Distributed Cache Service

Guía del usuario

Edición 01

Fecha 2024-07-29

HUAWEI TECHNOLOGIES CO., LTD.

Marcas y permisos


y otras marcas registradas de Huawei pertenecen a Huawei Technologies Co., Ltd.
Todas las demás marcas registradas y los otros nombres comerciales mencionados en este documento son propiedad de sus respectivos titulares.

Aviso

Las funciones, los productos y los servicios adquiridos están estipulados en el contrato celebrado entre Huawei y el cliente. Es posible que la totalidad o parte de los productos, las funciones y los servicios descritos en el presente documento no se encuentren dentro del alcance de compra o de uso. A menos que el contrato especifique lo contrario, ninguna de las afirmaciones, informaciones ni recomendaciones contenidas en este documento constituye garantía alguna, ni expresa ni implícita.

La información contenida en este documento se encuentra sujeta a cambios sin previo aviso. En la preparación de este documento se realizaron todos los esfuerzos para garantizar la precisión de sus contenidos. Sin embargo, ninguna declaración, información ni recomendación contenida en el presente constituye garantía alguna, ni expresa ni implícita.

Índice

1 Antes de comenzar

1.1 Acceso y uso de DCS

Acceso a DCS

Puede acceder al Distributed Cache Service (DCS) desde la consola de gestión basada en web o mediante interfaces de programación de aplicaciones (API) RESTful a través de solicitudes HTTPS.

- Uso de la consola de gestión

  Inicie sesión en la **consola de gestión** y seleccione **Distributed Cache Service** en la lista de servicios.

  Para obtener más información sobre cómo usar la consola de DCS, consulte los capítulos de **Compra de instancias de DCS** a **Gestión de contraseñas**.

  Los datos de monitoreo de DCS son registrados por Cloud Eye. Para ver las métricas de monitoreo o configurar reglas de alarma, vaya a la consola de Cloud Eye. Para más detalles, consulte **Consulta de Métricas**.

  Si ha habilitado Cloud Trace Service (CTS), CTS registra las operaciones de instancia de DCS. Puede ver el historial de operaciones en la consola de CTS. Para más detalles, consulte **Consulta de logs de auditoría**.

- Uso de las API

  DCS proporciona las API RESTful para que pueda integrar DCS en su propio sistema de aplicaciones. Para obtener más información sobre las API de DCS y las llamadas a la API, consulte la ***Referencia de la API de Distributed Cache Service***.

AVISO

1. Todas las funciones disponibles se pueden utilizar en la consola. Algunas funciones también se pueden usar a través de las API. Para obtener más información sobre cómo usar las funciones a través de las API, consulte ***Referencia de la API de Distributed Cache Service***.
2. Para obtener más información sobre las API para la supervisión y auditoría, consulta la documentación del **Cloud Eye** y **Cloud Trace Service**.

Uso de DCS

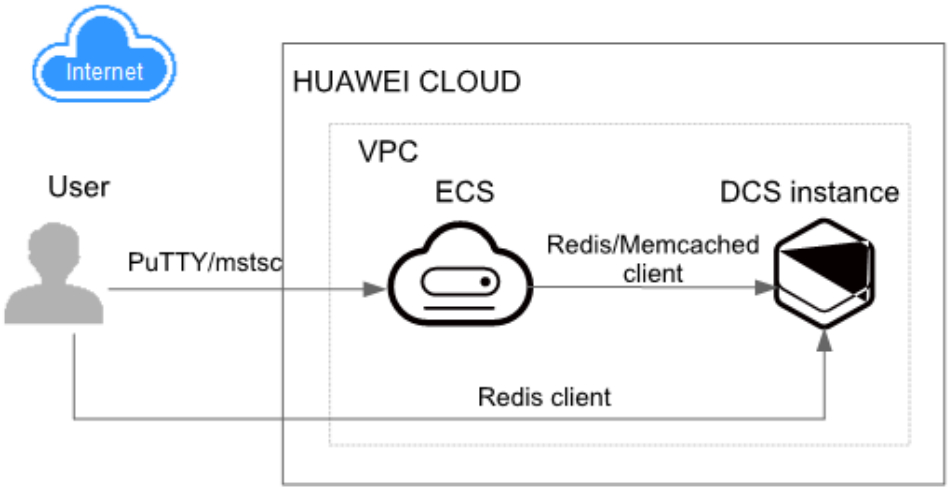
Después de comprar una instancia de DCS, acceda a ella haciendo referencia a **Acceso a una instancia de DCS para Redis**. Cualquier cliente que sea compatible con el protocolo de código abierto Redis o Memcached puede acceder respectivamente a una instancia de DCS compatible con Redis o Memcached. Después de acceder a una instancia de DCS, puede disfrutar de las operaciones de lectura/escritura rápidas habilitadas por DCS.

AVISO

DCS no implica información confidencial del usuario. Cuál, por qué, cuándo y cómo se procesan los datos con DCS deben cumplir con las leyes y regulaciones locales. Si es necesario transmitir o almacenar datos confidenciales, cifrar los datos antes de su transmisión o almacenamiento.

Para obtener más información sobre cómo acceder a una instancia de DCS, consulte **Figura 1-1**.

Figura 1-1 Accediendo a una instancia de DCS



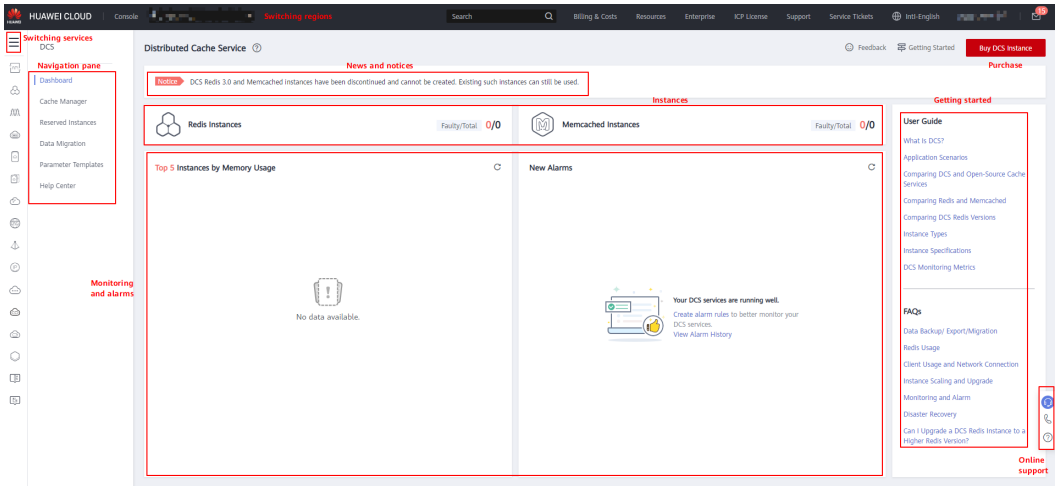
NOTA

- Actualmente, se puede acceder a una instancia de DCS a través de una red interna a través de un Elastic Cloud Server (ECS) que se encuentra en la misma nube privada virtual (VPC) que la instancia de DCS.
- Si el acceso público está habilitado, se puede acceder a una instancia de DCS Redis a través de una dirección IP elástica (EIP) a través de una red pública.

1.2 Uso de la consola de DCS

En la **consola de DCS**, puede comprar, usar y mantener instancias DCS, ver el estado de instancia y el uso de memoria, y buscar soporte en línea.

Figura 1-2 Consola de DCS



- Cambio de regiones

  Puede cambiar a una región más cercana a su aplicación.
- Servicios de conmutación

  Puede cambiar a consolas de otros servicios, como las consolas de VPC y Cloud Eye.
- Creación de una instancia

  Haga clic para comprar instancias de DCS compatible con Redis o con Memcached.
- Panel de navegación

  Esta área proporciona acceso a instancias de DCS operativas y a la migración de datos.
- Noticias y avisos

  Esta área le informa de las últimas funciones disponibles y ofertas especiales.
- Instancias

  Esta área muestra el número total de instancias y el número de instancias defectuosas del usuario actual.
- Monitoreo y alarmas

  Esta área muestra instancias con el mayor uso de memoria. Para obtener más información sobre cómo ver información sobre una instancia específica, consulte **Consulta de detalles de instancia**.

  Puede crear reglas de alarma para su instancia. Cuando se genera una alarma, puede manejarla inmediatamente. Para más detalles, consulte **Configuración de reglas de alarma para métricas críticas**.
- Tareas iniciales

  Al hacer clic en estos enlaces, se le dirigirá a la documentación para obtener más información sobre cómo usar DCS.
- Asistencia en línea

  Si tiene alguna pregunta mientras usa DCS, póngase en contacto con el soporte en línea.

2 Gestión de permisos

2.1 Creación de un usuario y concesión de permisos DCS

En este capítulo se describe cómo utilizar **IAM** para el control de permisos detallado para los recursos de DCS. Con IAM, usted puede:

- Crear usuarios de IAM para empleados en función de la estructura organizativa de su empresa. Cada usuario de IAM tendrá sus propias credenciales de seguridad para acceder a los recursos de DCS.
- Gestionar los permisos según el principio de permisos mínimos (PoLP).
- Confíe una cuenta de Huawei Cloud o un servicio en la nube para realizar operaciones eficientes en sus recursos DCS.

Si su cuenta de Huawei Cloud no requiere usuarios individuales de IAM, omita este capítulo.

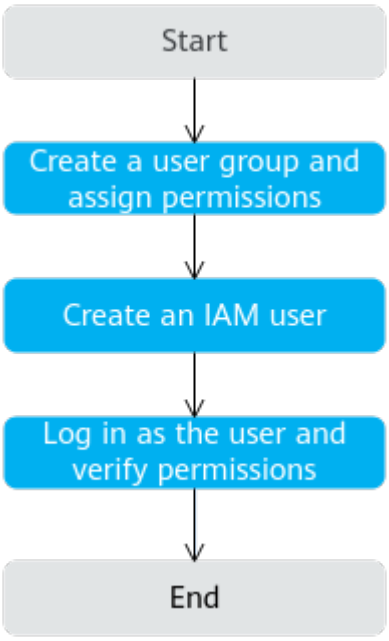
Esta sección describe el procedimiento para conceder el permiso **DCS ReadOnlyAccess** (véase **Figura 2-1**) como un ejemplo.

Prerrequisitos:

Obtener más información sobre los permisos (consulte **Roles y políticas definidas por el sistema compatibles con DCS**) compatibles con DCS y elija políticas o roles de acuerdo con sus requisitos. Para ver los permisos de otros servicios, consulte **Políticas de permisos**.

Flujo del proceso

Figura 2-1 Proceso de la concesión de permisos de DCS



1. **Crear un grupo de usuarios y asignar permisos**.

   Cree un grupo de usuarios en la consola de IAM y asigne la política **DCS ReadOnlyAccess** al grupo.
2. **Crear un usuario de IAM**.

   Cree un usuario en la consola de IAM y agregue el usuario al grupo creado en **1**.
3. **Iniciar sesión** y verificar los permisos.

   Inicie sesión en la consola DCS con el usuario recién creado y compruebe que el usuario solo tiene permisos de lectura para DCS.

2.2 Políticas personalizadas de DCS

Se pueden crear las políticas personalizadas para complementar las políticas definidas por el sistema de DCS. Para ver las acciones que se pueden agregar a las políticas personalizadas, consulte **Políticas de permisos y acciones admitidas**.

Puede crear las políticas personalizadas de cualquiera de las siguientes maneras:

- Visual editor: Seleccione servicios en la nube, acciones, recursos y condiciones de solicitud. Esto no requiere conocimiento de la sintaxis de políticas.
- JSON: Edite las políticas JSON desde cero o basándose en una política existente.

Para obtener más información, consulte **Creating a Custom Policy**. La siguiente sección contiene los ejemplos de políticas personalizadas DCS comunes.

NOTA

Debido al almacenamiento en caché de datos, una política que involucre acciones de OBS entrará en vigor cinco minutos después de que se adjunte a un usuario, grupo de usuarios o proyecto.

Ejemplo de las políticas personalizadas

- Ejemplo 1: Permitir a los usuarios eliminar y reiniciar las instancias de DCS y borrar datos de una instancia

```
{
    "Version": "1.1",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "
                    dcs:instance:delete
                    dcs:instance:modifyStatus
                 "
            ]
        }
    ]
}
```

- Ejemplo 2: Denegar la eliminación de instancia de DCS

  Una política con solo los permisos "Deny" debe usarse junto con otras políticas para que surtan efecto. Si los permisos asignados a un usuario contienen tanto "Allow" como "Deny", los permisos "Deny" tienen prioridad sobre los permisos "Allow".

  Por ejemplo, si desea asignar todos los permisos de la política **DCS FullAccess** a un usuario, excepto para eliminar instancias de DCS, puede crear una política personalizada para denegar solo la eliminación de instancias de DCS. Cuando se aplica tanto la política **DCS FullAccess** como la política personalizada que deniega la eliminación de instancia de DCS, ya que "Deny" siempre tiene prioridad sobre "Allow", se aplicará el permiso "Deny" para ese permiso en conflicto. A continuación, el usuario podrá realizar todas las operaciones en instancias de DCS, excepto eliminar instancias de DCS. A continuación se muestra un ejemplo de una política de denegación:

```
{
        "Version": "1.1",
        "Statement": [
                {
                        "Effect": "Deny",
                        "Action": [
                                "dcs:instance:delete"
                        ]
                }
        ]
}
```


3 Compra de instancias de DCS

3.1 Identificación de requisitos

Antes de comprar una instancia de DCS, identifique sus requisitos:

1. Decida el motor de caché necesario.

   Elija un motor de caché basado en los requisitos de servicio. El motor de caché no se puede cambiar una vez creada la instancia.

   – Para obtener más información acerca de los motores de caché Redis y Memcached, consulte **Descripción general de DCS**.
   – Para obtener más información sobre las diferencias entre Redis y Memcached, consulte **Comparación entre Redis y Memcached**.

2. Decida la versión del motor de caché requerida.

   Realice este paso si elige Redis como motor de caché.

   NOTA

   DCS for Redis 3.0 ya no se proporciona. Puede utilizar DCS for Redis 4.0 o 5.0 en su lugar.

   The high-performance edition uses Huawei proprietary lightweight LibOS. Las versiones diferentes de Redis tienen características diferentes. Para obtener más información, consulte **Comparación entre las versiones de Redis**.

3. Elija el tipo de instancia.

   DCS proporciona tipos de instancias de nodo único, principal/en standby, de Clúster Proxy y de Clúster Redis. Para obtener más información sobre las arquitecturas de instancia, consulte **Arquitectura de instancia de DCS**.

4. Decida la especificación de instancia requerida.

   Cada especificación especifica el máximo de la memoria disponible, el número de conexiones y el ancho de banda. Para obtener más información, consulte **Especificaciones de las instancias de DCS**.

5. Decida sobre la región y si se requiere el despliegue cruzado de la zona de disponibilidad.

   Elija una región más cercana a su aplicación para reducir la latencia.

   Una región consiste en varias zonas de disponibilidad (AZ) con fuentes de alimentación y redes físicamente aisladas. Las instancias principal/en standby y de clúster de DCS se

pueden implementar en AZ. Las aplicaciones también se pueden implementar en AZ para lograr alta disponibilidad (HA) tanto para datos como para aplicaciones.

NOTA

- Si se implementa una instancia principal/en standby o de clúster de DCS a través de AZ, los errores en una AZ no afectan a los nodos de caché en otras AZ. Esto se debe a que cuando el nodo principal es defectuoso, el nodo de memoria caché en standby se convertirá automáticamente en el nodo principal para proporcionar servicios. Dicha implementación logra una mejor recuperación ante desastres.
- La implementación de una instancia de DCS en las AZ reduce ligeramente la eficiencia de la red en comparación con la implementación de una instancia dentro de una AZ. Por lo tanto, si se implementa una instancia de DCS a través de AZ, la sincronización entre los nodos de caché principal y en standby es ligeramente menos eficiente.

6. Decida si se requieren políticas de copia de seguridad.

   Actualmente, las políticas de backup solo se pueden configurar para instancias de DCS principal/en standby, de Clúster Proxy y de Clúster Redis. Para obtener más información sobre la copia de seguridad y la restauración, consulte **Descripción general**.

3.2 Preparación de los recursos requeridos

Descripción general

Antes de crear una instancia de DCS, prepare los recursos necesarios, incluidas las reglas de VPC, subred, grupo de seguridad y grupo de seguridad. Cada instancia de DCS se implementa en una VPC y se enlaza a una subred y grupo de seguridad específicos, que proporcionan un entorno de red virtual aislado y políticas de protección de seguridad que puede configurar y administrar fácilmente.

Si ya tiene una VPC, una subred y un grupo de seguridad, puede usarlos para todas las instancias de DCS que cree posteriormente.

Recursos requeridos

En la siguiente tabla se enumeran los recursos requeridos por una instancia de DCS.

Tabla 3-1 Recursos de dependencia de una instancia de DCS

Recurso	Requisito	Operaciones
VPC y subred	Las diferentes instancias de DCS pueden usar las mismas VPC y subredes o diferentes según los requisitos del sitio. Tenga en cuenta lo siguiente al crear una VPC y una subred: ● La VPC y la instancia de DCS deben estar en la misma región. ● Conservar la configuración predeterminada a menos que se especifique lo contrario.	Para obtener más información sobre cómo crear una VPC y una subred, consulte **Creación de la VPC**. Si necesita crear y utilizar una nueva subred en una VPC existente, consulte **Creación de una subred para la VPC**.

Recurso	Requisito	Operaciones
Grupo de seguridad **NOTA** Los grupos de seguridad solo son requeridos por las instancias de DCS compatibles con Redis 3.0 y con Memcached.	Las diferentes instancias de DCS pueden utilizar el mismo grupo de seguridad o diferentes grupos de seguridad. Tenga en cuenta lo siguiente al crear un grupo de seguridad: ● Establezca **Template** en **Custom**. ● Después de crear un grupo de seguridad, conserve las reglas entrantes y salientes predeterminadas. ● Para utilizar DCS, debe agregar las reglas de grupo de seguridad descritas en **Tabla 3-2**. También puede agregar otras reglas basadas en los requisitos del sitio.	Para obtener más información sobre cómo crear un grupo de seguridad, consulte **Creación de un grupo de seguridad**. Para obtener más información sobre cómo agregar reglas a un grupo de seguridad, consulte **Adición de una regla del grupo de seguridad** .
EIP (opcional) **NOTA** Los EIP solo son compatibles con las instancias de DCS para Redis 3.0.	Si desea acceder a DCS a través de una red pública, asigne un EIP.	Para obtener más información sobre cómo asignar un EIP, consulte **Asignar un EIP**.

Tabla 3-2 Reglas de grupos de seguridad

Dirección	Protocolo	Puerto	Origen	Descripción
Entrante	TCP	36379	0.0.0.0/0	Acceda a una instancia de DCS Redis (con la encriptación SSL habilitada) a través de una red pública.
Entrante	TCP	6379	0.0.0.0/0	Acceda a una instancia de DCS Redis (con la encriptación SSL desactivada) a través de una red pública.
Entrante	TCP	6379	0.0.0.0/0	Acceda a una instancia de DCS Redis en una red privada. (La encriptación SSL no es compatible.)

Dirección	Protocolo	Puerto	Origen	Descripción
Entrante	TCP	11211	0.0.0.0/0	Acceda a una instancia de DCS Memcached a través de una red pública. (La encriptación SSL no es compatible.)

3.3 Compra de una instancia de DCS Redis

Puede comprar una o más instancias de DCS Redis con las capacidades informáticas y el espacio de almacenamiento requeridos según los requisitos de servicio.

NOTA

- DCS for Redis 3.0 ya no se proporciona. Puede utilizar DCS for Redis 4.0 o 5.0 en su lugar.
- Las versiones de Redis y los tipos de instancia compatibles varían según las regiones.

Prerrequisitos:

- Para lograr una gestión detallada de sus recursos de Huawei Cloud, cree grupos de usuarios y usuarios de IAM y conceda permisos específicos a los usuarios. Para más detalles, consulte **Gestión de permisos**.
- Usted ha preparado los **recursos necesarios**.

Procedimiento

Paso 1 Inicie sesión en la **consola DCS**.

Paso 2 Haga clic en en la esquina superior izquierda de la consola de gestión y seleccione una región.

NOTA

Seleccione la misma región que su servicio de aplicación.

Paso 3 Haga clic en **Buy DCS Instance** en la esquina superior derecha.

Paso 4 Especifique **Billing Mode**.

Paso 5 Seleccione una región más cercana a su aplicación para reducir la latencia y acelerar el acceso.

Paso 6 Especifique los siguientes parámetros de instancia en función de la información recopilada en **Identificación de requisitos**.

1. **Cache Engine**:

   Seleccione **Redis**.

2. **Version**:

   Versiones de Redis soportadas actualmente: 5.0, 4.0 y 3.0

3. Establezca **Instance Type** en **Single-node**, **Master/Standby**, **Proxy Cluster**, **Redis Cluster**, o **Read/Write splitting**.

4. Establezca **CPU Architecture** en **x86** o **Arm**.
5. Establecer **Replicas**. El valor predeterminado es **2**.

   Este parámetro sólo se muestra cuando se selecciona Redis 4.0 o Redis 5.0 y el tipo de instancia es principal/en standby o de Clúster Redis.
6. Si se selecciona **Proxy Cluster** o **Redis Cluster**, se muestra el parámetro **Sharding**.

   Use default: utilice las especificaciones de fragmento por defecto.

   Customize: Personaliza el tamaño de cada partición y, a continuación, selecciona las especificaciones de instancia correspondientes.
7. Seleccione una AZ.

   NOTA

   Para acelerar el acceso, implemente su instancia y su aplicación en la misma AZ.

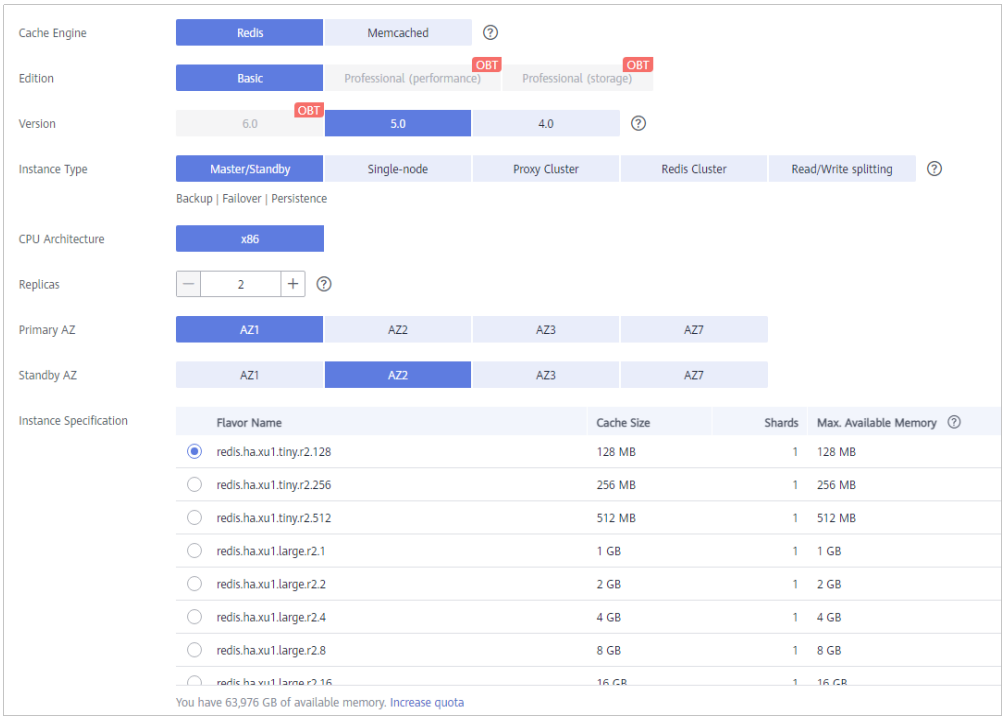
   Si el tipo de instancia es principal/en standby, Clúster Proxy o Clúster Redis, **AZ** se convierte en **Primary AZ** y se muestra **Standby AZ**. Seleccione una AZ para los nodos principal y en standby de la instancia.
8. **Instance Specification**:

   La cuota predeterminada se muestra en la consola.

   Para aumentar la cuota, haga clic en **Increase quota** debajo de las especificaciones. En la página mostrada, rellene un formulario de solicitud de cuota y haga clic en **Submit**.

   Figura 3-1 muestra la configuración del parámetro de instancia.

Figura 3-1 Compra de una instancia de DCS Redis



Paso 7 Configure los parámetros de red de instancia.

1. Seleccione una VPC y una subred.
2. Configure la dirección IP.

Las instancias de Clúster Redis solo admiten direcciones IP asignadas automáticamente. Los otros tipos de instancia admiten direcciones IP asignadas automáticamente y direcciones IP especificadas manualmente. Puede especificar manualmente una dirección IP privada para su instancia según sea necesario.

Para Redis 4.0/5.0, puede especificar una numeración de puertos en el rango de 1 a 65535. Si no se especifica ningún puerto, se utilizará el puerto predeterminado 6379.

Para Redis 3.0, no puede personalizar un puerto. Se utilizará el puerto predeterminado 6379.

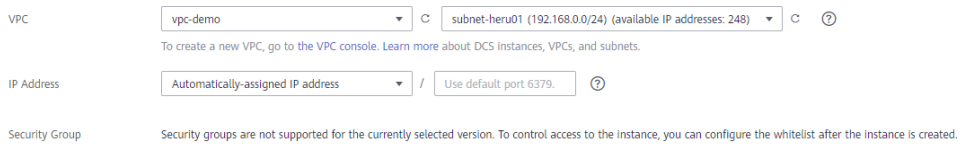
3. Seleccione un grupo de seguridad.

   Un grupo de seguridad es un conjunto de reglas que controlan el acceso a los ECS. Proporciona políticas de acceso para ECS de confianza mutua con los mismos requisitos de protección de seguridad en la misma VPC.

   DCS for Redis 4.0/5.0 se basa en VPC Endpoint y no requiere grupos de seguridad.

   Si el puerto 6379 no está habilitado para el grupo de seguridad seleccionado, se muestra la casilla de verificación **Enable port 6379** y se selecciona de forma predeterminada, lo que indica que después de crear la instancia, el puerto 6379 se habilitará para el grupo de seguridad seleccionado. Si el puerto 6379 no está habilitado para el grupo de seguridad seleccionado, las conexiones a la instancia pueden fallar.

Figura 3-2 Configuración de parámetros de red de instancia



Paso 8 Establezca la contraseña de la instancia.

- Seleccione **Yes** o **No** para **Password Protected**.

  NOTA

  - El acceso sin contraseña conlleva los riesgos de seguridad. Tenga cuidado al seleccionar este modo.
  - Si desea habilitar el acceso público para una instancia de DCS Redis 3.0, debe seleccionar el modo protegido con contraseña y establecer una contraseña.
  - Después de crear una instancia de DCS Redis para acceder en modo sin contraseña, puede establecer una contraseña para ella mediante la función de restablecimiento de contraseña. Para más detalles, consulte **Cambio de la configuración de contraseña para instancias de DCS Redis**.

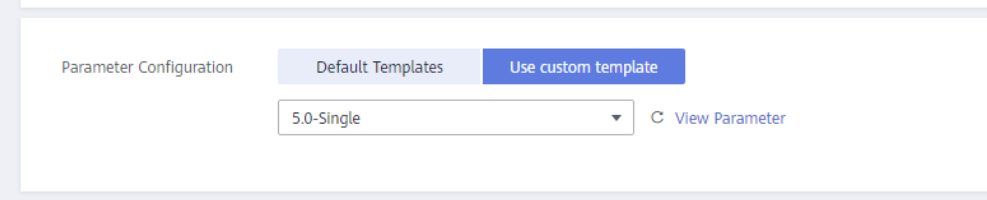
- **Password** y **Confirm Password**: Estos parámetros indican la contraseña de acceso a la instancia de DCS Redis y solo se muestran cuando **Password Protected** está establecido en **Yes**.

  NOTA

  Por motivos de seguridad, si el acceso sin contraseña está deshabilitado, el sistema le solicitará que introduzca una contraseña específica de la instancia cuando acceda a la instancia de DCS Redis. Mantenga su contraseña de instancia segura y cámbiela periódicamente.

Paso 9 Configurar **Parameter Configuration**.

Puede seleccionar **Default Templates** o **Use custom template**.



NOTA

- En la página de creación de instancia, el parámetro predeterminado modelos se utiliza de forma predeterminada.
- Puede seleccionar un modelo personalizado solo para instancias de Redis. La versión del motor de caché y el tipo de instancia seleccionados deben coincidir con los del modelo.

Paso 10 Especifique la duración y la cantidad de instancia requeridas para la facturación anual/mensual.

Paso 11 Introduzca un nombre de instancia y seleccione un proyecto de empresa.

Cuando se crea una sola instancia a la vez, el valor de **Name** puede contener entre 4 y 64 caracteres. Cuando crea más de una instancia a la vez, el valor de **Name** puede contener entre 4 y 56 caracteres. Estas instancias se nombran en el formato de "*name-n*", en el que *n* comienza desde 000 y se incrementa en 1. Por ejemplo, si crea dos instancias y establece **Name** en **dcs_demo**, las dos instancias se denominan respectivamente como **dcs_demo-000** y **dcs_demo-001**.

Paso 12 Haga clic en **More Settings** para mostrar más configuraciones, incluida la política de copia de seguridad y el cambio de nombre de comandos críticos.

1. Introduzca una descripción de la instancia.
2. Especifique la política de copia de seguridad.

   Este parámetro sólo se muestra cuando el tipo de instancia es principal/en standby o de clúster. Para obtener más información sobre cómo configurar una política de copia de seguridad, consulte **Copia de seguridad y restauración de instancias**.
3. Cambiar el nombre de los comandos críticos.

