_name, <<"rabbit@ip-172-31-32-101.us-east-2.compute.internal">>},
 {partitions, []}]
```

Paso 3. Únase al servidor RabbitMQ2 al cluster RabbitMQ1.

```
[root@ip-172-31-32-101 ~]# rabbitmqctl cluster_status
Cluster status of node 'rabbit@ip-172-31-32-101' ...
[{nodes, [{disc, ['rabbit@ip-172-31-32-101']}]},
 {running_nodes, ['rabbit@ip-172-31-32-101']},
 {cluster_name, <<"rabbit@ip-172-31-32-101.us-east-2.compute.internal">>},
 {partitions, []}]
```

Paso 4. Comience la aplicación rabbitmq2.

```
[root@ip-172-31-32-101 ~]# rabbitmqctl cluster_status
Cluster status of node 'rabbit@ip-172-31-32-101' ...
[{nodes, [{disc, ['rabbit@ip-172-31-32-101']}]},
 {running_nodes, ['rabbit@ip-172-31-32-101']},
 {cluster_name, <<"rabbit@ip-172-31-32-101.us-east-2.compute.internal">>},
 {partitions, []}]
```

Usted puede ver que los dos Nodos están unidos a en un cluster cuando usted funciona con el comando del **cluster_status** en cualquiera de los Nodos.

```
[root@ip-172-31-32-101 ~]# rabbitmqctl cluster_status
Cluster status of node 'rabbit@ip-172-31-32-101' ...
[{nodes, [{disc, ['rabbit@ip-172-31-32-101']}]},
 {running_nodes, ['rabbit@ip-172-31-32-101']},
 {cluster_name, <<"rabbit@ip-172-31-32-101.us-east-2.compute.internal">>},
 {partitions, []}]
```