   Si se selecciona Redis 4.0/5.0, se muestra el parámetro **Command Renaming**. Actualmente, solo puede cambiar el nombre de los comandos **COMMAND**, **KEYS**, **FLUSHDB**, **FLUSHALL**, y **HGETALL**.
4. Especifique la ventana de mantenimiento.

   Elija una ventana para que el personal de DCS O&M realice el mantenimiento de su instancia. Se le contactará antes de realizar cualquier actividad de mantenimiento.
5. Agregue una etiqueta.

   Las etiquetas se utilizan para identificar los recursos de la nube. Cuando tiene muchos recursos de nube del mismo tipo, puede usar etiquetas para clasificar los recursos de nube por dimensión (por ejemplo, por uso, propietario o entorno).

   – Si ha creado etiquetas predefinidas, seleccione un par predefinido de clave y valor de etiqueta. Haga clic en **View predefined tags**. En la consola del Tag Management Service (TMS), vea etiquetas predefinidas o cree nuevas etiquetas.

   – También puede agregar una etiqueta introduciendo la clave y el valor de la etiqueta. Para obtener más información sobre los requisitos de nombres de etiquetas, consulte **Gestión de etiquetas**.

Paso 13 Haga clic en **Next**.

La página mostrada muestra la información de instancia que ha especificado.

Paso 14 Confirme la información de la instancia y envíe la solicitud.

Paso 15 Vuelva a la página **Cache Manager** para ver y gestionar las instancias de DCS.

----Fin

3.4 Compra de una instancia de DCS Memcached (no disponible pronto)

Puede comprar una o más instancias de DCS Memcached con las capacidades informáticas y el espacio de almacenamiento requeridos según los requisitos de servicio.

NOTA

DCS for Memcached está a punto de no estar disponible y ya no se vende en algunas regiones. Puede utilizar DCS for Redis 4.0 o 5.0 en su lugar.

Compra de una instancia de DCS Memcached

Paso 1 Inicie sesión en la **consola DCS**.

Paso 2 Haga clic en en la esquina superior izquierda de la consola de gestión y seleccione una región.

NOTA

Seleccione la misma región que su servicio de aplicación.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Paso 4 Haga clic en **Buy DCS Instance** en la esquina superior derecha.

Paso 5 Especifique **Billing Mode**.

Paso 6 Seleccione una región más cercana a su aplicación para reducir la latencia y acelerar el acceso.

Paso 7 Especifique los siguientes parámetros de instancia en función de la información recopilada en **Identificación de requisitos**.

1. Establezca **Cache Engine** en **Memcached**.
2. Establezca **Instance Type** en **Single-node** o **Master/Standby**.
3. Seleccione una AZ.

   NOTA

   Para acelerar el acceso, implemente su instancia y su aplicación en la misma AZ. Para garantizar la fiabilidad de los datos, implemente en diferentes AZ.

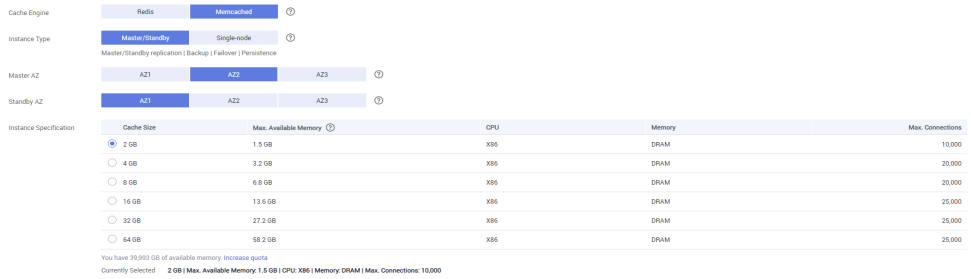
   Si el tipo de instancia es principal/en standby, se muestra **Standby AZ**. Seleccione una AZ en standby para el nodo en standby de la instancia.
4. Especifique **Instance Specification**.

La cuota predeterminada se muestra en la consola.

Para aumentar la cuota, haga clic en **Increase quota** debajo de las especificaciones. En la página mostrada, rellene un formulario de solicitud de cuota y haga clic en **Submit**.

Figura 3-3 muestra la configuración del parámetro de instancia.

Figura 3-3 Compra de una instancia de DCS Memcached



Paso 8 Configure los parámetros de red de instancia.

1. Para **VPC**, seleccione una VPC, una subred y especifique la dirección IP.

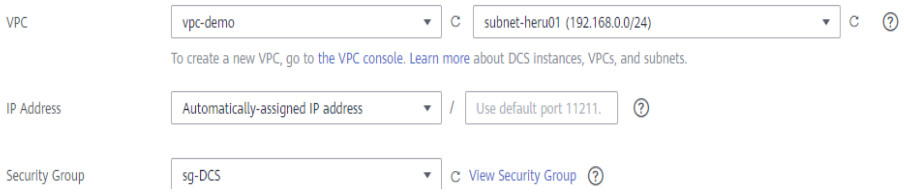
   Puede optar por obtener una dirección IP asignada automáticamente o especificar manualmente una dirección IP que esté disponible en la subred seleccionada.

2. Seleccione un grupo de seguridad.

   Un grupo de seguridad es un conjunto de reglas que controlan el acceso a los ECS. Proporciona políticas de acceso para ECS de confianza mutua con los mismos requisitos de protección de seguridad en la misma VPC.

   Si el puerto 11211 no está habilitado para el grupo de seguridad seleccionado, se muestra la casilla de verificación **Enable port 11211** y se selecciona de forma predeterminada, lo que indica que después de crear la instancia, el puerto 11211 se habilitará para el grupo de seguridad seleccionado. Si el puerto 11211 no está habilitado para el grupo de seguridad seleccionado, las conexiones a la instancia pueden fallar.

Figura 3-4 Configuración de parámetros de red de instancia



Paso 9 Establezca la contraseña de la instancia.

- Seleccione **Yes** o **No** para **Password Protected**.

  NOTA

  - El acceso sin contraseña conlleva los riesgos de seguridad. Tenga cuidado al seleccionar este modo.
  - Después de crear una instancia de DCS Memcached en modo protegido con contraseña, puede restablecer la contraseña o cambiarla en modo sin contraseña. Para más detalles, consulte **Cambio de la configuración de contraseña para instancias de DCS Memcached**.
  - Si se deshabilita el acceso sin contraseña, se debe acceder a las instancias de Memcached de DCS mediante el protocolo binario de Memcached y mediante autenticación SASL.

- Nombre de usuario requerido para acceder a la nueva instancia de DCS.

> **NOTA**
>
> Este parámetro sólo se muestra cuando **Password Protected** está establecido en **Yes**.

- **Password** y **Confirm Password**: Estos parámetros indican la contraseña de acceso a la instancia de DCS Memcached y solo se muestran cuando **Password Protected** está establecido en **Yes**.

  > **NOTA**
  >
  > Por motivos de seguridad, si el acceso sin contraseña está deshabilitado, el sistema le solicitará que introduzca una contraseña específica de la instancia cuando acceda a la instancia de DCS Memcached. Mantenga su contraseña de instancia segura y cámbiela periódicamente.

Paso 10 Especifique la duración y la cantidad requeridas.

Paso 11 Introduzca un nombre de instancia y seleccione un proyecto de empresa.

Cuando se crea una sola instancia a la vez, el valor de **Name** puede contener entre 4 y 64 caracteres. Cuando crea más de una instancia a la vez, el valor de **Name** puede contener entre 4 y 56 caracteres. Estas instancias se nombran en el formato de "*name-n*", en el que *n* comienza desde 000 y se incrementa en 1. Por ejemplo, si crea dos instancias y establece **Name** en **dcs_demo**, las dos instancias se denominan respectivamente como **dcs_demo-000** y **dcs_demo-001**.

Paso 12 Haga clic en **More Settings** para mostrar más configuraciones, incluidas las etiquetas de instancia y de política de copia de seguridad.

1. Introduzca una descripción de la instancia.
2. Especifique la política de copia de seguridad.

   Este parámetro sólo se muestra cuando el tipo de instancia es principal/en standby. Para obtener más información sobre cómo configurar una política de copia de seguridad, consulte **Copia de seguridad y restauración de instancias**.
3. Especifique la ventana de mantenimiento.

   Especifique un período para que el personal de DCS O&M mantenga su instancia. Por ejemplo, si elige 02:00-03:00, los nodos de instancia se mantendrán durante este período.
4. Agregue una etiqueta.

   Las etiquetas se utilizan para identificar los recursos de la nube. Cuando tiene muchos recursos de nube del mismo tipo, puede usar etiquetas para clasificar los recursos de nube por dimensión (por ejemplo, por uso, propietario o entorno).

   – Si ha creado etiquetas predefinidas, seleccione un par predefinido de clave y valor de etiqueta. Haga clic en **View predefined tags**. En la consola del Tag Management Service (TMS), vea etiquetas predefinidas o cree nuevas etiquetas.

   – También puede crear nuevas etiquetas especificando **Tag key** y **Tag value**.

   Se pueden agregar hasta 10 etiquetas a cada instancia de DCS. Para obtener más información sobre los requisitos de las etiquetas, consulte **Gestión de etiquetas**.

Paso 13 Haga clic en **Next**.

La página mostrada muestra la información de instancia que ha especificado.

Paso 14 Confirme la información de la instancia y haga clic en **Submit**.

Paso 15 Vuelva a la página **Cache Manager** para ver y gestionar las instancias de DCS.

1. Se tarda de 5 a 15 minutos en crear una instancia de DCS.

2. Una vez que se ha creado correctamente una instancia de DCS, entra en el estado de **Running** de forma predeterminada.

----Fin

4 Acceso a una instancia de DCS para Redis

4.1 Restricciones

Puede acceder a una instancia de DCS a través de cualquier cliente de Redis. Para obtener más información sobre los clientes de Redis, consulte el **sitio web oficial de Redis**.

NOTA

DCS for Redis 3.0 ya no se proporciona. Puede utilizar DCS for Redis 4.0 o 5.0 en su lugar.

- Para acceder a una instancia de DCS Redis a través de un cliente en un ECS en la misma VPC que la instancia, tenga en cuenta que:
  - Una instancia de ECS y de DCS pueden comunicarse entre sí solo cuando pertenecen a la misma VPC.
    - Redis 3.0: la instancia y el ECS deben configurarse con el mismo grupo de seguridad o usar diferentes grupos de seguridad, pero pueden comunicarse entre sí según lo configurado por las reglas de grupo de seguridad.
    - Redis 4.0/5.0: La dirección IP del ECS debe estar en la lista blanca de la instancia DCS.
  - Si las instancias de ECS y de DCS Redis están en las VPC diferentes, establezca una conexión de pares de VPC para lograr conectividad de red entre las instancias de ECS y de DCS. Para obtener más información, consulte **¿Soporta DCS el acceso entre VPC?**
- Antes de acceder a una instancia de DCS Redis 3.0 a través de redes públicas, asegúrese de que:

  Las reglas de grupo de seguridad se han configurado correctamente para la instancia. Si la encriptación SSL está deshabilitada, permita el acceso público a través del puerto 6379. Si la encriptación SSL está habilitada, permita el acceso público a través del puerto 36379. Para obtener más información, consulte **¿Cómo configuro un grupo de seguridad?**
- Si el cliente y la instancia de DCS Redis no están en la misma región, el nombre de dominio de instancia no se puede resolver entre regiones y no se puede acceder a la instancia en su dirección de nombre de dominio. Puede asignar manualmente el nombre de dominio a la dirección IP del archivo **hosts** o acceder a la instancia en su dirección IP.

4.2 Acceso público a una instancia de DCS Redis 3.0 (no disponible)

4.2.1 Paso 1: Compruebe si se admite el acceso público

Puede acceder a una instancia de DCS Redis 3.0 a través de redes públicas. En comparación con el acceso intra-VPC, el acceso público puede traer pérdida de paquetes, fluctuación y mayor latencia. Por lo tanto, se recomienda habilitar el acceso público solo durante las fases de desarrollo y prueba del servicio.

Antes de conectarse a una instancia de DCS a través de redes públicas, compruebe si la instancia admite el acceso público.

- Redis 3.0

  Actualmente, solo las instancias de DCS Redis 3.0 admiten el acceso público. Puede habilitar o deshabilitar el acceso público.

- Redis 4.0 y Redis 5.0

  Las instancias de DCS Redis 4.0 y 5.0 no admiten el acceso público. Si se requiere acceso público para una instancia de nodo único, principal/en standby o de Clúster Proxy, utilice Nginx para redirigir conexiones a través de un ECS configurado con la misma VPC y grupo de seguridad que la instancia de DCS. Para obtener más información, consulte **Uso de Nginx para acceder a instancias de DCS Redis 4.0 o 5.0**.

  No se puede acceder a instancias de Clúster Redis a través de redes públicas.

- Memcached

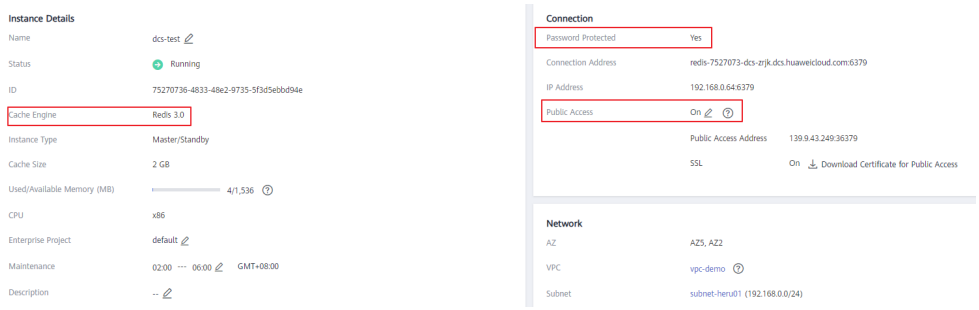
  Las instancias de DCS Memcached no admiten el acceso público. El ECS que sirve como cliente y la instancia de DCS a la que accederá el cliente deben pertenecer a la misma VPC. En la fase de desarrollo y depuración de aplicaciones, también puede usar un agente SSH para acceder a instancias de DCS Memcached en el entorno local.

Procedimiento

En la página **Basic Information** de la instancia, compruebe la siguiente configuración de parámetros:

- **Cache Engine**: Debe ser **Redis 3.0**. Si no es así, no se admite el acceso a la red pública.
- **Password Protected**: Debe ser **Yes**. Si no es así, habilite la protección con contraseña para la instancia haciendo referencia a **FAQs**.
- **Public Access**: Debe estar **On**. Si no es así, habilite el acceso público haciendo referencia a **Paso 2: Habilite el acceso público para una instancia de DCS Redis**.

Figura 4-1 Comprobación de la versión del motor de caché, la protección con contraseña y el acceso público



FAQs

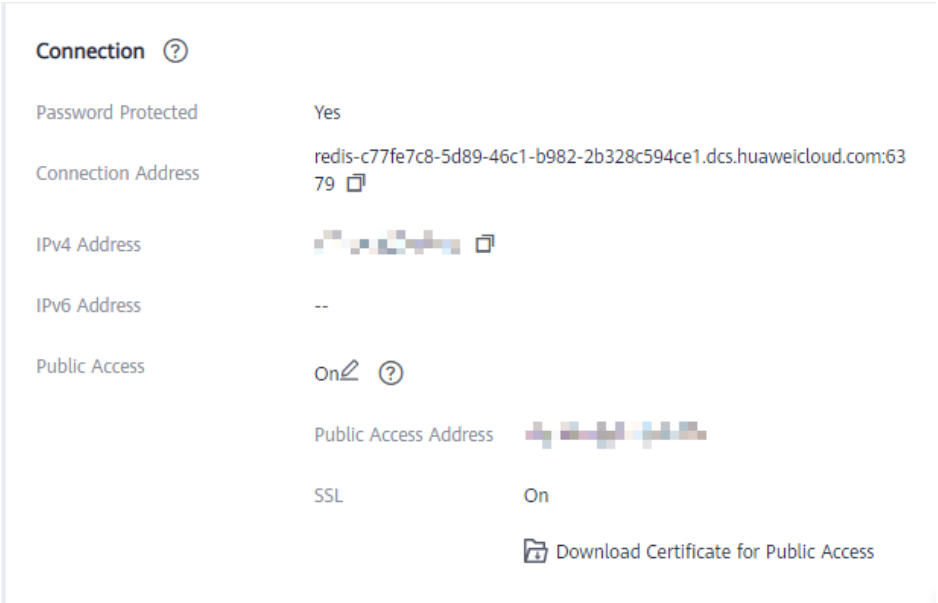
- ¿Qué puedo hacer si el botón de acceso público está atenuado cuando la instancia no está protegida con contraseña?

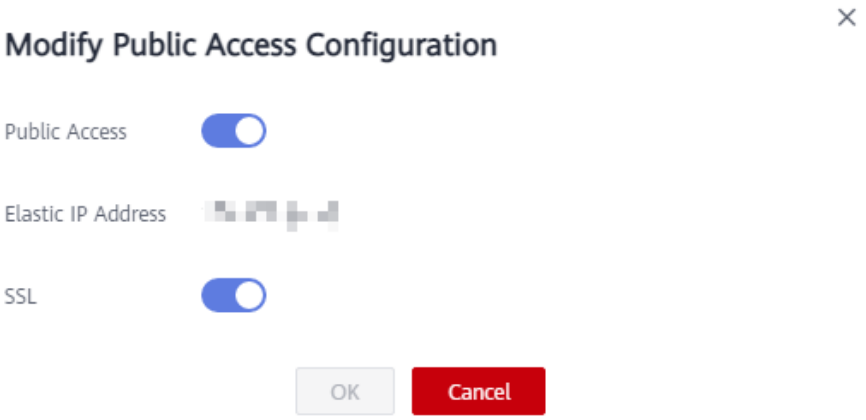
  En la esquina superior derecha de la página **Basic Information**, elija **More** > **Reset Password**. Después de restablecer la contraseña, el parámetro **Password Protected** cambia a **Yes**. Ahora se puede hacer clic en el botón de acceso público.

- ¿Cómo desactivo la encriptación SSL cuando se ha habilitado el acceso público?

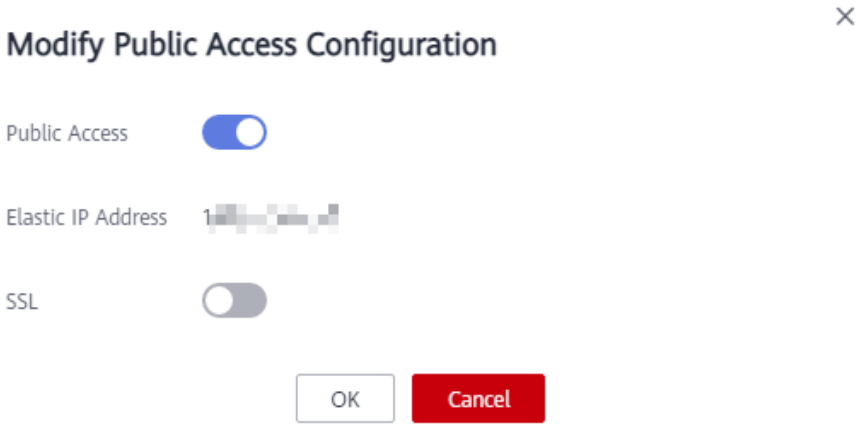
  La encriptación SSL está habilitada de forma predeterminada cuando se habilita el acceso público. Para deshabilitar la encriptación SSL, realice los siguientes pasos:

  a. Abra la página para configurar el acceso público.





b. Deshabilite la encriptación SSL y haga clic en **OK**.



c. En el área **Connection** de la página de detalles de la instancia, **SSL** está deshabilitado.

4.2.2 Paso 2: Habilite el acceso público para una instancia de DCS Redis

Si se ha habilitado el acceso público para la instancia, omita esta sección.

Si el acceso público no está habilitado, siga las instrucciones de esta sección. Puede habilitar o deshabilitar la encriptación SSL al habilitar el acceso público.

NOTA

- Antes de acceder a una instancia DCS a través de una red pública (con la encriptación SSL), descargue un certificado de CA para verificar el certificado de la instancia por motivos de seguridad.
- Cuando se accede a una instancia DCS a través de una red pública (sin encriptación SSL), se accede al EIP y al puerto 6379 de la instancia. No necesita descargar certificados ni instalar Stunnel en su cliente.
- Se recomienda habilitar SSL para cifrar los datos transmitidos entre su cliente de Redis y la instancia de DCS para evitar fugas de datos.

Procedimiento

Paso 1 Inicie sesión en la **consola DCS**.

Paso 2 Haga clic en  en la esquina superior izquierda de la consola de gestión y seleccione una región.

NOTA

Seleccione la misma región que su servicio de aplicación.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Paso 4 Haga clic en el nombre de la instancia de DCS Redis que desea configurar. Se muestra una página con información básica sobre la instancia de DCS.

Paso 5 Haga clic en  a la derecha de **Public Access**.

Paso 6 Haga clic en  para habilitar el acceso público.

Paso 7 Seleccione un EIP en la lista desplegable **Elastic IP Address**.

Si no hay EIP disponibles, haga clic en **View Elastic IP** para crear un EIP en la consola de red. Después de crear un EIP, haga clic en el botón de actualización a la derecha de **Elastic IP Address** para seleccionar el nuevo EIP.

Paso 8 (Opcional) Habilitar o deshabilitar SSL según sea necesario.

Se recomienda habilitar SSL para cifrar los datos transmitidos entre su cliente de Redis y la instancia de DCS para evitar fugas de datos.

Paso 9 Haz clic en **OK**.

Se tarda de 1 a 2 minutos para permitir el acceso público.

Se le redirigirá automáticamente a la página **Background Tasks**, donde se muestra el progreso de la tarea actual. Si el estado de la tarea es **Succeeded**, el acceso público se habilita correctamente.

----Fin

4.2.3 Paso 3: Acceda a una instancia de DCS Redis en Windows

Esta sección describe cómo acceder a una instancia de DCS Redis 3.0 a través de una red pública mediante redis-cli en Windows.

El acceso público ayuda al personal de I+D a establecer un entorno local para el desarrollo o las pruebas, mejorando la eficiencia del desarrollo. Sin embargo, en el entorno de producción (entorno oficial), acceda a una instancia de DCS Redis a través de una VPC para garantizar un acceso eficiente.

Prerrequisitos:

Antes de utilizar redis-cli para acceder a una instancia de DCS Redis a través de una red pública, asegúrese de que:

- La versión de instancia es Redis 3.0 y se ha habilitado el acceso público.

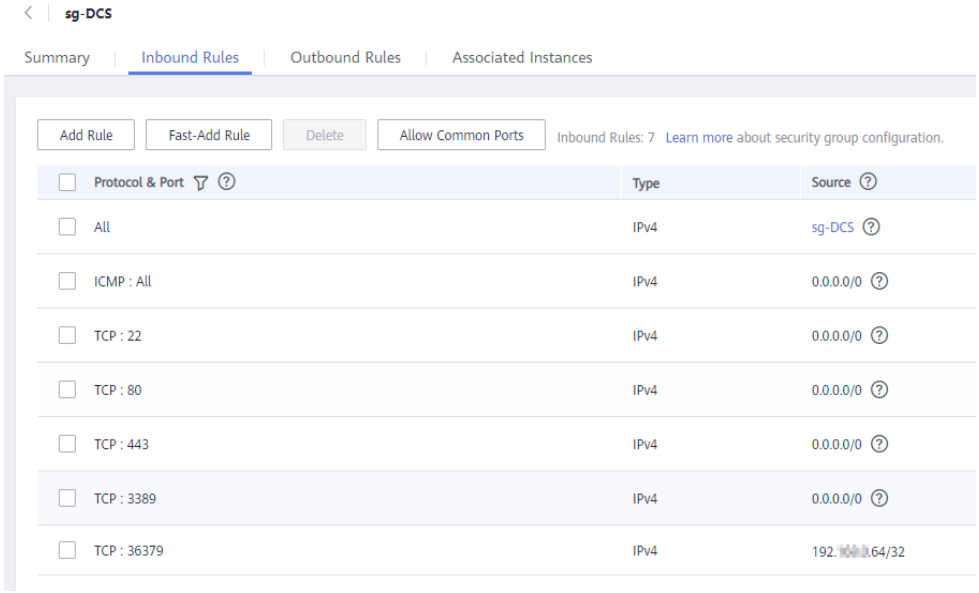
- Si se necesitan certificados para acceder a la instancia de DCS, descargue el certificado desde la página de detalles de la instancia de DCS. Para más detalles, consulte **Consulta de detalles de instancia**.

Conexión a Redis con la encriptación SSL

Paso 1 Asegúrese de que la regla de grupo de seguridad permita el acceso público a través del puerto 36379.

Cuando el encriptación SSL está habilitado, permita el acceso público a través del puerto 36379 e instale el cliente Stunnel.

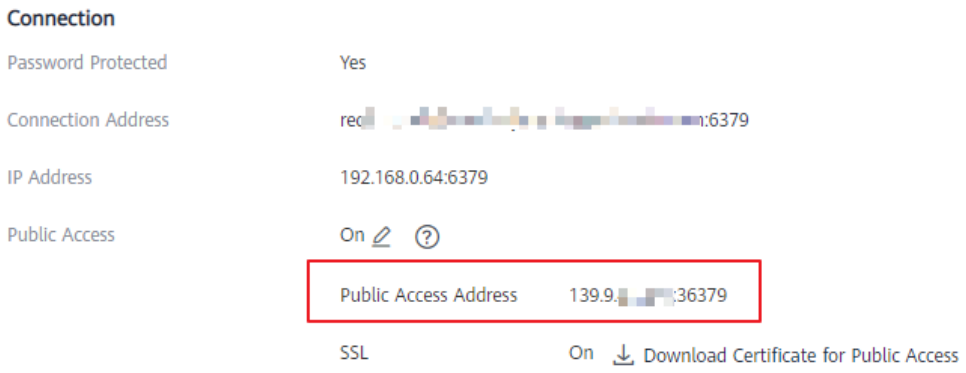
Figura 4-2 Regla de grupo de seguridad (puerto 36379)



Paso 2 Obtenga la dirección de acceso público y los certificados de la instancia en la página **Basic Information** de la instancia.

- La dirección de acceso público se muestra en la sección **Connection**.
- Los certificados se pueden descargar haciendo clic en **Download Certificate for Public Access** en la sección **Connection**. Después de la descompresión, obtendrá **dcs-ca.cer** (el certificado de clave pública en formato binario) y **dcs-ca-bundle.pem** (el archivo de certificado en formato de texto).

Figura 4-3 Consulta de la dirección de acceso público (SSL habilitado; puerto 36379)



Paso 3 Descargue el paquete de instalación más reciente de Windows Stunnel (por ejemplo, **stunnel-5.44-win32-installer.exe**) desde **https://www.stunnel.org/downloads.html** en el dispositivo local de Windows.

Paso 4 Ejecute el programa de instalación Stunnel e instale el cliente Stunnel.

Paso 5 Configurar el cliente Stunnel: Haga clic con el botón derecho en en la barra de tareas y elija **Edit Configuration**. Agregue la siguiente configuración y, a continuación, guarde y salga.

```
[redis-client]
client = yes
CAfile = D:\tmp\dcs\dcs-ca.cer
accept = 8000
connect = {public access address}
```

En la configuración:

- **client**: indica Stunnel. El valor fijo es **yes**.
- **CAfile**: especifica un certificado de CA, que es opcional. Si se requiere un certificado de CA, descargue y descomprima el certificado **dcs-ca.cer** como se indica en **Paso 2**. Si no es necesario, elimine este parámetro.
- **accept**: especifica el número de puerto de escucha definido por el usuario de Stunnel. Especifique este parámetro al acceder a una instancia de DCS mediante un cliente de Redis.
- **connect**: especifica la dirección de servicio y el número de puerto de Stunnel. Establezca este parámetro en la dirección de acceso público de instancia obtenida en **Paso 2**.

Cuando la encriptación SSL está habilitada, la configuración es similar a la siguiente:

```
[redis-client]
client = yes
CAfile = D:\tmp\dcs\dcs-ca.cer
accept = 8000
connect = 49.**.**.211:36379
```

Paso 6 Haga clic con el botón derecho en l en la barra de tareas y elija **Reload Configuration**.

Paso 7 Abra la herramienta CLI **cmd.exe** y ejecute el siguiente comando para comprobar si 127.0.0.1:8000 se está escuchando:

netstat -an |find "8000"

Supongamos que el puerto **8000** está configurado como el puerto de escucha en el cliente.

Si se muestra **127.0.0.1:8000** en el resultado devuelto y su estado es **LISTENING**, el cliente Stunnel se está ejecutando correctamente. Cuando el cliente de Redis se conecta a la dirección **127.0.0.1:8000**, Stunnel reenviará las solicitudes a la instancia de DCS Redis.

Paso 8 Acceda a la instancia de DCS Redis.

1. Obtenga y descomprima el paquete de instalación del cliente Redis.

   El paquete de instalación del cliente Windows Redis se puede descargar **aquí**

2. Abra la herramienta CLI **cmd.exe** y ejecute comandos para ir al directorio donde se guarda el paquete de instalación del cliente de Redis descomprimido.

   Por ejemplo, para ir al directorio **D:\redis-64.3.0.503**, ejecute los siguientes comandos:

   D:

   cd D:\redis-64.3.0.503

3. Ejecute el siguiente comando para acceder a la instancia de DCS Redis elegida:

   redis-cli -h 127.0.0.1 -p 8000 -a <*password*>

   ⚠ ATENCIÓN

   En el comando anterior:

   – La dirección siguiente **-h** indica la dirección del cliente Stunnel, que es **127.0.0.1**.

   – El puerto siguiente **-p** es el puerto de escucha del cliente Stunnel, que se ha configurado en el campo de **accept** en **Paso 5**. **8000** se usa un ejemplo aquí.

   No utilice la dirección de acceso público y el puerto mostrados en la consola para los parámetros **-h** y **-p**.

   <*contraseña*> indica la contraseña utilizada para iniciar sesión en la instancia de DCS Redis elegida. Esta contraseña se define durante la creación de una instancia de DCS Redis.

   Ha accedido correctamente a la instancia si se muestra el siguiente resultado del comando:

   ```
   127.0.0.1:8000>
   ```

   Ingrese **info** y se devolverá la información de la instancia de DCS. Si no se devuelve ninguna información o se interrumpe la conexión, haga clic con el botón derecho en el icono Stunnel de la barra de tareas y elija **Show Log Window** en el menú contextual para mostrar los registros de Stunnel para el análisis de causa.

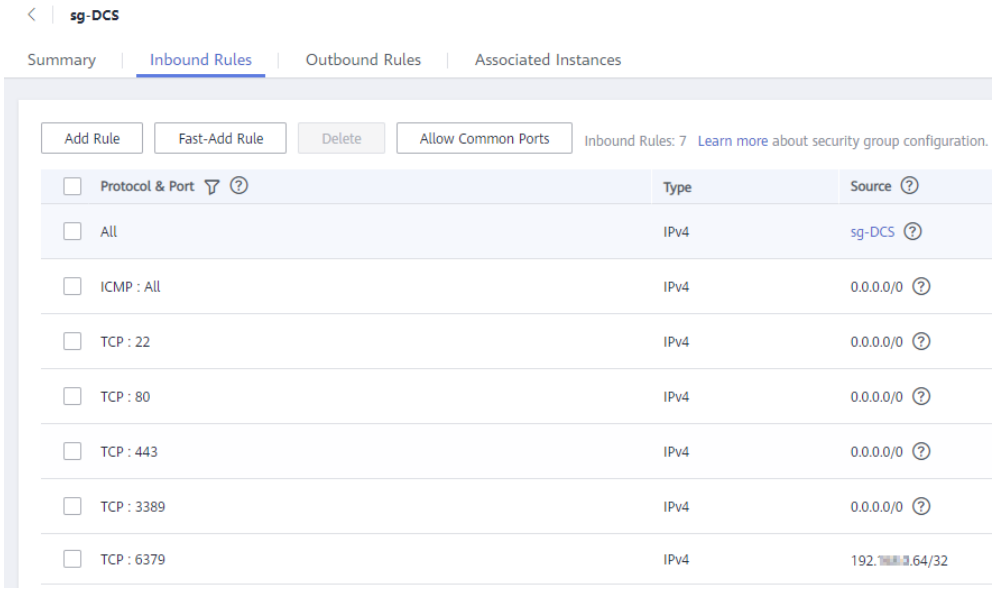
----Fin

Conexión a Redis sin la encriptación SSL

Paso 1 Asegúrese de que la regla de grupo de seguridad permita el acceso público a través del puerto 6379.

Cuando la encriptación SSL está deshabilitada, se puede acceder a la dirección de acceso público de instancia solo si se permite el acceso a través del puerto 6379.

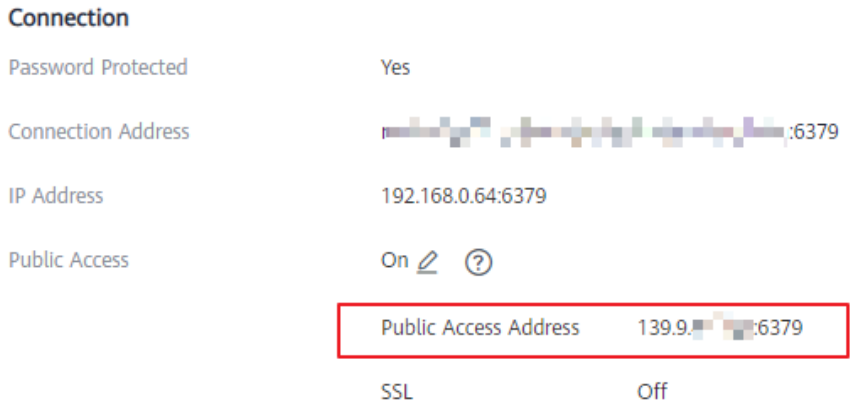
Figura 4-4 Regla de grupo de seguridad (puerto 6379)



Paso 2 Obtener la dirección de acceso público de la instancia.

La dirección de acceso público se muestra en la sección **Connection**.

Figura 4-5 Consulta de la dirección de acceso público (SSL deshabilitado; puerto 6379)



Paso 3 Obtenga y descomprima el paquete de instalación del cliente Redis.

El paquete de instalación del cliente Windows Redis se puede descargar **aquí**

Paso 4 Abra la herramienta CLI **cmd.exe** y ejecute comandos para ir al directorio donde se guarda el paquete de instalación del cliente de Redis descomprimido.

Por ejemplo, para ir al directorio **D:\redis-64.3.0.503**, ejecute los siguientes comandos:

D:

cd D:\redis-64.3.0.503

Paso 5 Ejecute el siguiente comando para acceder a la instancia de DCS Redis elegida:

redis-cli -h*{acceso a la red pública IP }* **-p 6379 -a** <*contraseña*>

En este comando, *{acceso a la red pública IP }* indica la dirección IP de la instancia de DCS Redis obtenida en **Paso 2**. *<contraseña>* indica la contraseña utilizada para iniciar sesión en la instancia de DCS Redis elegida. Esta contraseña se define durante la creación de una instancia de DCS Redis.

Ha accedido correctamente a la instancia si se muestra el siguiente resultado del comando:

```
139.**.**.175:6379>
```

Ingrese **info** y se devolverá la información de la instancia de DCS.

----Fin

Resolución de problemas

- **Síntoma**: se muestra "Error: Connection reset by peer" o se muestra un mensaje que indica que el host remoto cierra por la fuerza una conexión existente.

  Posible causa 1: El grupo de seguridad está configurado incorrectamente. Debe habilitar el puerto **36379** o **6379**.

  Posible causa 2: Se ha habilitado el encriptación SSL, pero Stunnel no está configurado durante la conexión. La dirección IP mostrada en la consola se utilizó para la conexión. En este caso, siga estrictamente las instrucciones proporcionadas en **Conexión a Redis con la encriptación SSL**.
- Para obtener más información acerca de los errores de conexión de Redis, consulte **Solución de problemas de conexión de Redis**.

4.2.4 Paso 3: Acceda a una instancia de DCS Redis en Linux

Esta sección describe cómo acceder a una instancia de DCS Redis 3.0 a través de una red pública mediante redis-cli en Linux.

El acceso público ayuda al personal de I+D a establecer un entorno local para el desarrollo o las pruebas, mejorando la eficiencia del desarrollo. Sin embargo, en el entorno de producción (entorno oficial), acceda a una instancia de DCS Redis a través de una VPC para garantizar un acceso eficiente.

Prerrequisitos:

Antes de utilizar redis-cli para acceder a una instancia de DCS Redis a través de una red pública, asegúrese de que:

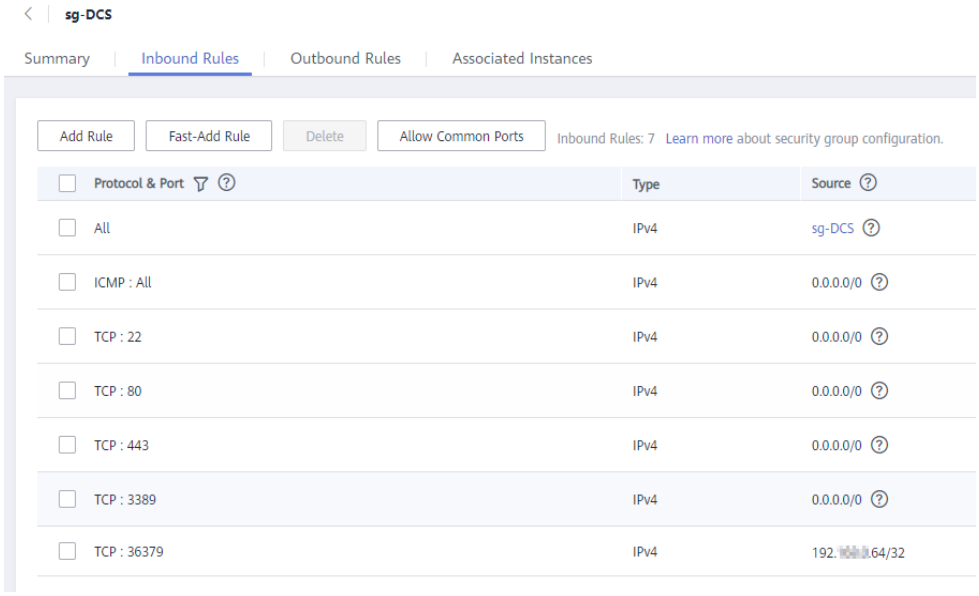
- La versión de instancia es Redis 3.0 y se ha habilitado el acceso público.
- Si se necesitan certificados para acceder a la instancia de DCS, descargue el certificado desde la página de detalles de la instancia de DCS. Para más detalles, consulte **Consulta de detalles de instancia**.

Conexión a Redis con la encriptación SSL

Paso 1 Asegúrese de que la regla de grupo de seguridad permita el acceso público a través del puerto 36379.

Cuando la encriptación SSL está habilitada, permita el acceso público a través del puerto 36379. Asegúrese de que el cliente Stunnel se ha instalado.

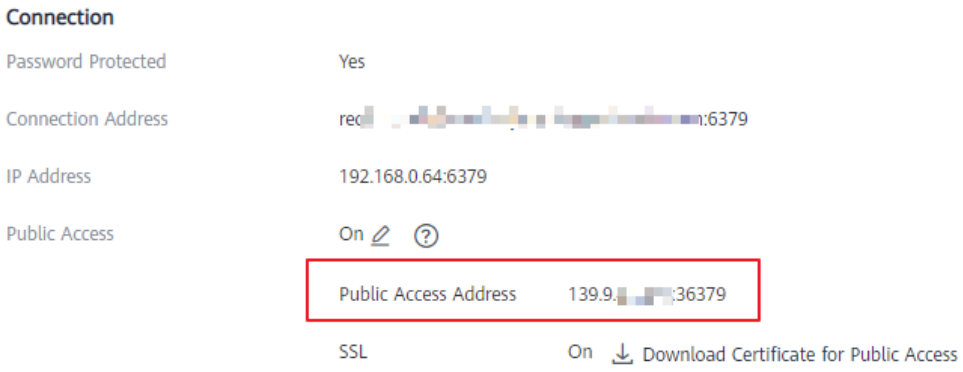
Figura 4-6 Regla de grupo de seguridad (puerto 36379)



Paso 2 Obtenga la dirección de acceso público y los certificados de la instancia en la página **Basic Information** de la instancia.

- La dirección de acceso público se muestra en la sección **Connection**.
- Los certificados se pueden descargar haciendo clic en **Download Certificate for Public Access** en la sección **Connection**. Después de la descompresión, obtendrá **dcs-ca.cer** (el certificado de clave pública en formato binario) y **dcs-ca-bundle.pem** (el archivo de certificado en formato de texto).

Figura 4-7 Consulta de la dirección de acceso público (SSL habilitado; puerto 36379)



Paso 3 Inicie sesión en el dispositivo local de Linux.

Paso 4 Instale el cliente Stunnel.

Utilice cualquiera de los siguientes métodos para instalar Stunnel.

NOTA

Se recomiendan los métodos de instalación **apt** y **yum**. Cualquier SO común de Linux debe soportar al menos uno de estos métodos de instalación.

- método **apt-get**:

  apt-get se utiliza para gestionar los paquetes de software DEB y es aplicable a SO de Debian como Ubuntu. Ejecute el siguiente comando para instalar Stunnel:

  apt install stunnel o **apt-get install stunnel**

  Si no puede encontrar Stunnel después de ejecutar el comando, ejecute el comando **apt update** para actualizar la configuración y luego vuelva a instalar Stunnel.

- método **yum**:

  yum se utiliza para administrar paquetes de software RPM y aplicable a SOs como Fedora, CentOS y Red Hat. Ejecute el siguiente comando para instalar Stunnel:

  yum install stunnel

Paso 5 Abra el archivo de configuración de Stunnel **stunnel.conf**.

- Si Stunnel se instala usando **apt-get**, el archivo de configuración se almacena en el directorio **/etc/stunnel/stunnel.conf** de forma predeterminada.

  Si este directorio no existe o no existe ningún archivo de configuración, agregue un directorio o archivo de configuración.

- Si Stunnel se instala usando **yum**, el archivo de configuración se almacena en el directorio **/usr/local/stunnel/stunnel.conf** por defecto.

  Si este directorio no existe o no existe ningún archivo de configuración, agregue un directorio o archivo de configuración.

  NOTA

  - Si no está seguro de dónde almacenar el archivo de configuración, introduzca el comando **stunnel** después de la instalación para ver el directorio para almacenar el archivo de configuración.
  - El archivo de configuración se puede almacenar en cualquier directorio. Especifique este archivo de configuración al iniciar Stunnel.

Paso 6 Agregue el siguiente contenido al archivo de configuración **stunnel.conf** y, a continuación, guarde y salga.

```
debug = 4
output = /var/log/stunnel.log
sslVersion = all
[redis-client]
client = yes
accept = 8000
connect = {public access address}
CAfile = /etc/stunnel/dcs-ca.cer
```

Modifique los siguientes parámetros según sea necesario y deje otros parámetros sin cambios:

- **client**: indica Stunnel. El valor fijo es **yes**.
- **CAfile**: especifica un certificado de CA, que es opcional. Si se requiere un certificado de CA, descargue y descomprima el certificado **dcs-ca.cer** como se indica en **Paso 2**. Si no es necesario, elimine este parámetro.
- **accept**: especifica el número de puerto de escucha definido por el usuario de Stunnel. Especifique este parámetro al acceder a una instancia de DCS mediante un cliente de Redis.
- **connect**: especifica la dirección de reenvío y el número de puerto de Stunnel. Establezca este parámetro en la dirección de acceso público de instancia obtenida en **Paso 2**.

El siguiente es un ejemplo de configuración:


```
[redis-client]
client = yes
CAfile = D:\tmp\dcs\dcs-ca.cer
accept = 8000
connect = 49.**.**.211:36379
```

Paso 7 Ejecute los siguientes comandos para iniciar Stunnel:

stunnel /*{customdir}*/stunnel.conf

En el comando anterior, ***{customdir}*** indica el directorio de almacenamiento personalizado para el archivo **stunnel.conf** descrito en **Paso 5**. El siguiente es un ejemplo de comando:

stunnel /etc/stunnel/stunnel.conf

NOTA

Para Ubuntu SO, ejecute el comando **/etc/init.d/stunnel4 start** para iniciar Stunnel. El nombre del servicio o proceso es **stunnel4** para la versión Stunnel 4.x.

Después de iniciar el cliente Stunnel, ejecute el comando **ps -ef|grep stunnel** para comprobar si el proceso se está ejecutando correctamente.

Paso 8 Ejecute el siguiente comando para comprobar si Stunnel está siendo escuchado:

netstat -plunt |grep *8000*|grep "LISTEN"

8000 indica el número de puerto de escucha definido por el usuario de Stunnel configurado en el campo de **accept** en **Paso 6**.

Si se muestra una línea que contiene el número de puerto 8000 en el resultado devuelto, Stunnel se está ejecutando correctamente. Cuando el cliente de Redis se conecta a la dirección **127.0.0.1:8000**, Stunnel reenviará las solicitudes a la instancia de DCS Redis.

Paso 9 Acceda a la instancia de DCS Redis.

1. Inicie sesión en el dispositivo local de Linux.
2. Ejecute el siguiente comando para descargar el paquete de código fuente de su cliente Redis desde **http://download.redis.io/releases/redis-5.0.8.tar.gz**:

   wget http://download.redis.io/releases/redis-5.0.8.tar.gz

   NOTA

   También puede instalar el cliente Redis ejecutando el siguiente comando yum o apt:
   - **yum install redis**
   - **apt install redis-server**
3. Ejecute el siguiente comando para descomprimir el paquete de código fuente de su cliente Redis:

   tar -xzf redis-5.0.8.tar.gz
4. Ejecute los siguientes comandos para ir al directorio Redis y compilar el código fuente de su cliente Redis:

   cd redis-5.0.8

   make
5. Ejecute los siguiente comandos para acceder a la instancia de DCS Redis elegida:

   cd src

   ./redis-cli -h 127.0.0.1 -p *8000*

ATENCIÓN

En el comando anterior:

- La dirección siguiente **-h** indica la dirección del cliente Stunnel, que es **127.0.0.1**.
- El puerto siguiente **-p** es el puerto de escucha del cliente Stunnel, que se ha configurado en el campo de **accept** en **Paso 6**. 8000 se usa un ejemplo.

No utilice la dirección de acceso público y el puerto mostrados en la consola para los parámetros **-h** y **-p**.

6. Ingrese la contraseña. Puede leer y escribir datos almacenados en caché solo después de verificar la contraseña.

   auth *{contraseña}*

   {contraseña} indica la contraseña utilizada para iniciar sesión en la instancia de DCS Redis elegida. Esta contraseña se define durante la creación de una instancia de DCS Redis.

   Ha accedido correctamente a la instancia si se muestra el siguiente resultado del comando:

   ```
   OK
   127.0.0.1:8000>
   ```

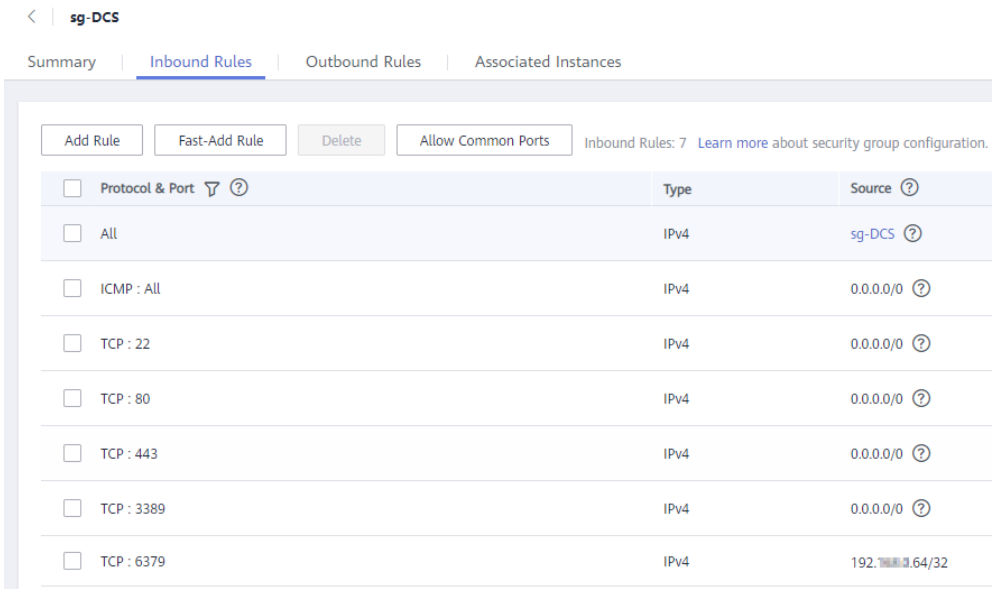
----Fin

Conexión a Redis sin la encriptación SSL

Paso 1 Asegúrese de que la regla de grupo de seguridad permita el acceso público a través del puerto 6379.

Cuando la encriptación SSL está deshabilitada, se puede acceder a la dirección de acceso público de instancia solo si se permite el acceso a través del puerto 6379.

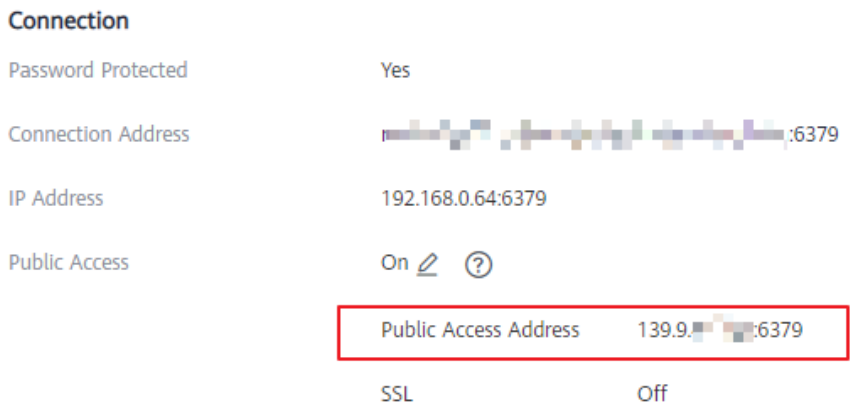
Figura 4-8 Regla de grupo de seguridad (puerto 6379)



Paso 2 Obtener la dirección de acceso público de la instancia.

La dirección de acceso público se muestra en la sección **Connection** de la página **Basic Information** de la instancia.

Figura 4-9 Consulta de la dirección de acceso público (SSL deshabilitado; puerto 6379)



Paso 3 Inicie sesión en el dispositivo local de Linux.

Paso 4 Ejecute el siguiente comando para descargar el paquete de código fuente de su cliente Redis desde http://download.redis.io/releases/redis-5.0.8.tar.gz:

wget http://download.redis.io/releases/redis-5.0.8.tar.gz

NOTA

También puede instalar el cliente Redis ejecutando el siguiente comando yum o apt:

- **yum install redis**
- **apt install redis-server**

Paso 5 Ejecute el siguiente comando para descomprimir el paquete de código fuente de su cliente Redis:

tar -xzf redis-5.0.8.tar.gz

Paso 6 Ejecute los siguientes comandos para ir al directorio Redis y compilar el código fuente de su cliente Redis:

cd redis-5.0.8

make

Paso 7 Ejecute los siguiente comandos para acceder a la instancia de DCS Redis elegida:

cd src

./redis-cli -h *{dirección de acceso público}* -p 6379

Reemplace {dirección de acceso público} con la dirección obtenida en **Paso 2**. Por ejemplo:

./redis-cli -h 49..**.211 -p 6379**

Paso 8 Ingrese la contraseña. Puede leer y escribir datos almacenados en caché solo después de verificar la contraseña.

auth *{contraseña}*

{contraseña} indica la contraseña utilizada para iniciar sesión en la instancia de DCS Redis elegida. Esta contraseña se define durante la creación de una instancia de DCS Redis.

Ha accedido correctamente a la instancia si se muestra el siguiente resultado del comando:

```
OK
49.**.**.211:6379>
```

----Fin

Resolución de problemas

- Síntoma: se muestra "Error: Connection reset by peer".

  Posible causa: El grupo de seguridad está configurado incorrectamente. Debe habilitar el puerto **36379** o **6379**.
- Cuando se utiliza redis-cli para conectarse a una instancia, se muestra el siguiente mensaje que indica que el host remoto cierra por la fuerza una conexión existente.

  Posible causa: Se ha habilitado el encriptación SSL, pero Stunnel no está configurado durante la conexión. La dirección IP mostrada en la consola se utilizó para la conexión. En este caso, siga estrictamente las instrucciones proporcionadas en **Conexión a Redis con la encriptación SSL**.
- Para obtener más información acerca de los errores de conexión de Redis, consulte **Solución de problemas de excepciones de conexión a Redis**.

4.3 Acceso en diferentes idiomas

4.3.1 redis-cli

Esta sección describe cómo utilizar redis-cli en un ECS en la misma VPC que una instancia de DCS Redis para conectarse a la instancia. Para obtener más información sobre más clientes, consulte el **sitio web oficial de Redis**.

Para obtener más información sobre cómo acceder a una instancia de DCS Redis a través de redes públicas, consulte **Paso 3: Acceda a una instancia de DCS Redis en Windows**.

NOTA

- Redis 3.0 no admite la personalización de puertos y solo permite el puerto 6379. Para Redis 4.0 y 5.0, puede especificar un puerto o utilizar el puerto predeterminado 6379. A continuación se utiliza el puerto predeterminado 6379. Si ha especificado un puerto, reemplace 6379 por el puerto real.
- Cuando se conecte a una instancia de Clúster Redis, asegúrese de que se agrega **-c** al comando. De lo contrario, la conexión fallará.
  - Ejecute el siguiente comando para conectarse a una instancia de Clúster Redis:

    ./redis-cli -h *{dcs_instancia_dirección}* **-p 6379 -a** *{contraseña}* **-c**
  - Ejecute el siguiente comando para conectarse a una instancia de nodo único, principal/en standby o de Clúster Proxy:

    ./redis-cli -h *{dcs_instancia_dirección}* **-p 6379 -a** *{contraseña}*

  Para más detalles, consulte **Paso 3**.

Prerrequisitos:

- Se ha creado una instancia de DCS Redis y se encuentra en el estado **Running**.

- Se ha creado un ECS. Para obtener más información sobre cómo crear un ECS, consulte **Compra de un ECS**.
- Si el ECS ejecuta el SO de Linux, asegúrese de que el entorno de compilación de GCC se ha instalado en el ECS.

Procedimiento (Linux)

Paso 1 Vea la dirección IP/nombre de dominio y el número de puerto de la instancia de DCS Redis a la que se debe acceder.

Para más detalles, consulte **Consulta de detalles de instancia**.

Paso 2 Para obtener más información sobre cómo instalar el cliente redis-cli, consulte las **instrucciones oficiales de Redis**.

Los siguientes pasos asumen que su cliente está instalado en el SO de Linux.

1. Inicie sesión en el ECS.
2. Ejecute el siguiente comando para descargar el paquete de código fuente de su cliente Redis desde **http://download.redis.io/releases/redis-5.0.8.tar.gz**:

   wget http://download.redis.io/releases/redis-5.0.8.tar.gz
3. Ejecute el siguiente comando para descomprimir el paquete de código fuente de su cliente Redis:

   tar -xzf redis-5.0.8.tar.gz
4. Ejecute los siguientes comandos para ir al directorio Redis y compilar el código fuente de su cliente Redis:

   cd redis-5.0.8

   make

   cd src

Paso 3 Acceda a la instancia de DCS Redis.

- Acceda a una instancia DCS de un tipo distinto de Clúster Redis.

  Realice el siguiente procedimiento para acceder a una instancia de DCS Redis 3.0 o a una instancia de DCS Redis 4.0 o 5.0 de nodo único, principal/en standby o de Clúster Proxy.

  a. Ejecute el siguiente comando para acceder a la instancia de DCS Redis elegida:

     ./redis-cli -h** {dcs_instancia_dirección} **-p 6379

     {dcs_instancia_dirección} indica la dirección IP/nombre de dominio de la instancia DCS y **6379** es el puerto utilizado para acceder a la instancia. La dirección IP/nombre de dominio y el número de puerto se obtienen en **Paso 1**.

     NOTA

     Para una instancia de Clúster Proxy DCS Redis, puede usar la **Connection Address** o la **IP Address** para {dcs_instancia_dirección}. Las direcciones se pueden obtener en la página de información básica de la instancia en la consola, como se muestra en **Figura 4-10**.

     - **Connection Address** y **IP Address** son las direcciones de LB. Las solicitudes se distribuyen entre nodos proxy.
     - Puede utilizar **Backend Addresses** para conectarse directamente al nodo proxy especificado de una instancia de DCS Redis 3.0 de Clúster Proxy.

Figura 4-10 Obtención de las direcciones para conectarse a instancias de Clúster Proxy DCS

En el ejemplo siguiente se utiliza la dirección de nombre de dominio de una instancia de DCS Redis. Cambie el nombre de dominio y el puerto según sea necesario.

```
[root@ecs-redis redis-5.0.8]# cd src
[root@ecs-redis src]# ./redis-cli -h redis-069949a-dcs-
lxy.dcs.huaweicloud.com -p 6379
redis-069949a-dcs-lxy.dcs.huaweicloud.com:6379>
```

b. Si ha establecido una contraseña para la instancia DCS, introduzca la contraseña en este paso. Puede leer y escribir datos almacenados en caché solo después de verificar la contraseña.

auth *{contraseña}*

{contraseña} indica la contraseña utilizada para iniciar sesión en la instancia de DCS Redis elegida. Esta contraseña se define durante la creación de una instancia de DCS Redis.

La salida del comando es la siguiente:

```
redis-069949a-dcs-lxy.dcs.huaweicloud.com:6379> auth *******
OK
redis-069949a-dcs-lxy.dcs.huaweicloud.com:6379>
```

- Acceda a una instancia DCS del tipo Clúster Redis.

  Realice el siguiente procedimiento para acceder a una instancia de DCS Redis 4.0 o 5.0 en tipo Clúster Redis.

  a. Ejecute el siguiente comando para acceder a la instancia de DCS Redis elegida:

  ./redis-cli -h *{dcs_instancia_dirección}* -p 6379 -a *{contraseña}* -c

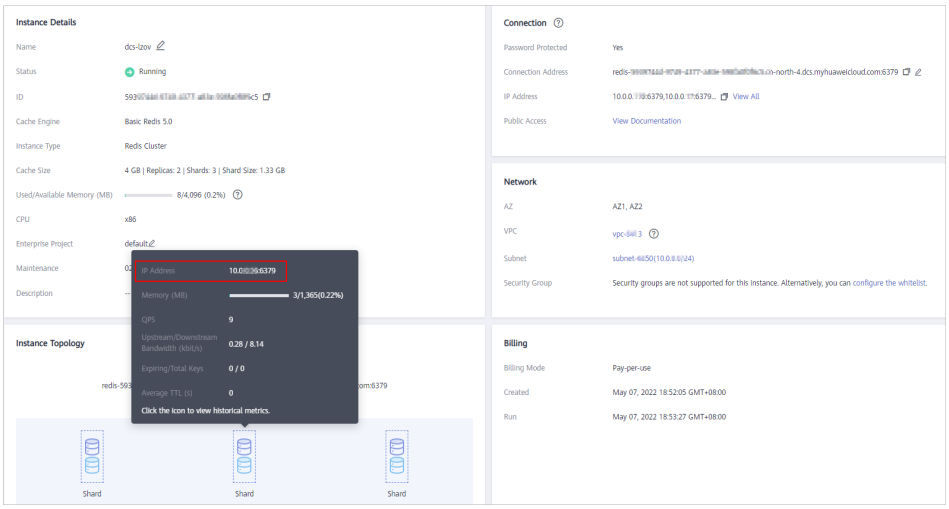
  {dcs_instancia_dirección} indica la dirección IP/nombre de dominio de la instancia de DCS Redis, **6379** es el puerto utilizado para acceder a la instancia, *{contraseña}* es la contraseña de la instancia, y -c se utiliza para acceder a los nodos de Clúster Redis. La dirección IP/nombre de dominio y el número de puerto se obtienen en **Paso 1**.

  NOTA

  Puede establecer {dcs_instancia_dirección} en **Connection Address** o **IP Address** en la sección **Connection** o **IP Address** en la sección **Instance Topology**. Las direcciones se pueden obtener en la página de información básica de la instancia en la consola, como se muestra en **Figura 4-11**.

  - El campo **IP Address** proporciona dos direcciones IP. Puede utilizar cualquiera de ellos para conectarse a la instancia. El algoritmo CRC16 (clave) mod 16384 se usa para calcular cuál es el intervalo hash de una clave dada.
  - Mediante el uso de **IP Address** en la sección **Instance Topology**, puede conectarse al partición especificado.

Figura 4-11 Obtención de las direcciones para conectarse a instancias de Clúster Redis DCS



- En el ejemplo siguiente se utiliza la dirección IP de una instancia de DCS Redis. Cambie la dirección IP y el puerto según sea necesario.

```
root@ecs-redis:~/redis-5.0.8/src# ./redis-cli -h 192.168.0.85 -p
6379 -a ****** -c
192.168.0.85:6379>
```

- En el ejemplo siguiente se utiliza el nombre de dominio de una instancia de DCS para Redis. Cambie el nombre de dominio y el puerto según sea necesario.

```
root@ecs-redis:~/redis-5.0.8/src# ./redis-cli -h redis-51e463c-dcs-
lxy.dcs.huaweicloud.com -p 6379 -a ****** -c
redis-51e463c-dcs-lxy.dcs.huaweicloud.com:6379>
```

b. Ejecute el siguiente comando para ver la información del nodo Clúster Redis:

cluster nodes

Cada partición de un Clúster Redis tiene un principal y una réplica por defecto. El comando de procedimiento proporciona toda la información de los nodos del clúster.

```
192.168.0.85:6379> cluster nodes
0988ae8fd3686074c9afdcce73d7878c81a33ddc 192.168.0.231:6379@16379 slave
f0141816260ca5029c56333095f015c7a058f113 0 1568084030
000 3 connected
1a32d809c0b743bd83b5e1c277d5d201d0140b75 192.168.0.85:6379@16379
myself,master - 0 1568084030000 2 connected 5461-10922
c8ad7af9a12cce3c8e416fb67bd6ec9207f0082d 192.168.0.130:6379@16379 slave
1a32d809c0b743bd83b5e1c277d5d201d0140b75 0 1568084031
000 2 connected
7ca218299c254b5da939f8e60a940ac8171adc27 192.168.0.22:6379@16379 master
- 0 1568084030000 1 connected 0-5460
f0141816260ca5029c56333095f015c7a058f113 192.168.0.170:6379@16379 master
- 0 1568084031992 3 connected 10923-16383
19b1a400815396c6223963b013ec934a657bdc52 192.168.0.161:6379@16379 slave
7ca218299c254b5da939f8e60a940ac8171adc27 0 1568084031
000 1 connected
```

Las operaciones de escritura solo se pueden realizar en nodos principals. El algoritmo CRC16 (clave) mod 16384 se usa para calcular cuál es el intervalo hash de una clave dada.

Como se muestra a continuación, el valor del **CRC16 (KEY) mode 16384** determina el intervalo hash en el que está situada una clave dada y redirige al cliente al nodo en el que está situada el intervalo hash.


```
192.168.0.170:6379> set hello world
-> Redirected to slot [866] located at 192.168.0.22:6379
OK
192.168.0.22:6379> set happy day
OK
192.168.0.22:6379> set abc 123
-> Redirected to slot [7638] located at 192.168.0.85:6379
OK
192.168.0.85:6379> get hello
-> Redirected to slot [866] located at 192.168.0.22:6379
"world"
192.168.0.22:6379> get abc
-> Redirected to slot [7638] located at 192.168.0.85:6379
"123"
192.168.0.85:6379>
```

----Fin

Procedimiento (Windows)

Descargue el paquete de instalación del cliente Windows Redis. Descomprima el paquete, abra la herramienta CLI **cmd.exe** y vaya al directorio donde se guarda el paquete de instalación del cliente de Redis descomprimido. A continuación, ejecute el siguiente comando para acceder a la instancia de DCS Redis:

redis-cli -h *XXX* -p *6379*

XXX indica la dirección IP/nombre de dominio de la instancia DCS y **6379** es un número de puerto de ejemplo usado para acceder a la instancia DCS. Para obtener más información sobre cómo obtener la dirección IP/nombre de dominio y el puerto, consulte **Consulta de detalles de instancia**. Cambie la dirección IP/nombre de dominio y el puerto según sea necesario.

4.3.2 Java

4.3.2.1 Jedis

Acceda a una instancia de DCS Redis a través de Jedis en un ECS en la misma VPC. Para obtener más información sobre cómo usar otros clientes de Redis, visite **el sitio web oficial de Redis**.

Prerrequisitos:

- Se ha creado una instancia de DCS Redis y se encuentra en el estado **Running**.
- Se ha creado un ECS. Para obtener más información sobre cómo crear un ECS, consulte **Compra de ECS**.
- Si el ECS ejecuta el SO de Linux, asegúrese de que el entorno de compilación de Java se ha instalado en el ECS.

Procedimiento

Paso 1 Vea la dirección IP/nombre de dominio y el número de puerto de la instancia de DCS Redis a la que se debe acceder.

Para obtener más información, consulte **Consulta de detalles de instancia**.

Paso 2 Inicie sesión en el ECS.

Paso 3 Utilice Maven para agregar la siguiente dependencia al archivo **pom.xml**:

```
<dependency>
    <groupId>redis.clients</groupId>
    <artifactId>jedis</artifactId>
    <version>4.1.1</version>
</dependency>
```

Paso 4 Acceda a la instancia de DCS mediante Jedis.

Obtener el **código fuente** del cliente Jedis. Utilice cualquiera de los dos métodos siguientes para acceder a una instancia de DCS Redis a través de Jedis:

- Conexión Jedis única
- Piscina de Jedis

Código de ejemplo:

1. Ejemplo de uso de Jedis para conectarse a una instancia de DCS Redis de nodo único, principal/en standby o de Clúster Proxy con una sola conexión

```
//Creating a connection in password mode
 String host = "192.168.0.150";
 int port = 6379;
 String pwd = "passwd";
 Jedis client = new Jedis(host, port);
 client.auth(pwd);
 client.connect();
//Run the SET command.
 String result = client.set("key-string", "Hello, Redis!");
 System.out.println( String.format("set command result:%s", result) );
//Run the GET command.
 String value = client.get("key-string");
System.out.println( String.format("get command result:%s", value) );

//Creating a connection in password-free mode
 String host = "192.168.0.150";
 int port = 6379;
 Jedis client = new Jedis(host, port);
 client.connect();
//Run the SET command.
 String result = client.set("key-string", "Hello, Redis!");
 System.out.println( String.format("set command result:%s", result) );
//Run the GET command.
 String value = client.get("key-string");
 System.out.println( String.format("get command result:%s", value) );
```

   host indica la dirección IP de ejemplo/nombre de dominio de la instancia DCS y port indica el número de puerto de la instancia DCS. Para obtener más información sobre cómo obtener la dirección IP/nombre de dominio y el puerto, consulte **Paso 1**. Cambie la dirección IP/dominio y el puerto según sea necesario. *pwd* indica la contraseña utilizada para iniciar sesión en la instancia de DCS Redis elegida. Esta contraseña se define durante la creación de una instancia de DCS Redis.

2. Ejemplo de uso de Jedis para conectarse a una instancia de DCS Redis de nodo único, principal/en standby o de Clúster Proxy con agrupación de conexiones

```
//Generate configuration information of a Jedis pool
 String ip = "192.168.0.150";
 int port = 6379;
 String pwd = "passwd";
 GenericObjectPoolConfig config = new GenericObjectPoolConfig();
 config.setTestOnBorrow(false);
 config.setTestOnReturn(false);
 config.setMaxTotal(100);
 config.setMaxIdle(100);
 config.setMaxWaitMillis(2000);
JedisPool pool = new JedisPool(config, ip, port, 100000, pwd);//Generate a
Jedis pool when the application is being initialized
```



```
//Get a Jedis connection from the Jedis pool when a service operation occurs
 Jedis client = pool.getResource();
 try {
     //Run commands
     String result = client.set("key-string", "Hello, Redis!");
     System.out.println( String.format("set command result:%s", result) );
     String value = client.get("key-string");
     System.out.println( String.format("get command result:%s", value) );
 } catch (Exception e) {
     // TODO: handle exception
 } finally {
     //Return the Jedis connection to the Jedis pool when the service
operation is completed
     if (null != client) {
         pool.returnResource(client);
     }
 } // end of try block
 //Destroy the Jedis pool when the application is closed
 pool.destroy();

//Configure the connection pool in password-free mode
 String ip = "192.168.0.150";
 int port = 6379;
 GenericObjectPoolConfig config = new GenericObjectPoolConfig();
 config.setTestOnBorrow(false);
 config.setTestOnReturn(false);
 config.setMaxTotal(100);
 config.setMaxIdle(100);
 config.setMaxWaitMillis(2000);
 JedisPool pool = new JedisPool(config, ip, port, 100000);//Generate a
JedisPool when the application is being initialized
 //Get a Jedis connection from the Jedis pool when a service operation occurs
 Jedis client = pool.getResource();
 try {
     //Run commands
     String result = client.set("key-string", "Hello, Redis!");
     System.out.println( String.format("set command result:%s", result) );
     String value = client.get("key-string");
     System.out.println( String.format("get command result:%s", value) );
 } catch (Exception e) {
     // TODO: handle exception
 } finally {
     //Return the Jedis connection to the Jedis pool when the service
operation is completed
     if (null != client) {
         pool.returnResource(client);
     }
 } // end of try block
 //Destroy the Jedis pool when the application is closed
 pool.destroy();
```

ip indica la dirección IP/nombre de dominio de la instancia DCS y port indica el número de puerto de la instancia DCS. Para obtener más información sobre cómo obtener la dirección IP/nombre de dominio y el puerto, consulte **Paso 1**. Cambie la dirección IP/dominio y el puerto según sea necesario. *pwd* indica la contraseña utilizada para iniciar sesión en la instancia de DCS Redis elegida. Esta contraseña se define durante la creación de una instancia de DCS Redis.

3. Código de ejemplo para conectarse al Clúster Redis mediante una conexión única
   - Con una contraseña

     ```
     //The following shows password-protected access.
     int port = 6379;
     String host = "192.168.144.37";
     //Create JedisCluster.
     Set<HostAndPort> nodes = new HashSet<HostAndPort>();
     nodes.add(new HostAndPort(host, port));
     JedisCluster cluster = new JedisCluster(nodes, 5000, 3000, 10,
     "password", new JedisPoolConfig());
     ```



```
cluster.set("key", "value");
System.out.println("Connected to RedisCluster:" + cluster.get("key"));
cluster.close();
```

– Sin contraseña

```
int port = 6379;
String host = "192.168.144.37";
//Create JedisCluster.
Set<HostAndPort> nodes = new HashSet<HostAndPort>();
nodes.add(new HostAndPort(host, port));
JedisCluster cluster = new JedisCluster(nodes);
cluster.set("key", "value");
System.out.println("Connected to RedisCluster:" + cluster.get("key"));
cluster.close();
```

host indica la dirección IP de ejemplo/nombre de dominio de la instancia DCS y port indica el número de puerto de la instancia DCS. Para obtener más información sobre cómo obtener la dirección IP/nombre de dominio y el puerto, consulte **Paso 1**. Cambie la dirección IP/dominio y el puerto según sea necesario. *contraseña* indica la contraseña utilizada para iniciar sesión en la instancia de DCS Redis elegida. Esta contraseña se define durante la creación de una instancia de DCS Redis.

Paso 5 Compilar código de acuerdo con el archivo **readme** en el código fuente del cliente Jedis. Ejecute el cliente Jedis para acceder a la instancia de DCS Redis elegida.

----Fin

4.3.2.2 Lettuce

Acceda a una instancia de DCS Redis a través de Lettuce en un ECS en la misma VPC. Para obtener más información sobre cómo usar otros clientes de Redis, visite **el sitio web oficial de Redis**.

Prerrequisitos:

- Se ha creado una instancia de DCS Redis y se encuentra en el estado **Running**.
- Se ha creado un ECS. Para obtener más información sobre cómo crear un ECS, consulte **Compra de ECS**.
- Si el ECS ejecuta el SO de Linux, asegúrese de que el entorno de compilación de Java se ha instalado en el ECS.

Procedimiento

Paso 1 Vea la dirección IP/nombre de dominio y el número de puerto de la instancia de DCS Redis a la que se debe acceder.

Para obtener más información, consulte **Consulta de detalles de instancia**.

Paso 2 Inicie sesión en el ECS.

Paso 3 Utilice Maven para agregar la siguiente dependencia al archivo **pom.xml**:

```
<dependency>
  <groupId>io.lettuce</groupId>
  <artifactId>lettuce-core</artifactId>
  <version>6.1.6.RELEASE</version>
</dependency>
```

Paso 4 Utilice Lettuce (un cliente de Java) para conectarse a la instancia DCS.

- Ejemplo de uso de Lettuce para conectarse a una instancia de DCS Redis de nodo único, principal/en standby o de Clúster Proxy con una sola conexión


```
// password indicates the connection password. If there is no password,
delete "password@". If there is a password and it contains special
characters, conversion is required.
RedisClient redisClient = RedisClient.create("redis://password@host:port");
StatefulRedisConnection<String, String> connection = redisClient.connect();
RedisCommands<String, String> syncCommands = connection.sync();
syncCommands.set("key", "value");
System.out.println("Connected to Redis:" + syncCommands.get("key"));
// Close the connection.
connection.close();
// Close the client.
redisClient.shutdown();
```

- Ejemplo de uso de Lettuce para conectarse a una instancia de DCS Redis de nodo único, principal/en standby o de Clúster Proxy con agrupación de conexiones

  a. Agregue la siguiente dependencia además de la anterior dependencia Maven:

  ```
  <dependency>
     <groupId>org.apache.commons</groupId>
     <artifactId>commons-pool2</artifactId>
      <version>2.11.1</version>
   </dependency>
  ```

  b. El código es el siguiente:

  ```
  // password indicates the connection password. If there is no password,
  delete "password@". If there is a password and it contains special
  characters, conversion is required.
  RedisClient clusterClient = RedisClient.create("redis://
  password@host:port");
  GenericObjectPoolConfig<StatefulRedisConnection<String, String>>
  genericObjectPoolConfig = new GenericObjectPoolConfig();
  // Connection pool parameters
  genericObjectPoolConfig.setMaxIdle(3);
  genericObjectPoolConfig.setMinIdle(2);
  genericObjectPoolConfig.setMaxTotal(3);
  genericObjectPoolConfig.setMaxWaitMillis(-1);
  GenericObjectPool<StatefulRedisConnection<String, String>> pool =
  ConnectionPoolSupport
          .createGenericObjectPool(() -> clusterClient.connect(),
  genericObjectPoolConfig);
  // Obtain a connection to perform operations.
  try (StatefulRedisConnection<String, String> con = pool.borrowObject()) {
      RedisCommands<String, String> sync = con.sync();
      sync.set("key", "value");
      System.out.println("Connected by pool:" + sync.get("key"));
  } catch (Exception e) {
      e.printStackTrace();
  }finally {
      // Close the resources.
      pool.close();
      clusterClient.shutdown();
  }
  ```

- Ejemplo de uso de Lettuce para conectarse a una instancia de Clúster Redis de DCS para Redis con una sola conexión (se debe habilitar la actualización automática de la topología)

  ```
  public class SingleConnectionToCluster {
      public static void main(String[] args) {
           // Enable automated topology refresh.
          ClusterTopologyRefreshOptions topologyRefreshOptions =
  ClusterTopologyRefreshOptions.builder()
           // Periodic refresh: every time milliseconds.
              .enablePeriodicRefresh(Duration.ofMillis(time))
           // Triggers of adaptive refresh: MOVED redirection, ASK redirection,
  reconnection, unknown node (since 5.1), and slot not in any of the current
  shards (since 5.2).
              .enableAllAdaptiveRefreshTriggers()
              .build();
          // password indicates the connection password. If there is no
  password, delete "password@". If there is a password and it contains special
  ```



```
characters, conversion is required.
        RedisClusterClient redisClient = RedisClusterClient.create("redis://
password@host:port");
        redisClient.setOptions(ClusterClientOptions.builder()
            .topologyRefreshOptions(topologyRefreshOptions)
            .build());
        StatefulRedisClusterConnection<String, String> connection =
redisClient.connect();
         // Preferentially read data from the replicas.
         connection.setReadFrom(ReadFrom.REPLICA_PREFERRED);
        RedisAdvancedClusterCommands<String, String> syncCommands =
connection.sync();
        syncCommands.set("key", "value");
        System.out.println("Connected to RedisCluster:" +
syncCommands.get("key"));
        // Close the connection.
        connection.close();
        // Close the client.
        redisClient.shutdown();
    }
}
```

- Ejemplo de código para conectar al Clúster Redis con pooling de conexiones

  a. Agregue la siguiente dependencia además de la anterior dependencia Maven:

  ```
  <dependency>
     <groupId>org.apache.commons</groupId>
     <artifactId>commons-pool2</artifactId>
      <version>2.11.1</version>
   </dependency>
  ```

  b. El código es el siguiente (se debe habilitar la actualización automática de la topología):

  ```
  public class PoolConnectionToCluster {
      public static void main(String[] args) {
           // Enable automated topology refresh.
          ClusterTopologyRefreshOptions topologyRefreshOptions =
  ClusterTopologyRefreshOptions.builder()
           // Periodic refresh every time milliseconds.
              .enablePeriodicRefresh(Duration.ofMillis(time))
           // Triggers of adaptive refresh: MOVED redirection, ASK
  redirection, reconnection, unknown node (since 5.1), and slot not in any
  of the current shards (since 5.2).
              .enableAllAdaptiveRefreshTriggers()
              .build();
          // password indicates the connection password. If there is no
  password, delete "password@". If there is a password and it contains
  special characters, conversion is required.
          RedisClusterClient redisClient =
  RedisClusterClient.create("redis://password@host:port");
          redisClient.setOptions(ClusterClientOptions.builder()
              .topologyRefreshOptions(topologyRefreshOptions)
              .build());
          GenericObjectPoolConfig<StatefulRedisClusterConnection<String,
  String>> genericObjectPoolConfig
              = new GenericObjectPoolConfig();
          // Connection pool parameters
          genericObjectPoolConfig.setMaxIdle(3);
          genericObjectPoolConfig.setMinIdle(2);
          genericObjectPoolConfig.setMaxTotal(3);

  genericObjectPoolConfig.setTimeBetweenEvictionRuns(Duration.ofMillis(2000
  ));
          genericObjectPoolConfig.setMaxWait(Duration.ofMillis(5000));
          GenericObjectPool<StatefulRedisClusterConnection<String,
  String>> pool = ConnectionPoolSupport
              .createGenericObjectPool(() -> redisClient.connect(),
  genericObjectPoolConfig);
          // Obtain a connection to perform operations.
          try (StatefulRedisClusterConnection<String, String> con =
  ```



```
pool.borrowObject()) {
            // Preferentially read data from the replicas.
             con.setReadFrom(ReadFrom.REPLICA_PREFERRED);
            RedisAdvancedClusterCommands<String, String> syncCommands =
con.sync();
            syncCommands.set("key", "value");
            System.out.println("Connected to RedisCluster:" +
syncCommands.get("key"));
        } catch (Exception e) {
            e.printStackTrace();
        } finally {
    // Close the resources.
            pool.close();
            redisClient.shutdown();
        }
    }
}
```

host es la dirección IP/nombre de dominio de la instancia DCS, port es el número de puerto de la instancia DCS y password es la contraseña de la instancia DCS. Especifique estos parámetros según sea necesario antes de ejecutar el código. Se recomienda la agrupación de conexiones. Ajuste parámetros como **timeout**, **MaxTotal** (número máximo de conexiones), **MinIdle** (número mínimo de conexiones inactivas), **MaxIdle** (número máximo de conexiones inactivas) y **MaxWait** (tiempo máximo de espera) según los requisitos de servicio.

----Fin

4.3.2.3 Redisson

Acceda a una instancia de DCS Redis a través de Redisson en un ECS en la misma VPC. Para obtener más información sobre cómo usar otros clientes de Redis, visite **el sitio web oficial de Redis**.

NOTA

- Si se estableció una contraseña durante la creación de la instancia de DCS Redis, configure la contraseña para conectarse a Redis mediante Redisson. No codifique la contraseña de texto sin formato.
- Para conectarse a una instancia de nodo único, principal/en standby o de Clúster Proxy, utilice el método **useSingleServer** del objeto **SingleServerConfig** de Redisson. Para conectarse a una instancia de Clúster Redis, utilice el método **useClusterServers** del objeto **ClusterServersConfig**.

Prerrequisitos:

- Se ha creado una instancia de DCS Redis y se encuentra en el estado **Running**.
- Se ha creado un ECS. Para obtener más información sobre cómo crear un ECS, consulte **Compra de ECS**.
- Si el ECS ejecuta el SO de Linux, asegúrese de que el entorno de compilación de Java se ha instalado en el ECS.

Procedimiento

Paso 1 Vea la dirección IP/nombre de dominio y el número de puerto de la instancia de DCS Redis a la que se debe acceder.

Para obtener más información, consulte **Consulta de detalles de instancia**.

Paso 2 Inicie sesión en el ECS.

Paso 3 Utilice Maven para agregar la siguiente dependencia al archivo **pom.xml**:

```
<dependency>
  <groupId>org.redisson</groupId>
  <artifactId>redisson</artifactId>
  <version>3.16.8</version>
</dependency>
```

Paso 4 Acceda a la instancia de DCS mediante Redisson (un cliente de Java).

- Ejemplo de uso de Redisson para conectarse a una instancia de DCS Redis de nodo único, principal/en standby o de Clúster Proxy con una sola conexión

```
Config config = new Config();
SingleServerConfig singleServerConfig = config.useSingleServer();
singleServerConfig.setAddress("redis://host:port");
// singleServerConfig.setPassword("9client!");
RedissonClient redisson = Redisson.create(config);
//Test concurrentMap. Data is synchronized to Redis when the put method is
used.
ConcurrentMap<String, Object> map = redisson.getMap("FirstMap");
map.put("wanger", "male");
map.put("zhangsan", "nan");
map.put("lisi", "female");
ConcurrentMap resultMap = redisson.getMap("FirstMap");
System.out.println("resultMap==" + resultMap.keySet());
//Test Set
Set mySet = redisson.getSet("MySet");
mySet.add("wanger");
mySet.add("lisi");
Set resultSet = redisson.getSet("MySet");
System.out.println("resultSet===" + resultSet.size());
//Test Queue
Queue myQueue = redisson.getQueue("FirstQueue");
myQueue.add("wanger");
myQueue.add("lili");
myQueue.add("zhangsan");
myQueue.peek();
myQueue.poll();
Queue resultQueue = redisson.getQueue("FirstQueue");
System.out.println("resultQueue===" + resultQueue);
//Close the connection.
redisson.shutdown();
```

- Ejemplo de uso de Redisson para conectarse a una instancia de DCS Redis de nodo único, principal/en standby o de Clúster Proxy con agrupación de conexiones

```
//1. Initialization
Config config = new Config();
SingleServerConfig singleServerConfig = config.useSingleServer();
singleServerConfig.setAddress("redis://host:6379");
//Set the maximum number of connections in the connection pool of the master
node to 500.
singleServerConfig.setConnectionPoolSize(500);
//The connections will be automatically closed and removed from the
connection pool. The time unit is millisecond.
singleServerConfig.setIdleConnectionTimeout(10000);
RedissonClient redisson = Redisson.create(config);
//Test concurrentMap. Data is synchronized to Redis when the put method is
used.
ConcurrentMap<String, Object> map = redisson.getMap("FirstMap");
map.put("wanger", "male");
map.put("zhangsan", "nan");
map.put("lisi", "female");
ConcurrentMap resultMap = redisson.getMap("FirstMap");
System.out.println("resultMap==" + resultMap.keySet());
//Test Set
Set mySet = redisson.getSet("MySet");
mySet.add("wanger");
mySet.add("lisi");
Set resultSet = redisson.getSet("MySet");
System.out.println("resultSet===" + resultSet.size());
```



```
//Test Queue
Queue myQueue = redisson.getQueue("FirstQueue");
myQueue.add("wanger");
myQueue.add("lili");
myQueue.add("zhangsan");
myQueue.peek();
myQueue.poll();
Queue resultQueue = redisson.getQueue("FirstQueue");
System.out.println("resultQueue===" + resultQueue);
//Close the connection.
redisson.shutdown();
```

- Ejemplo de uso de Redisson para conectarse a un Clúster Redis

```
Config config = new Config();
ClusterServersConfig clusterServersConfig = config.useClusterServers();
clusterServersConfig.addNodeAddress("redis://host:port");
//Set a password.
// clusterServersConfig.setPassword("");
RedissonClient redisson = Redisson.create(config);
ConcurrentMap<String, Object> map = redisson.getMap("FirstMap");
map.put("wanger", "male");
map.put("zhangsan", "nan");
map.put("lisi", "female");
ConcurrentMap resultMap = redisson.getMap("FirstMap");
System.out.println("resultMap==" + resultMap.keySet());
//2. Test Set
Set mySet = redisson.getSet("MySet");
mySet.add("wanger");
mySet.add("lisi");
Set resultSet = redisson.getSet("MySet");
System.out.println("resultSet===" + resultSet.size());
//3. Test Queue
Queue myQueue = redisson.getQueue("FirstQueue");
myQueue.add("wanger");
myQueue.add("lili");
myQueue.add("zhangsan");
myQueue.peek();
myQueue.poll();
Queue resultQueue = redisson.getQueue("FirstQueue");
System.out.println("resultQueue===" + resultQueue);
//Close the connection.
redisson.shutdown();
```

----Fin

4.3.3 Integración de Lettuce con Spring Boot

Prerrequisitos:

- Se ha creado una instancia de DCS Redis y se encuentra en el estado **Running**.
- Se ha creado un ECS. Para obtener más información sobre cómo crear un ECS, consulte **Compra de ECS**.
- Si el ECS ejecuta el SO de Linux, asegúrese de que el entorno de compilación de Java se ha instalado en el ECS.

Procedimiento

Paso 1 Vea la dirección IP/nombre de dominio y el número de puerto de la instancia de DCS Redis a la que se debe acceder.

Para más detalles, consulte **Consulta de detalles de instancia**.

Paso 2 Inicie sesión en el ECS.

Paso 3 Utilice Maven para agregar la siguiente dependencia al archivo **pom.xml**:

NOTA

- Desde Spring Boot 2.0, Lettuce se utiliza como el cliente predeterminado para las conexiones.
- Se utilizan Spring Boot 2.6.6 y Lettuce 6.1.8.

```
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-web</artifactId>
</dependency>
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-data-redis</artifactId>
</dependency>
```

Paso 4 Utilice Spring Boot integrado con Lettuce para conectarse a la instancia.

- Ejemplo de uso de Spring Boot y Lettuce para conectarse a una instancia DCS Redis de nodo único, principal/en standby o de Clúster Proxy con una sola conexión

  a. Agregue la configuración de Redis al archivo de configuración **application.properties**.

  ```
  spring.redis.host=host
  spring.redis.database=0
  spring.redis.password=pwd
  spring.redis.port=port
  ```

  b. Clase RedisConfiguration de configuración de Redis

  ```
  @Bean
  public RedisTemplate<String, Object>
  redisTemplate(LettuceConnectionFactory lettuceConnectionFactory) {
      RedisTemplate<String, Object> template = new RedisTemplate<>();
      template.setConnectionFactory(lettuceConnectionFactory);
        // Replace the default JdkSerializationRedisSerializer with
  Jackson2JsonRedisSerializer to serialize and deserialize the Redis value.
      Jackson2JsonRedisSerializer<Object> jackson2JsonRedisSerializer =
  new Jackson2JsonRedisSerializer<>(Object.class);
      ObjectMapper mapper = new ObjectMapper();
      mapper.setVisibility(PropertyAccessor.ALL,
  JsonAutoDetect.Visibility.ANY);
      mapper.activateDefaultTyping(LaissezFaireSubTypeValidator.instance,
          ObjectMapper.DefaultTyping.NON_FINAL,
  JsonTypeInfo.As.PROPERTY);
      jackson2JsonRedisSerializer.setObjectMapper(mapper);
      StringRedisSerializer stringRedisSerializer = new
  StringRedisSerializer();
        // String serialization of keys
      template.setKeySerializer(stringRedisSerializer);
        // String serialization of hash keys
      template.setHashKeySerializer(stringRedisSerializer);
        // Jackson serialization of values
      template.setValueSerializer(jackson2JsonRedisSerializer);
        // Jackson serialization of hash values
      template.setHashValueSerializer(jackson2JsonRedisSerializer);
      template.afterPropertiesSet();
      return template;
  }
  ```

  c. Clase RedisUtil de operación de Redis

  ```
   /**
    * Obtain data from the cache.
    * @param key
    * @return value
    */
   public Object get(String key){
       return key==null?null:redisTemplate.opsForValue().get(key);
   }
  ```



```
/**
 * Write data to the cache.
 * @param key
 * @param value
 * @return true (successful) false (failed)
 */
public boolean set(String key,Object value) {
    try {
        redisTemplate.opsForValue().set(key, value);
        return true;
    } catch (Exception e) {
        e.printStackTrace();
        return false;
    }
}
```

d. Escribe la clase de controlador para la prueba.

```
  @RestController
public class HelloRedis {
    @Autowired
    RedisUtil redisUtil;


    @RequestMapping("/setParams")
    @ResponseBody
    public String setParams(String name) {
            redisUtil.set("name", name);
            return "success";
        }
    @RequestMapping("/getParams")
    @ResponseBody
    public String getParams(String name) {
    System.out.println("--------------" + name + "-------------");
    String retName = redisUtil.get(name) + "";
    return retName;
    }
    }
```

- Ejemplo de uso de Spring Boot y Lettuce para conectarse a una instancia DCS Redis de nodo único, principal/en standby o de Clúster Proxy con agrupación de conexiones

  a. Agregue la siguiente dependencia además de la anterior dependencia Maven:

  ```
  <dependency>
    <groupId>org.apache.commons</groupId>
    <artifactId>commons-pool2</artifactId>
  </dependency>
  ```

  b. Agregue la configuración de Redis al archivo de configuración **application.properties**.

  ```
  spring.redis.host=host
  spring.redis.database=0
  spring.redis.password=pwd
  spring.redis.port=port
  # Connection timeout.
  spring.redis.timeout=1000
  # Maximum number of connections in the connection pool. A negative value
  indicates no limit.
  spring.redis.lettuce.pool.max-active=50
  # Minimum number of idle connections in the connection pool.
  spring.redis.lettuce.pool.min-idle=5
  # Maximum number of idle connections in the connection pool.
  spring.redis.lettuce.pool.max-idle=50
  # Maximum time for waiting for connections in the connection pool. A
  negative value indicates no limit.
  spring.redis.lettuce.pool.max-wait=5000
  # Interval for scheduling an eviction thread.
  spring.redis.pool.time-between-eviction-runs-millis=2000
  ```

  c. Clase RedisConfiguration de configuración de conexión Redis

  ```
  @Bean
  public RedisTemplate<String, Object>
  ```



```
redisTemplate(LettuceConnectionFactory lettuceConnectionFactory) {
    lettuceConnectionFactory.setShareNativeConnection(false);
    RedisTemplate<String, Object> template = new RedisTemplate<>();
    template.setConnectionFactory(lettuceConnectionFactory);
    // Use Jackson2JsonRedisSerializer to replace the default
JdkSerializationRedisSerializer to serialize and deserialize the Redis
value.
    Jackson2JsonRedisSerializer<Object> jackson2JsonRedisSerializer =
new Jackson2JsonRedisSerializer<>(Object.class);
    ObjectMapper mapper = new ObjectMapper();
    mapper.setVisibility(PropertyAccessor.ALL,
JsonAutoDetect.Visibility.ANY);
    mapper.activateDefaultTyping(LaissezFaireSubTypeValidator.instance,
        ObjectMapper.DefaultTyping.NON_FINAL,
JsonTypeInfo.As.PROPERTY);
    jackson2JsonRedisSerializer.setObjectMapper(mapper);
    StringRedisSerializer stringRedisSerializer = new
StringRedisSerializer();
    // String serialization of keys
    template.setKeySerializer(stringRedisSerializer);
    // String serialization of hash keys
    template.setHashKeySerializer(stringRedisSerializer);
    // Jackson serialization of values
    template.setValueSerializer(jackson2JsonRedisSerializer);
    // Jackson serialization of hash values
    template.setHashValueSerializer(jackson2JsonRedisSerializer);
    template.afterPropertiesSet();
    return template;
}
```

- Ejemplo de código para usar Spring Boot y Lettuce para conectarse a Clúster Redis mediante una sola conexión

  a. Agregue la configuración de Redis al archivo de configuración **application.properties**.

  ```
  spring.redis.cluster.nodes=host:port
  spring.redis.cluster.max-redirects=3
  spring.redis.password= pwd
  # Automated refresh interval
  spring.redis.lettuce.cluster.refresh.period=60
  # Enable automated refresh
  spring.redis.lettuce.cluster.refresh.adaptive=true
  spring.redis.timeout=60
  ```

  b. Clase RedisConfiguration de configuración de Redis (se debe habilitar la actualización automática de la topología).SpringBoot 2.3.0 and later support automated topology refresh.

  ```
  @Bean
  public LettuceConnectionFactory lettuceConnectionFactory() {
       String[] nodes = clusterNodes.split(",");
       List<RedisNode> listNodes = new ArrayList();
       for (String node : nodes) {
           String[] ipAndPort = node.split(":");
           RedisNode redisNode = new RedisNode(ipAndPort[0],
  Integer.parseInt(ipAndPort[1]));
           listNodes.add(redisNode);
       }
       RedisClusterConfiguration redisClusterConfiguration = new
  RedisClusterConfiguration();
       redisClusterConfiguration.setClusterNodes(listNodes);
       redisClusterConfiguration.setPassword(password);
       redisClusterConfiguration.setMaxRedirects(maxRedirects);
        // Configure automated topology refresh.
       ClusterTopologyRefreshOptions topologyRefreshOptions =
  ClusterTopologyRefreshOptions.builder()
           .enablePeriodicRefresh(Duration.ofSeconds(period)) // Refresh
  the topology periodically.
           .enableAllAdaptiveRefreshTriggers() // Refresh the topology
  based on events.
  ```



```
            .build();

        ClusterClientOptions clusterClientOptions =
ClusterClientOptions.builder()
            // Redis command execution timeout. Only when the command
execution times out will a reconnection be triggered using the new
topology.
            .timeoutOptions(TimeoutOptions.enabled(Duration.ofSeconds(period
)))
            .topologyRefreshOptions(topologyRefreshOptions)
            .build();
        LettuceClientConfiguration clientConfig =
LettucePoolingClientConfiguration.builder()
                .commandTimeout(Duration.ofSeconds(timeout))
                .readFrom(ReadFrom.REPLICA_PREFERRED) // Preferentially
read data from the replicas.
                .clientOptions(clusterClientOptions)
                .build();
        LettuceConnectionFactory factory = new
LettuceConnectionFactory(redisClusterConfiguration, clientConfig);
        return factory;
}

@Bean
public RedisTemplate<String, Object>
redisTemplate(LettuceConnectionFactory lettuceConnectionFactory) {
    RedisTemplate<String, Object> template = new RedisTemplate<>();
    template.setConnectionFactory(lettuceConnectionFactory);
     // Use Jackson2JsonRedisSerializer to replace the default
JdkSerializationRedisSerializer to serialize and deserialize the Redis
value.
    Jackson2JsonRedisSerializer<Object> jackson2JsonRedisSerializer =
new Jackson2JsonRedisSerializer<>(Object.class);
    ObjectMapper mapper = new ObjectMapper();
    mapper.setVisibility(PropertyAccessor.ALL,
JsonAutoDetect.Visibility.ANY);
    mapper.activateDefaultTyping(LaissezFaireSubTypeValidator.instance,
        ObjectMapper.DefaultTyping.NON_FINAL,
JsonTypeInfo.As.PROPERTY);
    jackson2JsonRedisSerializer.setObjectMapper(mapper);
    StringRedisSerializer stringRedisSerializer = new
StringRedisSerializer();
     // String serialization of keys
    template.setKeySerializer(stringRedisSerializer);
     // String serialization of hash keys
    template.setHashKeySerializer(stringRedisSerializer);
     // Jackson serialization of values
    template.setValueSerializer(jackson2JsonRedisSerializer);
     // Jackson serialization of hash values
    template.setHashValueSerializer(jackson2JsonRedisSerializer);
    template.afterPropertiesSet();
    return template;
}
```

- Ejemplo de código para usar Spring Boot y Lettuce para conectarse al Clúster Redis con conexión pooling

  a. Agregue la configuración de Redis al archivo de configuración **application.properties**.

  ```
  spring.redis.cluster.nodes=host:port
  spring.redis.cluster.max-redirects=3
  spring.redis.password=pwd
  spring.redis.lettuce.cluster.refresh.period=60
  spring.redis.lettuce.cluster.refresh.adaptive=true
  # Connection timeout.
  spring.redis.timeout=60s
  # Maximum number of connections in the connection pool. A negative value
  indicates no limit.
  spring.redis.lettuce.pool.max-active=50
  # Minimum number of idle connections in the connection pool.
  ```



```
spring.redis.lettuce.pool.min-idle=5
# Maximum number of idle connections in the connection pool.
spring.redis.lettuce.pool.max-idle=50
# Maximum time for waiting for connections in the connection pool. A
negative value indicates no limit.
spring.redis.lettuce.pool.max-wait=5000
# Interval for scheduling an eviction thread.
spring.redis.lettuce.pool.time-between-eviction-runs=2000
```

b. Clase RedisConfiguration de configuración de Redis (se debe habilitar la actualización automática de la topología).SpringBoot 2.3.0 and later support automated topology refresh.

```
@Bean
 public LettuceConnectionFactory lettuceConnectionFactory() {
     GenericObjectPoolConfig genericObjectPoolConfig = new
GenericObjectPoolConfig();
     genericObjectPoolConfig.setMaxIdle(maxIdle);
     genericObjectPoolConfig.setMinIdle(minIdle);
     genericObjectPoolConfig.setMaxTotal(maxActive);
     genericObjectPoolConfig.setMaxWait(Duration.ofMillis(maxWait));

genericObjectPoolConfig.setTimeBetweenEvictionRuns(Duration.ofMillis(time
BetweenEvictionRunsMillis));
     String[] nodes = clusterNodes.split(",");
     List<RedisNode> listNodes = new ArrayList();
     for (String node : nodes) {
         String[] ipAndPort = node.split(":");
         RedisNode redisNode = new RedisNode(ipAndPort[0],
Integer.parseInt(ipAndPort[1]));
         listNodes.add(redisNode);
     }
     RedisClusterConfiguration redisClusterConfiguration = new
RedisClusterConfiguration();
     redisClusterConfiguration.setClusterNodes(listNodes);
     redisClusterConfiguration.setPassword(password);
     redisClusterConfiguration.setMaxRedirects(maxRedirects);
      // Configure automated topology refresh.
     ClusterTopologyRefreshOptions topologyRefreshOptions =
ClusterTopologyRefreshOptions.builder()
         .enablePeriodicRefresh(Duration.ofSeconds(period)) // Refresh
the topology periodically.
         .enableAllAdaptiveRefreshTriggers() // Refresh the topology
based on events.
         .build();

     ClusterClientOptions clusterClientOptions =
ClusterClientOptions.builder()
         // Redis command execution timeout. Only when the command
execution times out will a reconnection be triggered using the new
topology.
         .timeoutOptions(TimeoutOptions.enabled(Duration.ofSeconds(period
)))
         .topologyRefreshOptions(topologyRefreshOptions)
         .build();
     LettuceClientConfiguration clientConfig =
LettucePoolingClientConfiguration.builder()
             .commandTimeout(Duration.ofSeconds(timeout))
             .poolConfig(genericObjectPoolConfig)
             .readFrom(ReadFrom.REPLICA_PREFERRED) // Preferentially
read data from the replicas.
             .clientOptions(clusterClientOptions)
             .build();
     LettuceConnectionFactory factory = new
LettuceConnectionFactory(redisClusterConfiguration, clientConfig);
     return factory;
 }

@Bean
public RedisTemplate<String, Object>
```



```
redisTemplate(LettuceConnectionFactory lettuceConnectionFactory) {
    lettuceConnectionFactory.setShareNativeConnection(false);
    RedisTemplate<String, Object> template = new RedisTemplate<>();
    template.setConnectionFactory(lettuceConnectionFactory);
     // Use Jackson2JsonRedisSerializer to replace the default
JdkSerializationRedisSerializer to serialize and deserialize the Redis
value.
    Jackson2JsonRedisSerializer<Object> jackson2JsonRedisSerializer =
new Jackson2JsonRedisSerializer<>(Object.class);
    ObjectMapper mapper = new ObjectMapper();
    mapper.setVisibility(PropertyAccessor.ALL,
JsonAutoDetect.Visibility.ANY);
    mapper.activateDefaultTyping(LaissezFaireSubTypeValidator.instance,
        ObjectMapper.DefaultTyping.NON_FINAL,
JsonTypeInfo.As.PROPERTY);
    jackson2JsonRedisSerializer.setObjectMapper(mapper);
    StringRedisSerializer stringRedisSerializer = new
StringRedisSerializer();
     // String serialization of keys
    template.setKeySerializer(stringRedisSerializer);
     // String serialization of hash keys
    template.setHashKeySerializer(stringRedisSerializer);
     // Jackson serialization of values
    template.setValueSerializer(jackson2JsonRedisSerializer);
     // Jackson serialization of hash values
    template.setHashValueSerializer(jackson2JsonRedisSerializer);
    template.afterPropertiesSet();
    return template;
}
```

host es la dirección IP/nombre de dominio de la instancia DCS, port es el número de puerto de la instancia DCS y pwd es la contraseña de la instancia DCS. Especifique estos parámetros según sea necesario antes de ejecutar el código. Se recomienda la agrupación de conexiones. Ajuste parámetros como **TimeOut**, **MaxTotal** (número máximo de conexiones), **MinIdle** (número mínimo de conexiones inactivas), **MaxIdle** (número máximo de conexiones inactivas) y **MaxWait** (tiempo máximo de espera) según los requisitos de servicio.

----Fin

4.3.4 Clientes en Python

Acceda a una instancia de DCS Redis a través de redis-py en un ECS en la misma VPC. Para obtener más información sobre cómo usar otros clientes de Redis, visite **el sitio web oficial de Redis**.

NOTA

Utilice redis-py para conectarse a instancias de nodo único, principal/en standby y de Clúster Proxy y redis-py-cluster para conectarse a instancias de Clúster Redis.

Prerrequisitos:

- Se ha creado una instancia de DCS Redis y se encuentra en el estado **Running**.
- Se ha creado un ECS. Para obtener más información sobre cómo crear un ECS, consulte **Compra de ECS**.
- Si el ECS ejecuta el SO de Linux, asegúrese de que el entorno de compilación de Python se ha instalado en el ECS.

Procedimiento

Paso 1 Vea la dirección IP/nombre de dominio y el número de puerto de la instancia de DCS Redis a la que se debe acceder.

Para obtener más información, consulte **Consulta de detalles de instancia**.

Paso 2 Inicie sesión en el ECS.

A continuación se utiliza CentOS como ejemplo para describir cómo acceder a una instancia mediante un cliente de Python.

Paso 3 Acceda a la instancia de DCS Redis.

Si el sistema no proporciona Python, ejecute el siguiente comando **yum** para instalarlo:

yum install python

NOTA

La versión de Python debe ser 3.6 o posterior. Si la versión predeterminada de Python es anterior a la 3.6, realice las siguientes operaciones para cambiarla:

1. Ejecute el comando rm -rf python para eliminar el enlace simbólico de Python.
2. Ejecute el comando **ln -s python***X.X.X* **python** para crear otro enlace de Python. En el comando, *X.X.X* indica el número de versión de Python.

- Si la instancia es una instancia de nodo único, principal/en standby o de Clúster Proxy:

  a. Instalar Python y redis-py.

     i. Si el sistema no proporciona Python, ejecute el siguiente comando **yum** para instalarlo.

     ii. Ejecute el siguiente comando para descargar y descomprimir el paquete redis-py:

     wget https://github.com/andymccurdy/redis-py/archive/master.zip

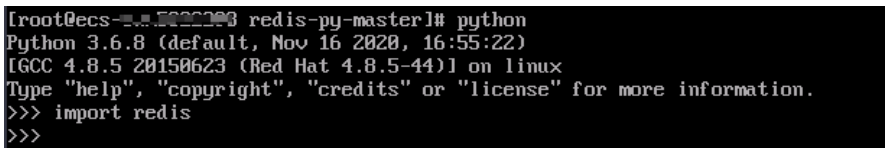
     unzip master.zip

     iii. Vaya al directorio donde se guarda el paquete redis-py descomprimido e instale redis-py.

     python setup.py install

     Después de la instalación, ejecute el comando **python**. redis-py se ha instalado correctamente si se muestra el siguiente resultado del comando:

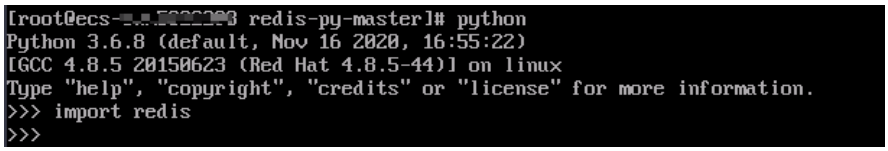
     Figura 4-12 Ejecutar el comando python

  b. Utilice el cliente redis-py para conectarse a la instancia. En los siguientes pasos, los comandos se ejecutan en modo CLI. (Alternativamente, escriba los comandos en un script de Python y luego ejecute el script.)

     i. Ejecute el comando **python** para entrar en el modo CLI. Ha entrado en el modo CLI si se muestra el siguiente resultado del comando:

Figura 4-13 Entrar en el modo CLI



ii. Ejecute el siguiente comando para acceder a la instancia de DCS Redis elegida:

```
r = redis.StrictRedis(host='XXX.XXX.XXX.XXX', port=6379,
password='******');
```

XXX.XXX.XXX.XXX indica la dirección IP/nombre de dominio de la instancia DCS y **6379** es un número de puerto de ejemplo de la instancia. Para obtener más información sobre cómo obtener la dirección IP/nombre de dominio y el puerto, consulte **Paso 1**. Cambie la dirección IP/nombre de dominio y el puerto según sea necesario. ****** indica la contraseña utilizada para iniciar sesión en la instancia de DCS Redis elegida. Esta contraseña se define durante la creación de una instancia de DCS Redis.

Ha accedido correctamente a la instancia si se muestra el siguiente resultado del comando. Introduzca comandos para realizar operaciones de lectura y escritura en la base de datos.

Figura 4-14 Redis conectado correctamente

- Si la instancia es una de Clúster Redis:
  a. Instale el cliente redis-py-cluster.
     i. Descargue la versión publicada.

        wget https://github.com/Grokzen/redis-py-cluster/releases/download/2.1.3/redis-py-cluster-2.1.3.tar.gz

     ii. Descomprima el paquete.

        tar -xvf redis-py-cluster-2.1.3.tar.gz

     iii. Vaya al directorio donde se guarda el paquete redis-py-cluster descomprimido e instale redis-py-cluster.

        python setup.py install

  b. Acceda a la instancia de DCS Redis mediante redis-py-cluster.

     En los siguientes pasos, los comandos se ejecutan en modo CLI. (Alternativamente, escriba los comandos en un script de Python y luego ejecute el script.)

     i. Ejecute el comando **python** para entrar en el modo CLI.

     ii. Ejecute el siguiente comando para acceder a la instancia de DCS Redis elegida:

```
>>> from rediscluster import RedisCluster

>>> startup_nodes = [{"host": "192.168.0.143", "port": "6379"}]

>>> rc = RedisCluster(startup_nodes=startup_nodes,
decode_responses=True)
```



```
>>> rc.set("foo", "bar")
True
>>> print(rc.get("foo"))
'bar'
```

----Fin

4.3.5 go-redis

Acceda a una instancia de DCS Redis a través de go-redis en un ECS en la misma VPC. Para obtener más información sobre cómo usar otros clientes de Redis, visite **el sitio web oficial de Redis**.

Prerrequisitos:

- Se ha creado una instancia de DCS Redis y se encuentra en el estado **Running**.
- Se ha creado un ECS. Para obtener más información sobre cómo crear un ECS, consulte **Compra de ECS**.

Procedimiento

Paso 1 Vea la dirección IP/nombre de dominio y el número de puerto de la instancia de DCS Redis a la que se debe acceder.

Para obtener más información, consulte **Consulta de detalles de instancia**.

Paso 2 Inicie sesión en el ECS.

Un ECS de Windows se utiliza como ejemplo.

Paso 3 Instale Visual Studio Community 2017 en ECS.

Paso 4 Inicie Visual Studio y cree un proyecto. El nombre del proyecto se puede personalizar. En este ejemplo, el nombre del proyecto se establece en redisdemo.

Paso 5 Importe el paquete de dependencias de go-redis e ingrese **go get github.com/go-redis/redis** en el terminal.

Paso 6 Escribe el siguiente código:

```
package main

import (
    "fmt"
    "github.com/go-redis/redis"
)

func main() {
    // Single-node
    rdb := redis.NewClient(&redis.Options{
        Addr:     "host:port",
        Password: "********", // no password set
        DB:       0,  // use default DB
    })

    val, err := rdb.Get("key").Result()
    if err != nil {
        if err == redis.Nil {
            fmt.Println("key does not exists")
            return
        }
        panic(err)
    }
```



```
    fmt.Println(val)

    //Cluster
    rdbCluster := redis.NewClusterClient(&redis.ClusterOptions{
        Addrs:    []string{"host:port"},
        Password: "********",
    })
    val1, err1 := rdbCluster.Get("key").Result()
    if err1 != nil {
        if err == redis.Nil {
            fmt.Println("key does not exists")
            return
        }
        panic(err)
    }
    fmt.Println(val1)
}
```

host:port son la dirección IP/nombre de dominio y el número de puerto de la instancia de DCS Redis. Para obtener más información sobre cómo obtener la dirección IP/nombre de dominio y el puerto, consulte **Paso 1**. Cambie la dirección IP/nombre de dominio y el puerto según sea necesario. ******** indica la contraseña utilizada para iniciar sesión en la instancia de DCS Redis. Esta contraseña se define durante la creación de una instancia de DCS Redis.

Paso 7 Ejecute el comando **go build -o test main.go** para empaquetar el código en un archivo ejecutable, por ejemplo, **test**.

ATENCIÓN

Para ejecutar el paquete en el SO de Linux, establezca los siguientes parámetros antes de empaquetar:

set GOARCH=amd64

set GOOS=linux

Paso 8 Ejecute el comando**./test** para acceder a la instancia de DCS.

----Fin

4.3.6 hiredis in C++

Acceda a una instancia de DCS Redis a través de hiredis en un ECS en la misma VPC. Para obtener más información sobre cómo usar otros clientes de Redis, visite **el sitio web oficial de Redis**.

NOTA

Las operaciones descritas en esta sección se aplican solo a instancias de nodo único, principal/en standby y de Clúster Proxy. Para usar C++ para conectarse a una instancia de Clúster Redis, consulte la **descripción del cliente Redis de C++**.

Prerrequisitos:

- Se ha creado una instancia de DCS Redis y se encuentra en el estado **Running**.
- Se ha creado un ECS. Para obtener más información sobre cómo crear un ECS, consulte **Compra de ECS**.
- Si el ECS ejecuta el SO de Linux, asegúrese de que el entorno de compilación de GCC se ha instalado en el ECS.

Procedimiento

Paso 1 Vea la dirección IP/nombre de dominio y el número de puerto de la instancia de DCS Redis a la que se debe acceder.

Para obtener más información, consulte **Consulta de detalles de instancia**.

Paso 2 Inicie sesión en el ECS.

A continuación se utiliza CentOS como ejemplo para describir cómo acceder a una instancia en C++.

Paso 3 Instalar GCC, Make, y hiredis.

Si el sistema no proporciona un entorno de compilación, ejecute el siguiente comando **yum** para instalar el entorno:

yum install gcc make

Paso 4 Ejecute el siguiente comando para descargar y descomprimir el paquete contratado:

wget https://github.com/redis/hiredis/archive/master.zip

unzip master.zip

Paso 5 Ir al directorio donde se guarda el paquete descomprimido contratados, y compilar e instalar contratados.

make

make install

Paso 6 Acceda a la instancia de DCS mediante hiredis.

A continuación se describe la autenticación de conexión y contraseña de contratados. Para obtener más información sobre el uso de contratados, visite el sitio web oficial de Redis.

1. Edite el código de ejemplo para conectarse a una instancia de DCS y, a continuación, guarde el código y salga.

   vim connRedis.c

   Ejemplo:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <hiredis.h>
int main(int argc, char **argv) {
     unsigned int j;
     redisContext *conn;
     redisReply *reply;
     if (argc < 3) {
             printf("Usage: example {instance_ip_address} 6379 {password}\n");
             exit(0);
     }
     const char *hostname = argv[1];
     const int port = atoi(argv[2]);
     const char *password = argv[3];
     struct timeval timeout = { 1, 500000 }; // 1.5 seconds
     conn = redisConnectWithTimeout(hostname, port, timeout);
     if (conn == NULL || conn->err) {
        if (conn) {
             printf("Connection error: %s\n", conn->errstr);
             redisFree(conn);
        } else {
```



```
            printf("Connection error: can't allocate redis context\n");
        }
    exit(1);
    }
    /* AUTH */
    reply = redisCommand(conn, "AUTH %s", password);
    printf("AUTH: %s\n", reply->str);
    freeReplyObject(reply);

    /* Set */
    reply = redisCommand(conn,"SET %s %s", "welcome", "Hello, DCS for
Redis!");
    printf("SET: %s\n", reply->str);
    freeReplyObject(reply);

    /* Get */
    reply = redisCommand(conn,"GET welcome");
    printf("GET welcome: %s\n", reply->str);
    freeReplyObject(reply);

    /* Disconnects and frees the context */
    redisFree(conn);
    return 0;
}
```

2. Ejecute el siguiente comando para compilar el código:

   gcc connRedis.c -o connRedis -I /usr/local/include/hiredis -lhiredis

   Si se informa de un error, localice el directorio donde se guarda el archivo **hiredis.h** y modifique el comando de compilación.

   Después de la compilación, se obtiene un archivo ejecutable connRedis.

3. Ejecute el siguiente comando para acceder a la instancia de DCS Redis elegida:

   ./connRedis *{redis_ip_address}* 6379 *{password}*

   {redis_instance_address} indica la dirección IP/nombre de dominio de la instancia DCS y **6379** es un número de puerto de ejemplo de la instancia DCS. Para obtener más información sobre cómo obtener la dirección IP/nombre de dominio y el puerto, consulte **Paso 1**. Cambie la dirección IP/nombre de dominio y el puerto según sea necesario. *{contraseña}* indica la contraseña utilizada para iniciar sesión en la instancia de DCS elegida para Redis. Esta contraseña se define durante la creación de una instancia de DCS Redis.

   Ha accedido correctamente a la instancia si se muestra el siguiente resultado del comando:

```
AUTH: OK
SET: OK
GET welcome: Hello, DCS for Redis!
```


AVISO

Si se informa de un error, indicando que no se pueden encontrar los archivos de biblioteca contratados, ejecute los siguientes comandos para copiar archivos relacionados a los directorios del sistema y agregar vínculos dinámicos:

mkdir /usr/lib/hiredis

cp /usr/local/lib/libhiredis.so.0.13 /usr/lib/hiredis/

mkdir /usr/include/hiredis

cp /usr/local/include/hiredis/hiredis.h /usr/include/hiredis/

echo '/usr/local/lib' >>;>>;/etc/ld.so.conf

ldconfig

Reemplace las ubicaciones de los archivos **so** and **.h** por las actuales antes de ejecutar los comandos.

----Fin

4.3.7 C#

Acceda a una instancia de DCS Redis a través del cliente de C# StackExchange.Redis en un ECS en la misma VPC. Para obtener más información sobre cómo usar otros clientes de Redis, visite **el sitio web oficial de Redis**.

Prerrequisitos:

- Se ha creado una instancia de DCS Redis y se encuentra en el estado **Running**.
- Se ha creado un ECS. Para obtener más información sobre cómo crear un ECS, consulte **Compra de ECS**.
- Si el ECS ejecuta el SO de Linux, asegúrese de que el entorno de compilación de GCC se ha instalado en el ECS.

Procedimiento

Paso 1 Vea la dirección IP/nombre de dominio y el número de puerto de la instancia de DCS Redis a la que se debe acceder.

Para obtener más información, consulte **Consulta de detalles de instancia**.

Paso 2 Inicie sesión en el ECS.

Un ECS de Windows se utiliza como ejemplo.

Paso 3 Instale Visual Studio Community 2017 en ECS.

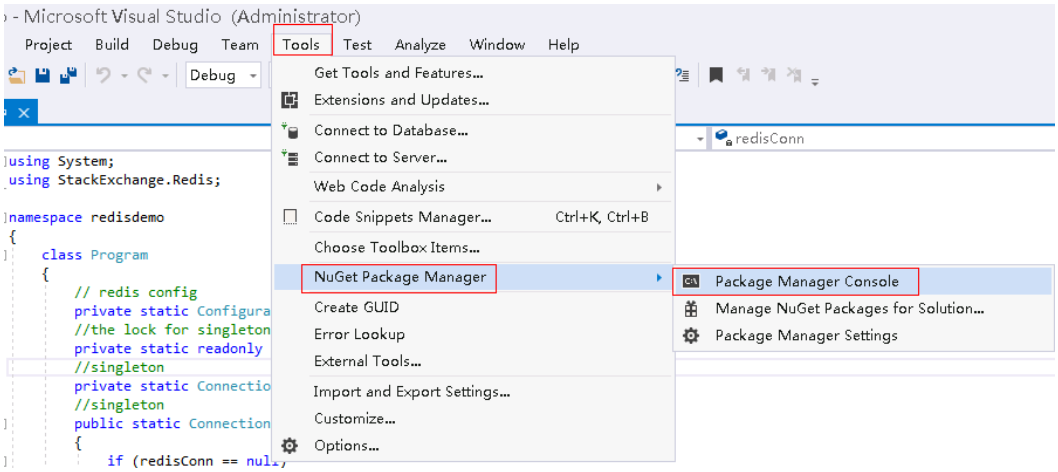
Paso 4 Inicie Visual Studio 2017 y cree un proyecto.

Establezca el nombre del proyecto en **redisdemo**.

Paso 5 Instale StackExchange.Redis con el administrador de paquetes NuGet de Visual Studio.

Acceda a la consola del administrador de paquetes NuGet de acuerdo con **Figura 4-15**, y escriba **Install-Package StackExchange.Redis -*Version 2.2.79***. (El número de versión es opcional).

Figura 4-15 Acceso a la consola del administrador de paquetes NuGet



Paso 6 Escriba el siguiente código y utilice los métodos String Set y Get para probar la conexión.

```
using System;
using StackExchange.Redis;

namespace redisdemo
{
    class Program
    {
        // redis config
        private static ConfigurationOptions connDCS =
ConfigurationOptions.Parse("10.10.38.233:6379,password=********,connectTimeout=200
0");
        //the lock for singleton
        private static readonly object Locker = new object();
        //singleton
        private static ConnectionMultiplexer redisConn;
        //singleton
        public static ConnectionMultiplexer getRedisConn()
        {
            if (redisConn == null)
            {
                lock (Locker)
                {
                    if (redisConn == null || !redisConn.IsConnected)
                    {
                        redisConn = ConnectionMultiplexer.Connect(connDCS);
                    }
                }
            }
            return redisConn;
        }
        static void Main(string[] args)
        {
            redisConn = getRedisConn();
            var db = redisConn.GetDatabase();
            //set get
            string strKey = "Hello";
            string strValue = "DCS for Redis!";
            Console.WriteLine( strKey + ", " + db.StringGet(strKey));

            Console.ReadLine();
        }
    }
}
```

10.10.38.233:6379 contiene un ejemplo de dirección IP/nombre de dominio y número de puerto de la instancia de DCS Redis. Para obtener más información sobre cómo obtener la dirección IP/nombre de dominio y el puerto, consulte **Paso 1**. Cambie la dirección IP/nombre

de dominio y el puerto según sea necesario. ******** indica la contraseña utilizada para iniciar sesión en la instancia de DCS Redis elegida. Esta contraseña se define durante la creación de una instancia de DCS Redis.

Paso 7 Ejecute el código. Ha accedido correctamente a la instancia si se muestra el siguiente resultado del comando:

```
Hello, DCS for Redis!
```

Para obtener más información acerca de otros comandos de StackExchange Redis, visite **StackExchange.Redis**.

----Fin

4.3.8 PHP

4.3.8.1 phpredis

Acceda a una instancia de DCS Redis a través de phpredis en un ECS en la misma VPC. Para obtener más información sobre cómo usar otros clientes de Redis, visite **el sitio web oficial de Redis**.

NOTA

Las operaciones descritas en esta sección se aplican solo a instancias de nodo único, principal/en standby y de Clúster Proxy. Para usar phpredis para conectarse a una instancia de Clúster Redis, consulte la **descripción de phpredis**.

Prerrequisitos:

- Se ha creado una instancia de DCS Redis y se encuentra en el estado **Running**.
- Se ha creado un ECS. Para obtener más información sobre cómo crear un ECS, consulte **Compra de ECS**.
- Si el ECS ejecuta el SO de Linux, asegúrese de que el entorno de compilación de GCC se ha instalado en el ECS.

Procedimiento

Paso 1 Vea la dirección IP/nombre de dominio y el número de puerto de la instancia de DCS Redis a la que se debe acceder.

Para obtener más información, consulte **Consulta de detalles de instancia**.

Paso 2 Inicie sesión en el ECS.

A continuación se utiliza CentOS como ejemplo para describir cómo acceder a una instancia a través de phpredis.

Paso 3 Instale los componentes de compilación GCC-C++ y Make.

yum install gcc-c++ make

Paso 4 Instale el paquete de desarrollo PHP y la herramienta CLI.

Ejecute el siguiente comando **yum** para instalar el paquete de desarrollo PHP:

yum install php-devel php-common php-cli

Una vez completada la instalación, ejecute el siguiente comando para consultar la versión de PHP y comprobar si la instalación se realiza correctamente:

php --version

Paso 5 Instale el cliente phpredis.

1. Descargue el paquete phpredis fuente.

   wget http://pecl.php.net/get/redis-5.3.7.tgz

   Esta versión se utiliza como ejemplo. Para descargar clientes phpredis de otras versiones, visite el sitio web oficial de Redis o PHP.

2. Descomprima el paquete phpredis fuente.

   tar -zxvf redis-5.3.7.tgz

   cd redis-5.3.7

3. Comando antes de la compilación.

   phpize

4. Configura el archivo **php-config**.

   ./configure --with-php-config=/usr/bin/php-config

   La ubicación del archivo varía dependiendo del modo de instalación SO y PHP. Se recomienda localizar el directorio donde se guarda el archivo antes de la configuración.

   find / -name php-config

5. Compilar e instalar el cliente phpredis.

   make && make install

6. Después de la instalación, agregue la configuración de **extensión** en el archivo **php.ini** para hacer referencia al módulo Redis.

   vim /etc/php.ini

   Agrega la siguiente configuración:

   ```
   extension = "/usr/lib64/php/modules/redis.so"
   ```

   NOTA

   El archivo redis.so puede guardarse en un directorio diferente de php.ini. Ejecute el siguiente comando para localizar el directorio:

   find / -name php.ini

7. Guarde la configuración y salga. A continuación, ejecute el siguiente comando para comprobar si la extensión tiene efecto:

   php -m |grep redis

   Si la salida del comando contiene **redis**, se ha configurado el entorno del cliente phpredis.

Paso 6 Acceda a la instancia de DCS mediante phpredis.

1. Edite un archivo redis.php.

   ```
   <?php
       $redis_host = "{redis_instance_address}";
       $redis_port = 6379;
       $user_pwd = "{password}";
       $redis = new Redis();
       if ($redis->connect($redis_host, $redis_port) == false) {
          die($redis->getLastError());
       }
       if ($redis->auth($user_pwd) == false) {
   ```



```
        die($redis->getLastError());
    }
    if ($redis->set("welcome", "Hello, DCS for Redis!") == false) {
        die($redis->getLastError());
    }
    $value = $redis->get("welcome");
    echo $value;
    $redis->close();
?>
```

{redis_instance_address} indica la dirección IP/nombre de dominio de la instancia DCS y 6379 es un número de puerto de ejemplo de la instancia DCS. Para obtener más información sobre cómo obtener la dirección IP/nombre de dominio y el puerto, consulte **Paso 1**. Cambie la dirección IP/nombre de dominio y el puerto según sea necesario. *{contraseña}* indica la contraseña utilizada para iniciar sesión en la instancia de DCS elegida para Redis. Esta contraseña se define durante la creación de una instancia de DCS Redis. Si el acceso sin contraseña está habilitado, proteja la declaración **if** para la autenticación con contraseña.

2. Ejecute el comando **php redis.php** para acceder a la instancia DCS.

----Fin

4.3.8.2 Predis

Acceda a una instancia de DCS Redis a través de Predis en un ECS en la misma VPC. Para obtener más información sobre cómo usar otros clientes de Redis, visite **el sitio web oficial de Redis**.

Prerrequisitos:

- Se ha creado una instancia de DCS Redis y se encuentra en el estado **Running**.
- Se ha creado un ECS. Para obtener más información sobre cómo crear un ECS, consulte **Compra de ECS**.
- Si el ECS ejecuta el SO de Linux, asegúrese de que el entorno de compilación de PHP se ha instalado en el ECS.

Procedimiento

Paso 1 Vea la dirección IP/nombre de dominio y el número de puerto de la instancia de DCS Redis a la que se debe acceder.

Para obtener más información, consulte **Consulta de detalles de instancia**.

Paso 2 Inicie sesión en el ECS.

Paso 3 Instale el paquete de desarrollo PHP y la herramienta CLI. Ejecute el siguiente comando **yum**:

yum install php-devel php-common php-cli

Paso 4 Una vez completada la instalación, compruebe el número de versión para asegurarse de que la instalación se realiza correctamente.

php --version

Paso 5 Descargue el paquete Predis en el directorio /usr/share/php.

1. Ejecute el siguiente comando para descargar el archivo fuente de Predis:

   wget https://github.com/predis/predis/archive/refs/tags/v1.1.10.tar.gz

NOTA

Esta versión se utiliza como ejemplo. Para descargar clientes Predis de otras versiones, visite el sitio web oficial de Redis o PHP.

2. Ejecute los siguientes comandos para descomprimir el paquete fuente Predis:

   tar -zxvf predis-1.1.10.tar.gz

3. Cambie el nombre del directorio Predis descomprimido por **predis** y muévalo a /usr/share/php/.

   mv predis-1.1.10 predis

Paso 6 Edite un archivo utilizado para conectarse a Redis.

- Ejemplo de uso de **redis.php** para conectarse a una instancia de DCS Redis de nodo único, principal/en standby o de Clúster Proxy:

```
<?php
    require 'predis/autoload.php';
    Predis\Autoloader::register();
    $client = new Predis\Client([
      'scheme' => 'tcp' ,
      'host'     => '{redis_instance_address}' ,
      'port'     => {port} ,
      'password' => '{password}'
    ]);
    $client->set('foo', 'bar');
    $value = $client->get('foo');
    echo $value;
?>
```

- Ejemplo de código para usar **redis-cluster.php** para conectarse al Clúster Redis:

```
<?php
   require 'predis/autoload.php';
        $servers = array(
         'tcp://{redis_instance_address}:{port}'
        );
       $options = array('cluster' => 'redis');
       $client = new Predis\Client($servers, $options);
       $client->set('foo', 'bar');
       $value = $client->get('foo');
       echo $value;
?>
```

{redis_instance_address} indica la dirección IP o el nombre de dominio real de la instancia DCS y {port} es el número de puerto real de la instancia DCS. Para obtener más información sobre cómo obtener la dirección IP/nombre de dominio y el puerto, consulte **Paso 1**. Cambie la dirección IP/nombre de dominio y el puerto según sea necesario. *{contraseña}* indica la contraseña utilizada para iniciar sesión en la instancia de DCS elegida para Redis. Esta contraseña se define durante la creación de una instancia de DCS Redis. Si se requiere acceso sin contraseña, elimine la línea que contiene "contraseña".

Paso 7 Ejecute el comando **php redis.php** para acceder a la instancia DCS.

----Fin

4.3.9 Node.js

Acceda a una instancia de DCS Redis a través de Node.js en un ECS en la misma VPC. Para obtener más información sobre cómo usar otros clientes de Redis, visite **el sitio web oficial de Redis**.

NOTA

Las operaciones descritas en esta sección se aplican solo a instancias de nodo único, principal/en standby y de Clúster Proxy. Para usar Node.js para conectarse a una instancia de Clúster Redis, consulte la **descripción del cliente de Node.js Redis**.

Prerrequisitos:

- Se ha creado una instancia de DCS Redis y se encuentra en el estado **Running**.
- Se ha creado un ECS. Para obtener más información sobre cómo crear un ECS, consulte **Compra de ECS**.
- Si el ECS ejecuta el SO de Linux, asegúrese de que el entorno de compilación de GCC se ha instalado en el ECS.

Procedimiento

- Para servidores cliente que ejecutan Ubuntu (serie Debian):

Paso 1 Vea la dirección IP/nombre de dominio y el número de puerto de la instancia de DCS Redis a la que se debe acceder.

Para obtener más información, consulte **Consulta de detalles de instancia**.

Paso 2 Inicie sesión en el ECS.

Paso 3 Instale Node.js.

apt install nodejs-legacy

Si el comando anterior no funciona, ejecute los siguientes comandos:

wget https://nodejs.org/dist/v0.12.4/node-v0.12.4.tar.gz --no-check-certificate

tar -xvf node-v4.28.5.tar.gz

cd node-v4.28.5

./configure

make

make install

NOTA

Una vez completada la instalación, ejecute el comando **node --version** para consultar la versión de Node.js para comprobar si la instalación se realiza correctamente.

Paso 4 Instale el administrador de paquetes de nodo (npm).

apt install npm

Paso 5 Instale el cliente de Redis ioredis.

npm install ioredis

Paso 6 Edite la secuencia de comandos de ejemplo para conectarse a una instancia de DCS.

Agregue el siguiente contenido al script **ioredisdemo.js**, incluyendo información sobre la conexión y la lectura de datos.


```
var Redis = require('ioredis');
var redis = new Redis({
  port: 6379,          // Redis port
  host: '192.168.0.196',   // Redis host
  family: 4,           // 4 (IPv4) or 6 (IPv6)
  password: '******',
  db: 0
});
redis.set('foo', 'bar');
redis.get('foo', function (err, result) {
  console.log(result);
});
// Or using a promise if the last argument isn't a function
redis.get('foo').then(function (result) {
  console.log(result);
});
// Arguments to commands are flattened, so the following are the same:
redis.sadd('set', 1, 3, 5, 7);
redis.sadd('set', [1, 3, 5, 7]);
// All arguments are passed directly to the redis server:
redis.set('key', 100, 'EX', 10);
```

host indica la dirección IP de ejemplo/nombre de dominio de la instancia DCS y el *port* indica el número de puerto de la instancia DCS. Para obtener más información sobre cómo obtener la dirección IP/nombre de dominio y el puerto, consulte **Paso 1**. Cambie la dirección IP/nombre de dominio y el puerto según sea necesario. ****** indica la contraseña utilizada para iniciar sesión en la instancia de DCS Redis elegida. Esta contraseña se define durante la creación de una instancia de DCS Redis.

Paso 7 Ejecute el script de ejemplo para acceder a la instancia de DCS elegida.

node ioredisdemo.js

----Fin

- Para servidores cliente que ejecutan CentOS (serie Red Hat):

Paso 1 Vea la dirección IP/nombre de dominio y el número de puerto de la instancia de DCS Redis a la que se debe acceder.

Para obtener más información, consulte **Consulta de detalles de instancia**.

Paso 2 Inicie sesión en el ECS.

Paso 3 Instale Node.js.

yum install nodejs

Si el comando anterior no funciona, ejecute los siguientes comandos:

wget https://nodejs.org/dist/v0.12.4/node-v0.12.4.tar.gz --no-check-certificate

tar -xvf node-v0.12.4.tar.gz

cd node-v0.12.4

./configure

make

make install

> **NOTA**
>
> Una vez completada la instalación, ejecute el comando **node --version** para consultar la versión de Node.js para comprobar si la instalación se realiza correctamente.

Paso 4 Instalar npm.

yum install npm

Paso 5 Instale el cliente de Redis ioredis.

npm install ioredis

Paso 6 Edite la secuencia de comandos de ejemplo para conectarse a una instancia de DCS.

Agregue el siguiente contenido al script **ioredisdemo.js**, incluyendo información sobre la conexión y la lectura de datos.

```
var Redis = require('ioredis');
var redis = new Redis({
  port: 6379,          // Redis port
  host: '192.168.0.196',   // Redis host
  family: 4,           // 4 (IPv4) or 6 (IPv6)
  password: '******',
  db: 0
});
redis.set('foo', 'bar');
redis.get('foo', function (err, result) {
  console.log(result);
});
// Or using a promise if the last argument isn't a function
redis.get('foo').then(function (result) {
  console.log(result);
});
// Arguments to commands are flattened, so the following are the same:
redis.sadd('set', 1, 3, 5, 7);
redis.sadd('set', [1, 3, 5, 7]);
// All arguments are passed directly to the redis server:
redis.set('key', 100, 'EX', 10);
```

host indica la dirección IP de ejemplo/nombre de dominio de la instancia DCS y el *port* indica el número de puerto de la instancia DCS. Para obtener más información sobre cómo obtener la dirección IP/nombre de dominio y el puerto, consulte **Paso 1**. Cambie la dirección IP/nombre de dominio y el puerto según sea necesario. ****** indica la contraseña utilizada para iniciar sesión en la instancia de DCS Redis elegida. Esta contraseña se define durante la creación de una instancia de DCS Redis.

Paso 7 Ejecute el script de ejemplo para acceder a la instancia de DCS elegida.

node ioredisdemo.js

----Fin

4.4 Acceso de la CLI web a una instancia de DCS para Redis 4.0/5.0

Acceda a una instancia de DCS Redis a través de Web CLI. Esta función solo es compatible con las instancias de DCS Redis 4.0/5.0, y no con las instancias de DCS Redis 3.0.

NOTA

- No introduzca información confidencial en Web CLI para evitar la divulgación.
- Si el valor está vacío, se devuelve **nil** después de ejecutar el comando **GET**.

Prerrequisitos:

La instancia de DCS Redis 4.0/5.0 a la que desea acceder a través de Web CLI está en el estado **Running**.

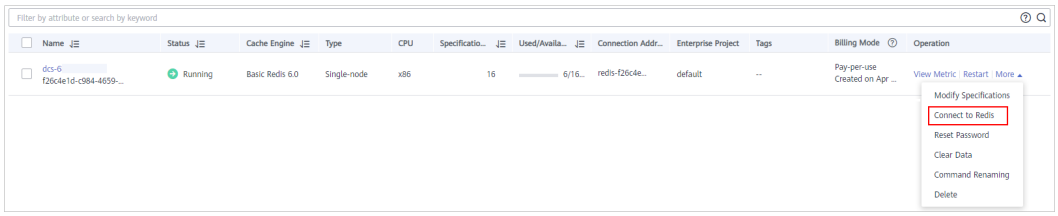
Procedimiento

Paso 1 Inicie sesión en la **consola DCS**.

Paso 2 Haga clic en en la esquina superior izquierda de la consola de gestión y seleccione la región donde se encuentra la instancia.

Paso 3 En el panel de navegación, elija **Cache Manager**. En la columna **Operation** de la instancia, elija **More** > **Connect to Redis**, como se muestra en **Figura 4-16**.

Figura 4-16 Acceso a la Web CLI



Paso 4 Introduzca la contraseña de acceso de la instancia de DCS. En Web CLI, seleccione la base de datos Redis actual, escriba un comando Redis en el cuadro de comando y presione **Enter**.

NOTA

Si no se realiza ninguna operación durante más de 5 minutos, el tiempo de conexión se agota. Debe introducir la contraseña de acceso para volver a conectarse a la instancia.

----Fin

5 Acceso a una instancia de DCS compatible con Memcached

5.1 telnet

Acceda a una instancia de DCS Memcached mediante telnet en un ECS en la misma VPC.

Prerrequisitos:

- La instancia de DCS Memcached a la que desea acceder está en el estado **Running**.
- Se ha creado un ECS en el que se ha instalado el cliente. Para obtener más información sobre cómo crear ECS, consulte la *Guía del usuario de Elastic Cloud Server*.

  NOTA

  Un ECS puede comunicarse con una instancia de DCS que pertenece a la misma VPC y está configurada con el mismo grupo de seguridad.

  - Si las instancias de ECS y de DCS están en las VPC diferentes, establezca una conexión de pares de VPC para lograr conectividad de red entre las instancias de ECS y de DCS. Para obtener más información, consulte **¿Soporta DCS el acceso entre VPC?**
  - Si se han configurado diferentes grupos de seguridad para la instancia de ECS y de DCS, establezca reglas de grupo de seguridad para lograr la conectividad de red entre la instancia de ECS y de DCS. Para obtener más información, consulte **¿Cómo configuro un grupo de seguridad?**
- Se han eliminado todas las anotaciones del código de ejemplo.
- Todas las líneas de comandos y bloques de código están codificados por UTF-8. El uso de otro esquema de codificación causará problemas de compilación o incluso fallos de comandos.

Procedimiento

Paso 1 Inicie sesión en la **consola DCS**.

Paso 2 Haga clic en  en la esquina superior izquierda de la consola de gestión y seleccione una región.

NOTA

Seleccione la misma región que su servicio de aplicación.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Paso 4 En la página **Cache Manager**, haga clic en el nombre de la instancia de DCS Memcached a la que desea tener acceso. Obtenga la dirección IP y el número de puerto de la instancia.

Paso 5 Acceda a la instancia de DCS elegida para Memcached.

1. Inicie sesión en el ECS.
2. Ejecute el siguiente comando para comprobar si telnet está instalado en el ECS:

   which telnet

   Si se muestra el directorio en el que se instala telnet, telnet se ha instalado en el ECS. Si no se muestra el directorio de instalación del cliente, instale telnet manualmente.

   NOTA

   - Si telnet no se ha instalado en Linux, ejecute el comando **yum -y install telnet** para instalarlo.
   - En el SO de Windows, elija **Start** > **Control Panel** > **Programs** > **Programs and Features** > **Turn Windows features on or off** y habilitar telnet.

3. Ejecute el siguiente comando para acceder a la instancia de DCS Memcached elegida:

   telnet ***{ip or domain name} {port}***

   En este comando: {ip address or domain name} indica la dirección IP o el nombre de dominio de la instancia de DCS Memcached. {port} indica el número de puerto de la instancia de DCS Memcached. Tanto la dirección IP o el nombre de dominio como el número de puerto se obtienen en **Paso 4**.

   Cuando ha accedido correctamente a la instancia de DCS Memcached elegida, se muestra información similar a la siguiente:

   ```
   Trying XXX.XXX.XXX.XXX...
   Connected to XXX.XXX.XXX.XXX.
   Escape character is '^]'.
   ```

   NOTA

   - Si **Password Protected** no está habilitado para la instancia, ejecute los siguientes comandos directamente después de que se haya accedido correctamente a la instancia.
   - Si **Password Protected** está habilitado para la instancia, los intentos de realizar operaciones en la instancia resultarán en el mensaje "ERROR authentication required", indicando que no tiene los permisos necesarios. En este caso, introduzca **auth** ***nombre de usuario@contraseña*** para autenticar primero. *nombre de usuario* y *contraseña* son los que se utilizan para acceder a la instancia de DCS para Memcached.

   Comandos de ejemplo para utilizar la instancia de DCS Memcached (líneas en negrita son los comandos y las otras líneas son la salida del comando):

   ```
   set hello 0 0 6
   world!
   STORED
   get hello
   VALUE hello 0 6
   world!
   END
   ```

----Fin

5.2 Java

Acceda a una instancia de DCS Memcached mediante un cliente Java en un ECS en la misma VPC.

Prerrequisitos:

- La instancia de DCS Memcached a la que desea acceder está en el estado **Running**.
- Se ha creado un ECS en el que se ha instalado el cliente. Para obtener más información sobre cómo crear ECS, consulte la *Guía del usuario de Elastic Cloud Server*.

  NOTA

  Un ECS puede comunicarse con una instancia de DCS que pertenece a la misma VPC y está configurada con el mismo grupo de seguridad.

  - Si las instancias de ECS y de DCS están en las VPC diferentes, establezca una conexión de pares de VPC para lograr conectividad de red entre las instancias de ECS y de DCS. Para obtener más información, consulte **¿Soporta DCS el acceso entre VPC?**
  - Si se han configurado diferentes grupos de seguridad para la instancia de ECS y de DCS, establezca reglas de grupo de seguridad para lograr la conectividad de red entre la instancia de ECS y de DCS. Para obtener más información, consulte **¿Cómo configuro un grupo de seguridad?**
- El kit de desarrollo Java (JDK) y los entornos de desarrollo integrados comunes (IDE) como Eclipse se han instalado en el ECS.
- Ha obtenido el paquete de dependencias **spymemcached-***x.y.z***.jar**.

  NOTA

  x.y.z indica la versión del paquete de dependencias. Se recomienda la versión más reciente.

Procedimiento

Paso 1 Inicie sesión en la **consola DCS**.

Paso 2 Haga clic en  en la esquina superior izquierda de la consola de gestión y seleccione una región.

NOTA

Seleccione la misma región que su servicio de aplicación.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Paso 4 En la página **Cache Manager**, haga clic en el nombre de la instancia de DCS Memcached a la que desea tener acceso. Obtenga la dirección IP o el nombre de dominio y el número de puerto de la instancia.

Paso 5 Sube el paquete de dependencias de **spymemcached-***x.y.z***.jar** obtenido al ECS creado.

Paso 6 Inicie sesión en el ECS.

Paso 7 Cree un proyecto Java en Eclipse e importe el paquete de dependencias **spymemcached-***x.y.z***.jar**. El nombre del proyecto es personalizable.

Paso 8 Cree una clase **ConnectMemcached1**, copie el siguiente código Java en la clase y modifique el código.

- Código de ejemplo para el modo de contraseña

  Cambie *ip or domain name:port* a la dirección IP y el número de puerto obtenido en **Paso 4**. Establezca *userName* y *password* respectivamente en el nombre de usuario y contraseña de la instancia de Memcached.

```
//Connect to the encrypted Memcached code using Java.
import java.io.IOException;
import java.util.concurrent.ExecutionException;

import net.spy.memcached.AddrUtil;
import net.spy.memcached.ConnectionFactoryBuilder;
import net.spy.memcached.ConnectionFactoryBuilder.Protocol;
import net.spy.memcached.MemcachedClient;
import net.spy.memcached.auth.AuthDescriptor;
import net.spy.memcached.auth.PlainCallbackHandler;
import net.spy.memcached.internal.OperationFuture;

public class ConnectMemcached1
{
    public static void main(String[] args)
    {
        final String connectionaddress = "ip or domain name:port";
        final String username = "userName";//Indicates the username.
        final String password = "password";//Indicates the password.
        MemcachedClient client = null;
        try
        {
            AuthDescriptor authDescriptor =
                new AuthDescriptor(new String[] {"PLAIN"}, new
PlainCallbackHandler(username,
                        password));
            client = new MemcachedClient(
                    new
ConnectionFactoryBuilder().setProtocol(Protocol.BINARY)
                            .setAuthDescriptor(authDescriptor)
                            .build(),
                    AddrUtil.getAddresses(connectionaddress));
            String key = "memcached";//Stores data with the key being
memcached in Memcached.
            String value = "Hello World";//The value is Hello World.
            int expireTime = 5; //Specifies the expiration time, measured in
seconds. The countdown starts from the moment data is written. After the
expireTime elapses, the data expires and can no longer be read.
            doExcute(client, key, value, expireTime);//Executes the operation.
        }
        catch (IOException e)
        {
            e.printStackTrace();
        }
    }

    /**
     *Method of writing data to Memcached
     */
    private static void doExcute(MemcachedClient client, String key, String
value, int expireTime)
    {
        try
        {
            OperationFuture<Boolean> future = client.set(key, expireTime,
value);
            future.get();//spymemcached set () is asynchronous. future.get ()
waits until the cache.set () operation is completed, or does not need to
wait. You can select based on actual requirements.
            System.out.println("The Set operation succeeded.");
            System.out.println("Get operation:" + client.get(key));
```



```
            Thread.sleep(6000);//Waits for 6000 ms, that is, 6s. Then the
data expires and can no longer be read.
            System.out.println("Perform the Get operation 6s later:" +
client.get(key));

        }
        catch (InterruptedException e)
        {
            e.printStackTrace();
        }
        catch (ExecutionException e)
        {
            e.printStackTrace();
        }
        if (client != null)
        {
            client.shutdown();
        }
    }
}
```

- Código de ejemplo para el modo sin contraseña

  Cambie **ip address or domain name:port** a la dirección IP y el número de puerto obtenido en **Paso 4**.

```
//Connect to the password-free Memcached code using Java.
import java.io.IOException;
import java.util.concurrent.ExecutionException;

import net.spy.memcached.AddrUtil;
import net.spy.memcached.BinaryConnectionFactory;
import net.spy.memcached.MemcachedClient;
import net.spy.memcached.internal.OperationFuture;

public class ConnectMemcached
{
    public static void main(String[] args)
    {
        final String connectionaddress = "ip or domain name:port";
        MemcachedClient client = null;
        try
        {
            client = new MemcachedClient(new BinaryConnectionFactory(),
AddrUtil.getAddresses(connectionaddress));
            String key = "memcached";//Stores data with the key being
memcached in Memcached.
            String value = "Hello World";//The value is Hello World.
            int expireTime = 5; //Specifies the expiration time, measured in
seconds. The countdown starts from the moment data is written. After the
expireTime elapses, the data expires and can no longer be read.
            doExcute(client, key, value, expireTime);//Executes the operation.
        }
        catch (IOException e)
        {
            e.printStackTrace();
        }
    }

    /**
     *Method of writing data to Memcached
     */
    private static void doExcute(MemcachedClient client, String key, String
value, int expireTime)
    {
        try
        {
            OperationFuture<Boolean> future = client.set(key, expireTime,
value);
            future.get();//spymemcached set () is asynchronous. future.get ()
waits until the cache.set () operation is completed, or does not need to
```



```
wait. You can select based on actual requirements.
            System.out.println("The Set operation succeeded.");
            System.out.println("Get operation:" + client.get(key));
            Thread.sleep(6000);//Waits for 6000 ms, that is, 6s. Then the
data expires and can no longer be read.
            System.out.println("Perform the Get operation 6s later:" +
client.get(key));

        }
        catch (InterruptedException e)
        {
            e.printStackTrace();
        }
        catch (ExecutionException e)
        {
            e.printStackTrace();
        }
        if (client != null)
        {
            client.shutdown();
        }
    }
}
```

Paso 9 Ejecute el método principal. El siguiente resultado se muestra en la ventana **Console** de Eclipse:

```
The Set operation succeeded.
Get operation: Hello World
Perform the Get operation 6s later: null
```

----Fin

5.3 Python

Acceda a una instancia de DCS Memcached mediante Python en un ECS en la misma VPC.

Prerrequisitos:

- La instancia de DCS Memcached a la que desea acceder está en el estado **Running**.
- Inicie sesión en el ECS. Para obtener más información sobre cómo crear ECS, consulte la *Guía del usuario de Elastic Cloud Server*.

  NOTA

  Un ECS puede comunicarse con una instancia de DCS que pertenece a la misma VPC y está configurada con el mismo grupo de seguridad.

  - Si las instancias de ECS y de DCS están en las VPC diferentes, establezca una conexión de pares de VPC para lograr conectividad de red entre las instancias de ECS y de DCS. Para obtener más información, consulte **¿Soporta DCS el acceso entre VPC?**
  - Si se han configurado diferentes grupos de seguridad para la instancia de ECS y de DCS, establezca reglas de grupo de seguridad para lograr la conectividad de red entre la instancia de ECS y de DCS. Para obtener más información, consulte **¿Cómo configuro un grupo de seguridad?**
- Python se ha instalado en el ECS. La versión recomendada es 2.7.6 o posterior.
- Ha obtenido el paquete de dependencias **python-binary-memcached-x.y.z.zip**.

  NOTA

  x.y.z indica la versión del paquete de dependencias. Se recomienda la versión más reciente.

Procedimiento

Paso 1 Inicie sesión en la **consola DCS**.

Paso 2 Haga clic en en la esquina superior izquierda de la consola de gestión y seleccione una región.

NOTA

Seleccione la misma región que su servicio de aplicación.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Paso 4 En la página **Cache Manager**, haga clic en el nombre de la instancia de DCS Memcached a la que desea tener acceso. Obtenga la dirección IP o el nombre de dominio y el número de puerto de la instancia.

Paso 5 Sube el paquete de dependencias obtenido (por ejemplo, el paquete **python-binary-memcached-*x.y.z*.zip**) al ECS creado.

Paso 6 Inicie sesión en el ECS.

Paso 7 Ejecute los siguientes comandos para instalar el paquete de dependencias:

unzip -xzvf python-binary-memcached-x.y.z.zip

cd python-binary-memcached-x.y.z

python setup.py install

NOTA

Si se informa de un error durante la instalación, utilice el método de instalación **apt** o **yum**. Por ejemplo, para instalar el paquete de dependencias mediante el método **apt**, ejecute los siguientes comandos:

apt install python-pip;

pip install python-binary-memcached;

Paso 8 Cree un archivo de Python llamado **dcs_test.py**, copie el siguiente código de Python en el archivo y modifique el código.

- Código de ejemplo para el modo de contraseña

  Cambie *ip or domain name:port* a la dirección IP o nombre de dominio y número de puerto obtenido en **Paso 4**. Establezca *userName* y *password* respectivamente en el nombre de usuario y contraseña de la instancia de Memcached.

```
import bmemcached
client = bmemcached.Client(('ip or domain name:port'), 'userName', 'password')
print "set('key', 'hello world!')"
print client.set('key', 'hello world!')
print "get('key')"
print client.get('key')
```

- Código de ejemplo para el modo sin contraseña

  Cambie **ip address or domain name:port** a la dirección IP y el número de puerto obtenido en **Paso 4**.

```
import bmemcached
client = bmemcached.Client('ip or domain name:port')
print "set('key', 'hello world!')"
print client.set('key', 'hello world!')
print "get('key')"
print client.get('key')
```


Paso 9 Ejecute el archivo **dcs_test.py**. Se muestra el siguiente resultado.

```
# python test.py
set('key', 'hello world!')
True
get('key')
hello world!
```

----Fin

5.4 C++

Acceda a una instancia de DCS Memcached mediante un cliente C++ en un ECS en la misma VPC.

Prerrequisitos:

- La instancia de DCS Memcached a la que desea acceder está en el estado **Running**.
- Inicie sesión en el ECS. Para obtener más información sobre cómo crear ECS, consulte la *Guía del usuario de Elastic Cloud Server*.

  NOTA

  Un ECS puede comunicarse con una instancia de DCS que pertenece a la misma VPC y está configurada con el mismo grupo de seguridad.
  - Si las instancias de ECS y de DCS están en las VPC diferentes, establezca una conexión de pares de VPC para lograr conectividad de red entre las instancias de ECS y de DCS. Para obtener más información, consulte **¿Soporta DCS el acceso entre VPC?**
  - Si se han configurado diferentes grupos de seguridad para la instancia de ECS y de DCS, establezca reglas de grupo de seguridad para lograr la conectividad de red entre la instancia de ECS y de DCS. Para obtener más información, consulte **¿Cómo configuro un grupo de seguridad?**
- GCC se ha instalado en el ECS. La versión recomendada es 4.8.4 o posterior.
- Ha obtenido el paquete de dependencias **libmemcached-x.y.z.tar.gz**.

  NOTA

  x.y.z indica la versión del paquete de dependencias. Se recomienda la versión más reciente.

Procedimiento

Paso 1 Inicie sesión en la **consola DCS**.

Paso 2 Haga clic en en la esquina superior izquierda de la consola de gestión y seleccione una región.

NOTA

Seleccione la misma región que su servicio de aplicación.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Paso 4 En la página **Cache Manager**, haga clic en el nombre de la instancia de DCS Memcached a la que desea tener acceso. Obtenga la dirección IP o el nombre de dominio y el número de puerto de la instancia.

Paso 5 Sube el paquete de dependencias de **libmemcached-*x.y.z*.tar.gz** obtenido al ECS creado.

Paso 6 Inicie sesión en el ECS.

Paso 7 Instalar paquetes de dependencias SASL relacionados.

Para SO de la serie Debian: **apt install libsasl2-dev cloog.ppl**

Para SO de la serie Red Hat: **yum install cyrus-sasl***

Paso 8 Ejecute los siguientes comandos para instalar el paquete de dependencias:

tar -xzvf libmemcached-x.y.z.tar.gz

cd libmemcached-x.y.z

./configure --enable-sasl

make

make install

Paso 9 Cree un archivo llamado **build.sh** y copie el siguiente código en el archivo.

```
g++ -o dcs_sample dcs_sample.cpp -lmemcached -std=c++0x -lpthread -lsasl2
```

NOTA

Si el archivo **libmemcached.so.11** no se encuentra durante la compilación, ejecute el comando **find** para encontrar el archivo y copie el archivo en el directorio **/usr/lib**.

Paso 10 Cree un archivo llamado **dcs_sample.cpp**, copie el siguiente código de C++ en el archivo y modifique el código.

- Código de ejemplo para el modo de contraseña

  Cambie *ip or domain name* y *port* a la dirección IP o el nombre de dominio y el número de puerto obtenido en **Paso 4**. Establezca *userName* y *password* respectivamente en el nombre de usuario y contraseña de la instancia de Memcached.

```
#include <iostream>
#include <string>
#include <libmemcached/memcached.h>
using namespace std;

#define IP  "ip or domain name"
#define PORT  "port"
#define USERNAME "userName"
#define PASSWORD "password"
memcached_return result;

memcached_st * init()
{
    memcached_st *memcached = NULL;
    memcached_server_st *cache;
    memcached = memcached_create(NULL);
    cache = memcached_server_list_append(NULL, IP, PORT, &result);

    sasl_client_init(NULL);
         memcached_set_sasl_auth_data(memcached, USERNAME, PASSWORD);

memcached_behavior_set(memcached,MEMCACHED_BEHAVIOR_BINARY_PROTOCOL,1);
         memcached_server_push(memcached,cache);
         memcached_server_list_free(cache);
            return memcached;
}

int main(int argc, char *argv[])
{
         memcached_st *memcached=init();
    string key = "memcached";
```



```
    string value = "hello world!";
    size_t value_length = value.length();
    int expire_time = 0;
    uint32_t flag = 0;

    result =
memcached_set(memcached,key.c_str(),key.length(),value.c_str(),value.length(),
expire_time,flag);
    if (result != MEMCACHED_SUCCESS){
      cout <<"set data failed: " << result << endl;
      return -1;
    }
    cout << "set succeed, key: " << key << ", value: " << value << endl;
    cout << "get key:" << key << endl;
    char* result =
memcached_get(memcached,key.c_str(),key.length(),&value_length,&flag,&result);
    cout << "value:" << result << endl;
    memcached_free(memcached);
    return 0;
}
```

- Código de ejemplo para el modo sin contraseña

  Cambie *ip or domain name* por la dirección IP o el nombre de dominio y el número de puerto obtenido en **Paso 4**.

```
#include <iostream>
#include <string>
#include <libmemcached/memcached.h>
using namespace std;

#define IP  "ip or domain name"
#define PORT  port
memcached_return result;

memcached_st * init()
{
    memcached_st *memcached = NULL;
    memcached_server_st *cache;
    memcached = memcached_create(NULL);
    cache = memcached_server_list_append(NULL, IP, PORT, &result);
         memcached_server_push(memcached,cache);
    memcached_server_list_free(cache);
         return memcached;
}

int main(int argc, char *argv[])
{
         memcached_st *memcached=init();
    string key = "memcached";
    string value = "hello world!";
    size_t value_length = value.length();
    int expire_time = 0;
    uint32_t flag = 0;

    result =
memcached_set(memcached,key.c_str(),key.length(),value.c_str(),value.length(),
expire_time,flag);
    if (result != MEMCACHED_SUCCESS){
      cout <<"set data failed: " << result << endl;
      return -1;
    }
    cout << "set succeed, key: " << key << " ,value: " << value << endl;
    cout << "get key:" << key << endl;
    char* result =
memcached_get(memcached,key.c_str(),key.length(),&value_length,&flag,&result);
    cout << "value:" << result << endl;
    memcached_free(memcached);
    return 0;
}
```


Paso 11 Ejecute los siguientes comandos para compilar el código fuente:

chmod 700 build.sh

./build.sh

The **dcs_sample** binary file is generated.

Paso 12 Ejecute el siguiente comando para acceder a la instancia de DCS Memcached elegida:

./dcs_sample

```
set succeed, key: memcached ,value: hello world!
get key:memcached
value:hello world!
```

----Fin

5.5 PHP

Acceda a una instancia de DCS Memcached en PHP en un ECS en la misma VPC.

Prerrequisitos:

- La instancia de DCS Memcached a la que desea acceder está en el estado **Running**.
- Inicie sesión en el ECS. Para obtener más información sobre cómo crear ECS, consulte la *Guía del usuario de Elastic Cloud Server*.

  NOTA

  Un ECS puede comunicarse con una instancia de DCS que pertenece a la misma VPC y está configurada con el mismo grupo de seguridad.

  - Si las instancias de ECS y de DCS están en las VPC diferentes, establezca una conexión de pares de VPC para lograr conectividad de red entre las instancias de ECS y de DCS. Para obtener más información, consulte **¿Soporta DCS el acceso entre VPC?**
  - Si se han configurado diferentes grupos de seguridad para la instancia de ECS y de DCS, establezca reglas de grupo de seguridad para lograr la conectividad de red entre la instancia de ECS y de DCS. Para obtener más información, consulte **¿Cómo configuro un grupo de seguridad?**

SO de la serie Red Hat

A continuación se utiliza CentOS 7.0 como ejemplo para describir cómo instalar un cliente PHP y usarlo para acceder a una instancia de DCS Memcached. El procedimiento también es aplicable a un cliente PHP que ejecute Red Hat o Fedora OS.

Paso 1 Instale los componentes de compilación GCC-C++ y Make.

yum install gcc-c++ make

Paso 2 Instalar paquetes de SASL relacionados.

yum install cyrus-sasl*

Paso 3 Instale la biblioteca libMemcached.

La instalación de la biblioteca libMemcached requiere parámetros de autenticación SASL. Por lo tanto, no puede instalar la biblioteca ejecutando el comando **yum**.

wget https://launchpad.net/libmemcached/1.0/1.0.18/+download/libmemcached-1.0.18.tar.gz

tar -xvf libmemcached-1.0.18.tar.gz

cd libmemcached-1.0.18

./configure --prefix=/usr/local/libmemcached --enable-sasl

make && make install

NOTA

Antes de instalar la biblioteca libMemcached, instale los componentes GCC-C++ y SASL. De lo contrario, se informará de un error durante la compilación. Después de resolver el error, ejecute el comando **make clean** y, a continuación, ejecute el comando **make** de nuevo.

Paso 4 Instale el entorno PHP.

yum install php-devel php-common php-cli

AVISO

PHP 7.x no soporta autenticación SASL. Use PHP 5.6. Si la versión yum php no es 5.6, descargue una de Internet.

Paso 5 Instale el cliente Memcached.

Tenga en cuenta que debe agregar un parámetro utilizado para habilitar SASL al ejecutar el comando **configure**.

wget http://pecl.php.net/get/memcached-2.1.0.tgz

tar zxvf memcached-2.1.0.tgz

cd memcached-2.1.0

phpize

./configure --with-libmemcached-dir=/usr/local/libmemcached --enable-memcached-sasl

make && make install

Paso 6 Modifique el archivo **php.ini**.

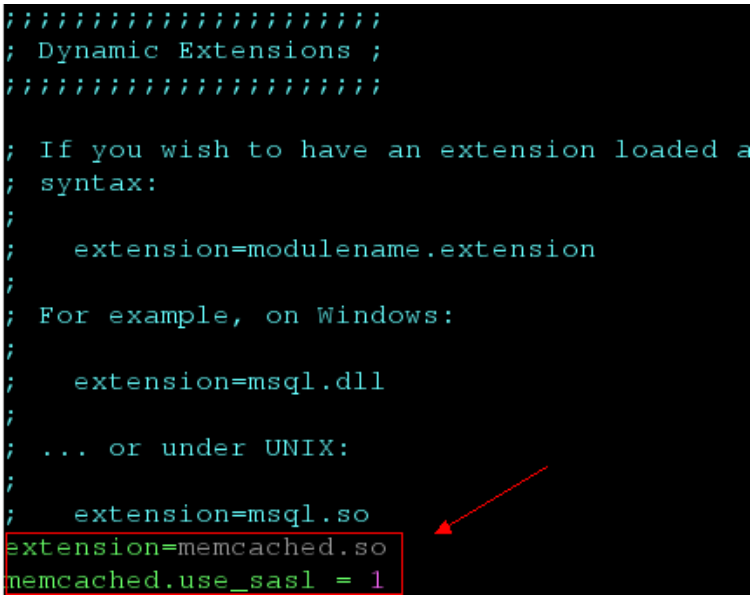
Run the **find** or **locate** command to find the **php.ini** file.

find / -name php.ini

Añada las siguientes dos líneas al archivo **php.ini**:

```
extension=memcached.so
memcached.use_sasl = 1
```


Figura 5-1 Modificar el archivo **php.ini**



Paso 7 Acceda a una instancia de DCS Memcached.

Cree un archivo **memcached.php** y agregue el siguiente contenido al archivo:

```
<?php
    $connect = new Memcached; //Declares a Memcached connection.
    $connect->setOption(Memcached::OPT_COMPRESSION, false); //Disables
compression.
    $connect->setOption(Memcached::OPT_BINARY_PROTOCOL, true); //Uses the binary
protocol.
    $connect->setOption(Memcached::OPT_TCP_NODELAY, true); //Disables the TCP
network delay policy.
    $connect->addServer('{memcached_instance_ip}', 11211); //Specifies the
instance IP address and port number.
    $connect->setSaslAuthData('{username}', '{password}'); //If password-free
access is enabled for the instance, delete or comment out this line.
    $connect->set("DCS", "Come on!");
    echo 'DCS: ',$connect->get("DCS");
    echo "\n";
    $connect->quit();
?>
```

Guarde y ejecute el archivo **memcached.php**. Se muestra el siguiente resultado.

```
[root@testphpmemcached ~]# php memcached.php
    DCS: Come on!
[root@testphpmemcached ~]#
```

----Fin

SO de la serie Debian

El siguiente ejemplo utiliza el SO de Ubuntu para describir cómo instalar un cliente PHP y usarlo para acceder a una instancia de DCS Memcached.

Paso 1 Instale los componentes de compilación de GCC y Make.

apt install gcc make

Paso 2 Instale el entorno PHP.

Se recomienda PHP 5.x para una mejor compatibilidad con la autenticación SASL.

Ejecute los siguientes comandos para agregar la fuente de imagen de PHP de una versión anterior, y luego instale los paquetes **php.5.6** y **php.5.6-dev**:

apt-get install -y language-pack-en-base;

LC_ALL=en_US.UTF-8;

add-apt-repository ppa:ondrej/php;

apt-get update;

apt-get install php5.6 php5.6-dev;

Una vez completada la instalación, ejecute el comando **php -version** para comprobar la versión de PHP. Si se muestra el siguiente resultado, la versión de PHP es 5.6, lo que indica que PHP 5.6 se ha instalado correctamente.

```
root@dcs-nodelete:/etc/apt# php -version
PHP 5.6.36-1+ubuntu16.04.1+deb.sury.org+1 (cli)
Copyright (c) 1997-2016 The PHP Group
```

NOTA

Para desinstalar PHP, ejecute los siguientes comandos:

apt install aptitude -y

aptitude purge `dpkg -l | grep php| awk '{print $2}' |tr "\n" " "`

Paso 3 Instale el componente SASL.

apt install libsasl2-dev cloog.ppl

Paso 4 Instale la biblioteca libMemcached.

wget https://launchpad.net/libmemcached/1.0/1.0.18/+download/libmemcached-1.0.18.tar.gz

tar -xvf libmemcached-1.0.18.tar.gz

cd libmemcached-1.0.18

./configure --prefix=/usr/local/libmemcached

make && make install

NOTA

Antes de instalar la biblioteca libMemcached, instale los componentes GCC-C++ y SASL. De lo contrario, se informará de un error durante la compilación. Después de resolver el error, ejecute el comando **make clean** y, a continuación, ejecute el comando **make** de nuevo.

Paso 5 Instale el cliente Memcached.

Instale el componente zlib.

apt install zlib1g.dev

Tenga en cuenta que debe agregar un parámetro utilizado para habilitar SASL al ejecutar el comando **configure**.

wget http://pecl.php.net/get/memcached-2.2.0.tgz;

tar zxvf memcached-2.2.0.tgz;

cd memcached-2.2.0;

phpize5.6;

./configure --with-libmemcached-dir=/usr/local/libmemcached --enable-memcached-sasl;

make && make install;

Paso 6 Modifique el archivo **pdo.ini**.

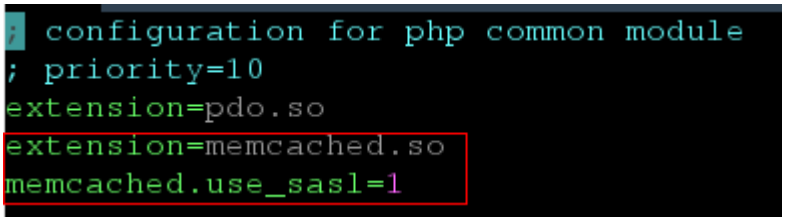
Ejecute el siguiente comando para encontrar el archivo **pdo.ini**:

find / -name pdo.ini

Por defecto, el archivo **pdo.ini** se almacena en el directorio **/etc/php/5.6/mods-available**. Añada las siguientes dos líneas al archivo **php.ini**:

```
extension=memcached.so
memcached.use_sasl = 1
```

Figura 5-2 Modificar el archivo **pdo.ini**



Paso 7 Acceda a una instancia de DCS Memcached.

Cree un archivo **memcached.php** y agregue el siguiente contenido al archivo:

```
<?php
    $connect = new Memcached; //Declares a Memcached connection.
    $connect->setOption(Memcached::OPT_COMPRESSION, false); //Disables
compression.
    $connect->setOption(Memcached::OPT_BINARY_PROTOCOL, true); //Uses the binary
protocol.
    $connect->setOption(Memcached::OPT_TCP_NODELAY, true); //Disables the TCP
network delay policy.
    $connect->addServer('{memcached_instance_ip}', 11211); //Specifies the
instance IP address and port number.
    $connect->setSaslAuthData('{username}', '{password}'); //If password-free
access is enabled for the instance, delete or comment out this line.
    $connect->set("DCS", "Come on!");
    echo 'DCS: ',$connect->get("DCS");
    echo "\n";
    $connect->quit();
?>
```

Guarde y ejecute el archivo **memcached.php**. Se muestra el siguiente resultado.

```
[root@dcs-nodelete ~]# php memcached.php
    DCS: Come on!
[root@dcs-nodelete ~]#
```

----Fin

6 Funcionamiento de instancias de DCS

6.1 Consulta de detalles de instancia

En la consola de DCS, puede consultar los detalles de la instancia de DCS.

NOTA

DCS for Redis 3.0 ya no se proporciona. Puede utilizar DCS for Redis 4.0 o 5.0 en su lugar.

Procedimiento

Paso 1 Inicie sesión en la **consola DCS**.

Paso 2 Haga clic en en la esquina superior izquierda de la consola de gestión y seleccione una región.

NOTA

Seleccione la misma región que su servicio de aplicación.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Paso 4 Busque instancias DCS mediante cualquiera de los métodos siguientes:

- Busque por palabra clave.

  Ingresar una palabra clave para buscar.
- Seleccione los atributos e introduzca las palabras clave que desee buscar.

  Actualmente, puede buscar por nombre, especificación, ID, dirección IP, AZ, estado, tipo de instancia, motor de caché, proyecto de empresa, modo de facturación y etiquetas.

  Por ejemplo, para filtrar instancias DCS por motor de caché o versión del motor de caché, haga clic en el cuadro de búsqueda, elija **Cache Engine**, y luego elija **Redis**, **Redis 3.0**, **Redis 4.0**, **Redis 5.0**, o **Memcached**.

Para obtener más información sobre cómo buscar, haga clic en el signo de interrogación situado a la derecha del cuadro de búsqueda.

Paso 5 Haga clic en el nombre de la instancia de DCS para mostrar más detalles sobre la instancia de DCS. **Tabla 6-1** describe los parámetros.

Tabla 6-1 Parámetros de la página Información básica de una instancia DCS

Sección	Parámetro	Descripción
Instance Details (Detalles de la instancia)	Name	Nombre de la instancia elegida. Para modificar el nombre de la instancia, haga clic en el icono  .
	Status	Estado de la instancia elegida.
	ID	ID de la instancia elegida.
	Cache Engine	Motor de caché utilizado por la instancia DCS, que puede ser Redis o Memcached. Si el motor de caché es Redis, es seguido por el número de versión, por ejemplo, Redis 3.0.
	Instance Type	Tipo de la instancia seleccionada. Actualmente, los tipos soportados incluyen nodo único, principal/en standby, Clúster Proxy, y Clúster Redis.
	Cache Size	Especificación de la instancia elegida.
	Used/ Available Memory (MB)	El espacio de memoria utilizado y el espacio de memoria máximo disponible de la instancia elegida. El espacio de memoria utilizado incluye: ● Tamaño de los datos almacenados en la instancia DCS ● Tamaño de los búferes del servidor Redis (incluidos los búferes del cliente y repl-backlog) y las estructuras de datos internas
	CPU	Arquitectura de CPU de la instancia elegida. Este parámetro sólo se muestra para instancias de DCS Redis.
	Enterprise Project	Proyecto de empresa al que pertenece la nueva instancia. Haga clic en  para ver el proyecto de empresa de la instancia.
	Maintenance	Rango de tiempo para cualquier actividad de mantenimiento programada en los nodos de caché de esta instancia de DCS. Para modificar la ventana, haga clic en el icono  .
	Description	Descripción de la instancia de DCS elegida. Para modificar la descripción, haga clic en el icono  .
Connection (Conexión)	Password Protected	Acceso protegido con contraseña o sin contraseña.

Sección	Parámetro	Descripción
	Connection Address	Nombre de dominio y número de puerto de la instancia. Puede hacer clic en  junto a **Connection Address** para cambiar el puerto. **NOTA** ● Para una instancia principal/standby de Redis 4.0/5.0, esta dirección indica el nombre de dominio y el número de puerto del nodo principal. **Read-only Address** es el nombre de dominio y el número de puerto del nodo en standby. Al conectarse a una instancia de este tipo, puede utilizar el nombre de dominio y el número de puerto del nodo principal o el nodo en standby. ● Puede cambiar el puerto solo para una instancia de DCS Redis 4.0, 5.0 o 6.0, pero no para una instancia de DCS Redis 3.0 o Memcached.
	IP Address	Dirección IP y número de puerto de la instancia. Se recomienda la dirección de conexión del nombre de dominio.
	Public Access	Un indicador de si el acceso público está habilitado. El acceso público solo se admite para Redis 3.0 y no para 5.0, 4.0 y Memcached.
	Public Access Address	EIP vinculado a la instancia de acceso público. Este parámetro sólo se muestra cuando el **Public Access** está habilitado. **NOTA** Haga clic en **Download Certificate for Public Access** para descargar un certificado, que se puede utilizar para verificar el certificado de la instancia de DCS cuando se accede a la instancia.
Network (Red)	AZ	Zona de disponibilidad en la que residen los nodos de caché que ejecutan la instancia de DCS seleccionada.
	VPC	VPC en la que reside la instancia elegida.
	Subnet	Subred en la que reside la instancia elegida.
	Security Group	Grupo de seguridad que controla el acceso a la instancia elegida. Para modificar el grupo de seguridad, haga clic en el icono  . El control de acceso a grupos de seguridad solo es compatible con instancias de DCS Redis 3.0 y Memcached. DCS for Redis 4.0/5.0 se basa en VPC Endpoint y no admite grupos de seguridad.
Instance Topology (Topología de la instancia)	-	Pase el cursor sobre un nodo para ver sus métricas o haga clic en el icono de un nodo para ver sus métricas históricas. Las topologías solo se soportan para instancias principal/en standby, de Clúster Proxy y de Clúster Redis.

Sección	Parámetro	Descripción
Billing (Facturación)	Billing Mode	Modo de facturación de la instancia.
	Created	Hora en la que comenzó a crearse la instancia elegida.
	Run	Hora en la que se creó la instancia.

----Fin

6.2 Modificación de las especificaciones

En la consola de DCS, puede escalar una instancia de DCS Redis o Memcached a una capacidad mayor o menor, o cambiar el tipo de instancia.

NOTA

- Modifique las especificaciones de las instancias en períodos de poca actividad. Si la modificación falló en las horas pico , (por ejemplo, cuando el uso de memoria o CPU es superior al 90% o cuando el tráfico de escritura aumenta) Inténtalo de nuevo durante las horas no pico.
- Si las instancias de DCS son demasiado antiguas para admitir la modificación de la especificación, póngase en contacto con el soporte técnico para actualizar las instancias.
- DCS for Redis 3.0 ya no se proporciona. Puede utilizar DCS for Redis 4.0 o 5.0 en su lugar.

Cambio del tipo de instancia

Tabla 6-2 Opciones de cambio de tipo de instancia admitidas por diferentes instancias de DCS

Versión	Cambio de tipo admitido	Precauciones
Redis 3.0	De nodo único a principal/en standby	La instancia no se puede conectar durante varios segundos y permanece de sólo lectura durante aproximadamente un minuto.
	De principal/en standby a Clúster Proxy	1. Si los datos de una instancia de DCS Redis 3.0 principal/en standby se almacenan en múltiples bases de datos, o en bases de datos que no son DB0, la instancia no se puede cambiar al tipo de proxy Clúster. Una instancia principal/en standby se puede cambiar al tipo Clúster Proxy solo si sus datos se almacenan solo en DB0. 2. La instancia no se puede conectar y permanece de sólo lectura durante 5 a 30 minutos.
Memcached	De nodo único a principal/en standby	Los servicios se interrumpen durante varios segundos y permanecen de solo lectura durante aproximadamente 1 minuto.

Versión	Cambio de tipo admitido	Precauciones
Redis 4.0/5.0	De principal/en standby a Clúster Proxy De la separación de lectura/escritura a Clúster Proxy De Clúster Proxy a principal/en standby De Clúster Proxy a la separación de lectura/escritura	1. Antes de cambiar el tipo de instancia a Clúster Proxy, evalúe el impacto en los servicios. Para obtener más información, consulte **¿Cuáles son las restricciones en la implementación de Multi-DB en una instancia Clúster Proxy?** y **Restricciones de comando**. 2. El uso de memoria debe ser inferior al 70% de la memoria máxima de la nueva variante. 3. Algunas claves pueden ser desalojadas si el uso actual de la memoria excede el 90% del total. 4. Después del cambio, **cree reglas de alarma** de nuevo para la instancia. 5. Para las instancias que actualmente son principales/en standby, asegúrese de que su dirección IP o nombre de dominio de solo lectura no sean utilizados por su aplicación. 6. Si la aplicación no puede volver a conectarse a Redis o controlar las excepciones, es posible que tenga que reiniciar la aplicación después del cambio. 7. Modifique las especificaciones de las instancias en períodos de poca actividad. Una instancia se interrumpe temporalmente y permanece en estado de solo lectura por aproximadamente 1 minuto durante el cambio de especificaciones.

No se admiten los cambios de tipo de instancia que no se enumeran en la tabla anterior. Para modificar las especificaciones mientras se cambia el tipo de instancia, consulte **Conmutación IP**.

Ajuste de escala

- Opciones de escala

Tabla 6-3 Opciones de escala compatibles con diferentes instancias

Motor de memoria caché	Nodo único	Principal/En standby	Clúster Redis	Clúster Proxy	Separación de lecturas/escrituras
Redis 3.0	Escalar hacia arriba/hacia abajo	Escalar hacia arriba/hacia abajo	Escalar hacia arriba/hacia abajo	Escalando hacia arriba	-
Redis 4.0	Escalar hacia arriba/hacia abajo	Escalado hacia arriba/abajo, hacia fuera/entra	Escalado hacia arriba/abajo, hacia fuera/entra	Escalar hacia arriba/hacia abajo	Escalado hacia arriba/abajo, hacia fuera/entra
Redis 5.0	Escalar hacia arriba/hacia abajo	Escalado hacia arriba/abajo, hacia fuera/entra	Escalado hacia arriba/abajo, hacia fuera/entra	Escalar hacia arriba/hacia abajo	Escalado hacia arriba/abajo, hacia fuera/entra
Memcached	Escalar hacia arriba/hacia abajo	Escalar hacia arriba/hacia abajo	-	-	-
Redis 6.0 edición básica	Escalar hacia arriba/hacia abajo	Escalar hacia arriba/hacia abajo	-	-	-
Redis 6.0 ediciones profesionales	-	No se admite ningún cambio.	-	-	-

NOTA

Si la memoria reservada de una instancia de DCS Redis 3.0 o Memcached es insuficiente, la modificación puede fallar cuando se agota la memoria. Para obtener más información, consulte **Memoria reservada**.

- Impacto de la escala

Tabla 6-4 Impacto de la escala

Tipo de instancia	Tipo de escala	Impacto
Nodo único, principal/en standby y separación de lectura/escritura	Escalar hacia arriba/hacia abajo	● Una instancia de DCS Redis 4.0 o 5.0 se desconectará durante varios segundos y permanecerá de sólo lectura durante aproximadamente 1 minuto. Una instancia de DCS Redis 3.0 se desconectará y permanecerá de sólo lectura durante 5 a 30 minutos. ● Para escalar, solo se expande la memoria de la instancia. La capacidad de procesamiento de la CPU no se mejora. ● Las instancias DCS de nodo único no admiten persistencia de datos. Los datos no se conservan durante el escalado. Después de escalar, compruebe si los datos están completos e importe los datos si es necesario. Si hay datos importantes, utilice una herramienta de migración para migrar los datos a otras instancias para realizar copias de seguridad. ● Los registros de copia de seguridad de instancias principal/en standby y de separación lectura/escritura no se pueden restaurar después de escalar.

Tipo de instancia	Tipo de escala	Impacto
Clúster Proxy y Clúster Redis	Escalar hacia arriba/hacia abajo	● El escalado implica la migración de datos, lo que aumenta la latencia de acceso. Para una instancia de Clúster Redis, asegúrese de que el cliente puede procesar correctamente los comandos MOVED y ASK. De lo contrario, las solicitudes fallarán. ● Si la memoria se llena durante el escalado debido a que se escribe una gran cantidad de datos, el escalado fallará. ● No se pueden restaurar los registros de copia de seguridad creados antes de la escala. ● Antes de escalar, compruebe si hay claves grandes a través de Análisis de caché. Redis tiene un límite en la migración de claves. Si la instancia tiene una clave única superior a 512 MB, el escalado fallará cuando se agote el tiempo de migración de clave grande entre nodos. Cuanto más grande sea la clave, más probabilidades hay de que la migración falle. ● Antes de escalar hacia arriba o hacia abajo una instancia de Clúster Redis, asegúrese de que la actualización automatizada de la topología del clúster esté habilitada si usa Lettuce. Si está deshabilitado, tendrá que reiniciar el cliente después de escalar. Para más detalles sobre cómo habilitar la actualización automatizada, vea **un ejemplo de uso de Lettuce para conectarse a una instancia de Clúster Redis**. ● El escalamiento no interrumpe las conexiones, sino que ocupará los recursos de la CPU, lo que reducirá el rendimiento hasta en un 20%. ● Durante la ampliación, se agregan nuevos nodos del servidor Redis y los datos se equilibran automáticamente en los nuevos nodos. ● Para reducir la escala de una instancia, asegúrese de que la memoria utilizada de cada nodo sea inferior al 70% de la memoria máxima por nodo del nuevo sabor. ● Si la cantidad de particiones disminuye durante la reducción de la escala, los nodos se eliminarán. Antes de reducir la escala, asegúrese de que los nodos eliminados no se referencian directamente en la aplicación, para evitar excepciones de acceso al servicio. ● Si la cantidad de particiones disminuye durante la reducción de la escala, los nodos se eliminarán y las conexiones se interrumpirán. Si la aplicación no puede volver a conectarse a Redis o controlar las excepciones, es posible que tenga que reiniciar la aplicación después de escalar.

Tipo de instancia	Tipo de escala	Impacto
Instancias principal/en standby, de separación de lectura/escritura y de Clúster Redis	Escalado de salida / entrada (cambio de la cantidad de réplicas)	● Antes de escalar hacia fuera o entra una instancia de Clúster Redis, asegúrese de que la actualización automatizada de la topología del clúster esté habilitada si usa Lettuce. Si está deshabilitado, tendrá que reiniciar el cliente después de escalar. Para más detalles sobre cómo habilitar la actualización automatizada, vea **un ejemplo de uso de Lettuce para conectarse a una instancia de Clúster Redis**. ● La eliminación de réplicas interrumpe las conexiones. Si la aplicación no puede volver a conectarse a Redis o manejar excepciones, debe reiniciar la aplicación después de escalar. ● Si el número de réplicas ya es el mínimo admitido por la instancia, ya no podrá eliminar réplicas.

Procedimiento

Paso 1 Inicie sesión en la **consola DCS**.

Paso 2 Haga clic en en la esquina superior izquierda de la consola de gestión y seleccione la región donde se encuentra la instancia.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Paso 4 Seleccione **More** > **Modify Specifications** en la fila que contiene la instancia DCS.

Paso 5 En la página **Modify Specifications**, seleccione la especificación deseada.

NOTA

Si decide cambiar la cantidad de réplicas para una instancia de Clúster Redis, el campo **Added Replicas** se mostrará en la página. La modificación de la especificación de instancia hará que se modifique la especificación de réplica, lo que da como resultado un cambio en el precio.

Paso 6 Establezca **Apply Change** en **Now** o **During maintenance**.

Seleccione **During maintenance** si la modificación interrumpe las conexiones.

Tabla 6-5 Escenarios donde la modificación de la especificación interrumpe las conexiones

Cambio	Cuando se interrumpen las conexiones
Escalar una instancia de nodo único o principal/en standby	La memoria aumenta de un tamaño inferior a 8 GB a 8 GB o más.
Escalar una instancia de Clúster Proxy y de Clúster Redis	Se reduce el número de las particiones.

Cambio	Cuando se interrumpen las conexiones
Cambio del tipo de instancia	El tipo de instancia se cambia entre principal/standby o de la separación de lectura/escritura y de Clúster Proxy.
Supresión de réplicas	Las réplicas se eliminan de una instancia principal/en standby, de Clúster Redis o de separación de lectura/escritura.

NOTA

- Si la modificación no interrumpe las conexiones, se aplicará inmediatamente incluso si selecciona **During maintenance**.
- La modificación no puede ser retirada una vez presentada. Para reprogramar una modificación, puede cambiar la ventana de mantenimiento. La ventana de mantenimiento se puede cambiar hasta tres veces.
- Las modificaciones en las instancias DCS Redis 3.0 y Memcached solo se pueden aplicar de inmediato.

Paso 7 Haga clic en **Next**, confirme los detalles y haga clic en **Submit**.

Puede ir a la página **Background Tasks** para ver el estado de modificación. Para obtener más información, consulte **Consulta de tareas del fondo**.

La modificación de la especificación de una instancia DCS de nodo único o principal/en standby tarda aproximadamente de 5 a 30 minutos en completarse, mientras que la de una instancia DCS de clúster tarda más tiempo. Una vez que una instancia se ha modificado correctamente, cambia al estado **Running**.

NOTA

- Si la modificación de la especificación de una instancia DCS de nodo único falla, la instancia no está disponible temporalmente para su uso. La especificación permanece sin cambios. Algunas operaciones de gestión (como la configuración de parámetros y la modificación de especificaciones) no se admiten temporalmente. Una vez completada la modificación de la especificación en el backend, la instancia cambia a la nueva especificación y vuelve a estar disponible para su uso.
- Si la modificación de la especificación de una instancia DCS principal/en standby o de clúster falla, la instancia todavía está disponible para su uso con sus especificaciones originales. Algunas operaciones de gestión (como la configuración de parámetros, la copia de seguridad, la restauración y la modificación de las especificaciones) no se admiten temporalmente. Recuerde no leer ni escribir más datos de los permitidos por las especificaciones originales; de lo contrario, puede ocurrir la pérdida de datos.
- Después de que la modificación de especificación tiene éxito, la nueva especificación de la instancia tiene efecto.

----Fin

6.3 Inicio de una instancia

En la consola de DCS, puede iniciar una o varias instancias de DCS a la vez.

Cuando se inicia una instancia de clúster, el estado y los datos se sincronizan entre los nodos de la instancia. Si una gran cantidad de datos se escribe continuamente en la instancia antes de que se complete la sincronización, la sincronización se prolongará y la instancia permanecerá

en el estado **Starting**. Una vez completada la sincronización, la instancia entra en el estado **Running**.

NOTA

Esta función solo es compatible con instancias antiguas de DCS Redis en el estado **Stopped**. No se pueden iniciar ni detener instancias nuevas.

Procedimiento

Paso 1 Inicie sesión en la **consola DCS**.

Paso 2 Haga clic en en la esquina superior izquierda de la consola de gestión y seleccione una región.

NOTA

Seleccione la misma región que su servicio de aplicación.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Paso 4 Seleccione la instancia que desea iniciar y haga clic en **Start** encima de la lista de instancias de DCS.

Paso 5 En la caja de diálogo que aparece, haga clic en **Yes**.

- Se tarda de 1 a 30 minutos en iniciar instancias de DCS.
- Una vez iniciadas las instancias DCS, sus estados cambian de **Stopped** a **Running**.

NOTA

Para iniciar una sola instancia, haga clic en **Start** en la columna **Operation** de la fila que contiene la instancia deseada.

----Fin

6.4 Reinicio de una instancia

En la consola de DCS, puede reiniciar una o varias instancias de DCS a la vez.

ADVERTENCIA

- Después de reiniciar una instancia DCS de nodo único, los datos se eliminarán de la instancia.
- Mientras se reinicia una instancia de DCS, no se puede leer ni escribir.
- Un intento de reiniciar una instancia de DCS mientras se está haciendo una copia de seguridad puede dar lugar a un error.

Prerrequisitos:

Las instancias de DCS que desea reiniciar están en el estado **Running** o **Faulty**.

Procedimiento

Paso 1 Inicie sesión en la **consola DCS**.

Paso 2 Haga clic en en la esquina superior izquierda de la consola de gestión y seleccione la región donde se encuentra la instancia.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Paso 4 En la página **Cache Manager**, seleccione una o más instancias de DCS que desea reiniciar.

Paso 5 Haga clic en **Restart** encima de la lista de instancias de DCS.

Paso 6 En la caja de diálogo que aparece, haga clic en **Yes**.

Se tarda de 1 a 30 minutos en reiniciar instancias de DCS. Después de reiniciar las instancias DCS, su estado cambia a **Running**.

NOTA

- De forma predeterminada, solo se reiniciará el proceso de instancia. Si selecciona **Force restart** para una instancia de DCS Redis 3.0 o Memcached, su VM se reiniciará. Las instancias de DCS Redis 4.0 o posteriores no admiten **Force restart**.
- Para reiniciar una sola instancia, también puede hacer clic en **Restart** en la fila que contiene la instancia deseada.
- El tiempo necesario para reiniciar una instancia de DCS depende del tamaño de caché de la instancia.

----Fin

6.5 Eliminación de una instancia

En la consola de DCS, puede eliminar una o varias instancias de DCS a la vez. También puede eliminar todas las tareas de creación de instancias que no se hayan ejecutado.

AVISO

- Después de eliminar una instancia de DCS, los datos de la instancia se eliminarán sin copia de seguridad. Además, se eliminarán los datos de copia de seguridad de la instancia. Por lo tanto, descargue los archivos de copia de seguridad de la instancia para almacenamiento permanente antes de eliminar la instancia.
- Si la instancia está en modo de clúster, se eliminarán todos los nodos del clúster.
- Las instancias facturadas anualmente/mensualmente no se pueden eliminar.

Prerrequisitos:

- Se han creado las instancias de DCS que desea eliminar.
- Las instancias de DCS que desea eliminar están en el estado **Running**, **Faulty**, o **Stopped**.

Procedimiento

Eliminación de instancias de DCS

Paso 1 Inicie sesión en la **consola DCS**.

Paso 2 Haga clic en en la esquina superior izquierda de la consola de gestión y seleccione la región donde se encuentra la instancia.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Paso 4 En la página **Cache Manager**, seleccione una o más instancias de DCS que desea eliminar.

Las instancias de DCS en el estado **Creating**, **Starting**, **Stopping**, o **Restarting** no se pueden eliminar.

Paso 5 Elija **More** > **Delete** encima de la lista de instancias.

Paso 6 Escriba **DELETE** y haga clic en **Yes** para eliminar la instancia de DCS.

Se tarda de 1 a 30 minutos en eliminar instancias de DCS.

NOTA

Para eliminar una sola instancia, elija **More** > **Delete** en la columna **Operation** de la fila que contiene la instancia.

----Fin

Supresión de tareas de creación de instancia que no se han ejecutado

Paso 1 Inicie sesión en la **consola DCS**.

Paso 2 Haga clic en en la esquina superior izquierda de la consola de gestión y seleccione una región y un proyecto.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Si hay instancias DCS que no se han creado, **Instance Creation Failures** y el número de instancias que no se han creado se muestran encima de la lista de instancias.

Paso 4 Haga clic en el icono o el número de tareas fallidas junto a **Instance Creation Failures**.

Aparece el cuadro de diálogo **Instance Creation Failures**.

Paso 5 Eliminar las tareas de creación de instancias con errores según sea necesario.

- Para eliminar todas las tareas fallidas, haga clic en **Delete All** encima de la lista de tareas.
- Para eliminar una sola tarea fallida, haga clic en **Delete** en la fila que contiene la tarea.

----Fin

6.6 Realización de una conmutación principal/en standby

En la consola DCS, puede cambiar manualmente los nodos principal y en standby de una instancia DCS principal/en standby. Esta operación se utiliza para fines especiales, por ejemplo, liberar todas las conexiones de servicio o terminar las operaciones de servicio en curso.

Para realizar una conmutación manual para una instancia Clúster Proxy o Clúster Redis DCS Redis 4.0 o 5.0, vaya a la página **Shards and Replicas** de la instancia. Para más detalles, consulte **Gestión de fragmentos y réplicas**.

AVISO

- Durante una conmutación principal/en standby, los servicios se interrumpirán durante un máximo de 10 segundos. Antes de realizar esta operación, asegúrese de que su aplicación admite el restablecimiento de la conexión en caso de desconexión.
- Durante una conmutación de nodo principal/en standby, se consumirá una gran cantidad de recursos para la sincronización de datos entre los nodos principal y en standby. Se aconseja realizar la operación durante las horas de menor actividad.
- Los datos de los nodos principal y en standby se sincronizan asincrónicamente. Por lo tanto, se puede perder una pequeña cantidad de datos que se están operando durante la conmutación.

Prerrequisitos:

La instancia de DCS para la que desea realizar una conmutación principal/en standby está en el estado **Running**.

Procedimiento

Paso 1 Inicie sesión en la **consola DCS**.

Paso 2 Haga clic en en la esquina superior izquierda de la consola de gestión y seleccione la región donde se encuentra la instancia.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Paso 4 En la columna **Operation** de la instancia, elija **More** > **Master/Standby Switchover**.

----Fin

6.7 Borrado de datos de la instancia de DCS

En la consola de DCS, solo puede borrar datos para instancias de DCS Redis 4.0/5.0. No se puede deshacer el borrado de datos de instancia y no se pueden recuperar los datos borrados. Tenga cuidado cuando realice esta acción.

Prerrequisitos:

La instancia de DCS Redis 4.0/5.0 está en el estado **Running**.

Procedimiento

Paso 1 Inicie sesión en la **consola DCS**.

Paso 2 Haga clic en en la esquina superior izquierda de la consola de gestión y seleccione la región donde se encuentra la instancia.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Paso 4 Seleccione una o más instancias de DCS.

Paso 5 Elija **More** > **Clear** encima de la lista de instancias.

Paso 6 En la caja de diálogo que aparece, haga clic en **Yes**.

----Fin

6.8 Exportación de lista de instancias

En la consola de DCS, puede exportar la información completa de la instancia de DCS a un archivo de Excel.

Procedimiento

Paso 1 Inicie sesión en la **consola DCS**.

Paso 2 Haga clic en en la esquina superior izquierda de la consola de gestión y seleccione la región donde se encuentra la instancia.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Paso 4 Haga clic en sobre la lista de instancias.

Haga clic en el resultado de exportación que se muestra en la esquina inferior izquierda de la página. **Figura 6-1** muestra un resultado de ejemplo.

Figura 6-1 Lista de instancias de DCS exportadas

Name	ID	Status	AZ	Cache Eng	Instance	Specifica	Used/Avai	Connecti	Created	Billing M	VPC	VPC ID	Enterprise Project
dcs-trpt	5e4f4c58-	Running	AZ1	Redis 5.0	Single-no	0.125	0/128 (0%	198.19.32	May 24, 2	Free	null	null	default
dcs-APITe	693491b0-	Running		Redis 3.0	Master/St	2	2/1,536	172.16.14	May 06, 2	Yearly/Mo	null	52267da0-	default

----Fin

6.9 Comandos del cambio de nombre

Después de crear una instancia de DCS Redis 4.0/5.0, puede cambiar el nombre de los siguientes comandos críticos: Actualmente, solo puede cambiar el nombre de los comandos COMMAND, KEYS, FLUSHDB, FLUSHALL y HGETALL.

Procedimiento

Paso 1 Inicie sesión en la **consola DCS**.

Paso 2 Haga clic en en la esquina superior izquierda de la consola de gestión y seleccione la región donde se encuentra la instancia.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Paso 4 En la columna **Operation** de una instancia, elija **More** > **Command Renaming**.

Paso 5 Seleccione un comando, escriba un nuevo nombre y haga clic en **OK**.

NOTA

- Puede cambiar el nombre de varios comandos a la vez.
- Los nuevos nombres de comandos surtirán efecto solo después de reiniciar la instancia. Recuerde los nuevos nombres de comandos porque no se mostrarán en la consola por motivos de seguridad.

----Fin

7 Gestión de instancias de DCS

7.1 Aviso de configuración

En la mayoría de los casos, las diferentes operaciones de gestión de instancias de DCS no pueden proceder simultáneamente. Si inicia una nueva operación de gestión mientras la operación actual está en curso, la consola de DCS le solicitará que vuelva a iniciar la nueva operación una vez completada la operación actual. Las operaciones de gestión de instancias de DCS incluyen:

- Crear instancia de DCS
- Configuración de parámetros
- Reinicio de una instancia de DCS
- Cambio de la contraseña de instancia
- Restablecimiento de la contraseña de instancia
- Escalado, copia de seguridad o restauración de una instancia

Puede reiniciar una instancia de DCS mientras se realiza la copia de seguridad, pero la tarea de copia de seguridad se interrumpirá a la fuerza y es probable que resulte en un error de copia de seguridad.

AVISO

En caso de que un nodo de caché de una instancia DCS esté defectuoso:

- La instancia permanece en el estado **Running** y puede seguir leyendo y escribiendo en la instancia. Esto se consigue gracias a la alta disponibilidad de DCS.
- Los nodos de caché pueden recuperarse de fallos internos automáticamente. También se admite la recuperación manual de fallas.
- Ciertas operaciones (como copia de seguridad, restauración y configuración de parámetros) en la zona de gestión no se admiten durante la recuperación de fallos. Puede ponerse en contacto con el servicio de atención al cliente o realizar estas operaciones después de que los nodos de caché se recuperen de errores.

7.2 Modificación de parámetros de configuración

7.2.1 Modificación de parámetros de configuración de una instancia

En la consola de DCS, puede configurar parámetros para una instancia para lograr un rendimiento de DCS óptimo.

Por ejemplo, si no necesita persistencia de datos, establezca **appendonly** en **no**.

Después de modificar los parámetros de configuración de la instancia, la modificación tiene efecto inmediatamente sin necesidad de reiniciar manualmente la instancia. Para una instancia de clúster, la modificación tiene efecto en todos los fragmentos.

Procedimiento

Paso 1 Inicie sesión en la **consola DCS**..

Paso 2 Haga clic en en la esquina superior izquierda de la consola de gestión y seleccione la región donde se encuentra la instancia.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Paso 4 En la página **Cache Manager**, haga clic en el nombre de la instancia de DCS que desea configurar.

Paso 5 En la página de detalles de la instancia, seleccione **Instance Configuration** > **Parameters**.

Paso 6 Haga clic en **Modify**.

Paso 7 Modifique los parámetros según sus requisitos.

Tabla 7-1 y **Tabla 7-2** describen los parámetros. En la mayoría de los casos, se conservan los valores predeterminados.

Tabla 7-1 Parámetros de configuración de instancia de DCS Redis

Parámetro	Descripción	Rango de valores	Valor predeterminado
timeout	La cantidad máxima de tiempo (en segundos) que se puede permitir que una conexión entre un cliente y la instancia de DCS permanezca inactiva antes de que finalice la conexión. Un ajuste de **0** significa que esta función está deshabilitada.	0–7200 segundos	0

Parámetro	Descripción	Rango de valores	Valor predeterminado
appendfsync	Controla la frecuencia con la que fsync () transfiere datos almacenados en la memoria caché al disco. Tenga en cuenta que algunos SO realizarán una transferencia de datos completa, pero otros solo hacen un intento de "mejor esfuerzo". Hay tres configuraciones: no: nunca se llama a fsync(). El SO descargará los datos cuando esté listo. Este modo ofrece el máximo rendimiento. always: fsync() se llama después de cada escritura en el AOF. Este modo es muy lento, pero también muy seguro. everysec: se llama a fsync() una vez por segundo. Este modo proporciona un compromiso entre seguridad y rendimiento.	● no ● always ● everysec	everysec
appendonly	Indica si se deben registrar todas las modificaciones de la instancia. Por defecto, los datos se escriben en discos de forma asíncrona en Redis. Si esta función está deshabilitada, los datos generados recientemente podrían perderse en el caso de un corte de energía. Opciones: yes: Los registros están habilitados, es decir, la persistencia está habilitada. no: Los registros están deshabilitados, es decir, la persistencia está deshabilitada.	● yes ● no	yes

Parámetro	Descripción	Rango de valores	Valor predeterminado
client-output-buffer-limit-slave-soft-seconds	Número de segundos que el búfer de salida permanece por encima de **client-output-buffer-slave-soft-limit** antes de que el cliente se desconecte.	0–60	60
client-output-buffer-slave-hard-limit	Límite invariable (en bytes) en el búfer de salida de los clientes de réplica. Una vez que el búfer de salida excede el límite invariable, el cliente se desconecta inmediatamente.	0–17,179,869,184	1,717,986,918
client-output-buffer-slave-soft-limit	Límite flexible (en bytes) en el búfer de salida de los clientes de réplica. Una vez que el búfer de salida excede el límite flexible y permanece continuamente por encima del límite durante el período especificado por el parámetro **client-output-buffer-limit-slave-soft-seconds**, el cliente se desconecta.	0–17,179,869,184	1,717,986,918

Parámetro	Descripción	Rango de valores	Valor predeterminado
maxmemory-policy	La política aplicada cuando se alcanza el límite maxmemory. Para obtener más información acerca de este parámetro, vea **https://docs.redis.com/latest/rs/databases/memory-performance/eviction-policy/**.	● volatile-lru ● allkeys-lru ● volatile-random ● allkeys-random ● volatile-ttl ● noeviction	volatile-lru **NOTA** Si la instancia de DCS Redis se creó antes de julio de 2020 y este parámetro no se ha modificado, el valor predeterminado es **noeviction**. Si la instancia se crea después de julio de 2020, el valor predeterminado es **volatile-lru**.
lua-time-limit	Tiempo máximo permitido para la ejecución de un script Lua (en milisegundos)	100–5000	5000
master-read-only	Configura la instancia como de solo lectura. No se podrá ejecutar ninguna operación de escritura.	● yes ● no	no
maxclients	Cantidad máxima de clientes que se puede conectar de forma simultánea a una instancia de DCS.	1000–50,000	10,000
proto-max-bulk-len	Tamaño máximo de una solicitud de elemento único (en bytes).	1,048,576–536,870,912	536,870,912

Parámetro	Descripción	Rango de valores	Valor predeterminado
repl-backlog-size	Tamaño del backlog de replicación (bytes). El backlog es un búfer que acumula datos de réplicas cuando estas se desconectan de la instancia principal. Cuando una réplica se vuelve a conectar, se realiza una sincronización parcial para sincronizar los datos que se perdieron mientras las réplicas estuvieron desconectadas.	16,384–1,073,741,824	1,048,576
repl-backlog-ttl	La cantidad de tiempo, en segundos, antes de que se libere el búfer de backlog, calculada a partir de la última que se desconectó una réplica. El valor **0** indica que el backlog nunca se libera.	0–604,800	3600
repl-timeout	Fin de tiempo de espera de la replicación (en segundos).	30–3600	60
hash-max-ziplist-entries	Número máximo de hashes que se pueden codificar mediante ziplist, una estructura de datos optimizada para reducir el uso de memoria.	1–10,000	512
hash-max-ziplist-value	El valor más grande permitido para un hash codificado usando ziplist, una estructura de datos especial optimizada para el uso de memoria.	1–10,000	64

Parámetro	Descripción	Rango de valores	Valor predeterminado
set-max-intset-entries	Si un conjunto se compone únicamente de cadenas de caracteres que son números enteros en base 10 dentro del rango de números enteros con signo de 64 bits, el conjunto se codifica usando intset, una estructura de datos optimizada para el uso de memoria.	1–10,000	512
zset-max-ziplist-entries	Número máximo de conjuntos ordenados que se pueden codificar mediante ziplist, una estructura de datos optimizada para reducir el uso de memoria.	1–10,000	128
zset-max-ziplist-value	El valor más grande permitido para un conjunto ordenado codificado usando ziplist, una estructura de datos especial optimizada para el uso de memoria.	1–10,000	64

Parámetro	Descripción	Rango de valores	Valor predeterminado
latency-monitor-threshold	La cantidad mínima de latencia que se registrará como picos de latencia ● Establecer en 0: La supervisión de la latencia está deshabilitada. ● Establecer a más de 0: Todos con al menos este número de ms de latencia se registrarán. Al ejecutar el comando LATENCY, puede realizar operaciones relacionadas con el monitoreo de latencia, como la obtención de datos estadísticos y la configuración y habilitación del monitoreo de latencia. Para obtener más información acerca de la latencia-monitor-threshold, visite **https://redis.io/docs/reference/optimization/latency-monitor/**.	0–86,400,000 ms	0

Parámetro	Descripción	Rango de valores	Valor predeterminado
notify-keyspace-events	Controla qué tipo de notificaciones están habilitadas para los eventos de espacio de claves. Si se configura este parámetro, la función Redis Pub/Sub permitirá a los clientes recibir una notificación de evento cuando se modifique un conjunto de datos Redis. Las instancias de Clúster Proxy no tienen este parámetro.	Se puede usar una combinación de diferentes valores para habilitar notificaciones para varios tipos de eventos. Los valores posibles incluyen: K: Eventos de Keyspace, publicados con el __keyspace@__ prefix E: Eventos de Keyevent, publicados con __keyevent@__ prefix Comandos genéricos (sin un tipo específico) como DEL, EXPIRE y RENAME: $: Comandos de cadena l: Lista de comandos s: Establecer comandos h: Comandos hash z: Comandos de conjunto ordenado x: Eventos expirados (eventos generados cada vez que expira una clave) e: Eventos desalojados (eventos generados cuando una clave es desalojada de maxmemory) Para obtener más información, consulte la siguiente nota.	Ex
slowlog-log-slower-than	La cantidad máxima de tiempo permitido, en microsegundos, para la ejecución de comandos. Si se supera este umbral, el registro de consultas lentas de Redis registrará el comando.	0–1,000,000	10,000

Parámetro	Descripción	Rango de valores	Valor predeterminado
slowlog-max-len	Número máximo permitido de consultas lentas que se pueden registrar. El registro de consultas lento consume memoria, pero puede recuperar esta memoria ejecutando el comando **SLOWLOG RESET**.	0–1000	128

NOTA

1. Para obtener más información acerca de los parámetros descritos en **Tabla 7-1**, visite **https://redis.io/topics/memory-optimization**.
2. El parámetro **latency-monitor-threshold** se utiliza normalmente para la localización de fallos. Después de localizar fallos basados en la información de latencia recopilada, cambie el valor de **latency-monitor-threshold** a **0** para evitar latencia innecesaria.
3. Más información sobre el parámetro **notify-keyspace-events**:
   - La configuración del parámetro debe contener al menos una K o una E.
   - A es un alias para "g$lshzxe" y no se puede usar junto con ninguno de los caracteres en "g $lshzxe".
   - Por ejemplo, el valor **KI** significa que Redis notificará a los clientes de Pub/Sub acerca de los eventos de espacio de claves y los comandos de lista. El valor **AKE** significa que Redis notificará a los clientes Pub/Sub sobre todos los eventos.
4. Los parámetros configurables varían según el tipo de instancia.

Tabla 7-2 Parámetros de configuración de instancia de DCS Memcached

Parámetro	Descripción	Rango de valores	Default Value
timeout	La cantidad máxima de tiempo (en segundos) que se puede permitir que una conexión entre un cliente y la instancia de DCS permanezca inactiva antes de que finalice la conexión. Un ajuste de **0** significa que esta función está deshabilitada.	0–7200 segundos	0
maxclients	Cantidad máxima de clientes que se puede conectar de forma simultánea a una instancia de DCS.	1000–10,000	10,000

Parámetro	Descripción	Rango de valores	Default Value
maxmemory-policy	La política aplicada cuando se alcanza el límite maxmemory. Para obtener más información acerca de este parámetro, vea **https://docs.redis.com/latest/rs/databases/memory-performance/eviction-policy/**.	volatile-lru allkeys-lru volatile-random allkeys-random volatile-ttl noeviction	noeviction
reserved-memory-percent	Porcentaje de la memoria máxima disponible que está reservada para procesos en segundo plano, como la persistencia y la replicación de datos.	0–80	30

Paso 8 Una vez que haya terminado de configurar los parámetros, haga clic en **Save**.

Paso 9 Haga clic en **Yes** para confirmar la modificación.

----Fin

7.2.2 Modificación de parámetros de configuración en lotes

En la consola DCS, puede configurar varios parámetros a la vez para que una instancia logre un rendimiento de DCS óptimo.

Después de modificar los parámetros de configuración de la instancia, la modificación tiene efecto inmediatamente sin necesidad de reiniciar manualmente la instancia. Para una instancia de clúster, la modificación tiene efecto en todos los fragmentos.

Procedimiento

Paso 1 Inicie sesión en la **consola DCS**..

Paso 2 Haga clic en  en la esquina superior izquierda de la consola de gestión y seleccione la región donde se encuentra la instancia.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Paso 4 En la página **Cache Manager**, seleccione todas las instancias de DCS que desee configurar.

Paso 5 Elija **More** > **Modify Parameters**.

NOTA

- Los parámetros mostrados en la página **Modify Parameters** son el conjunto de unión de los parámetros de las instancias seleccionadas. Por ejemplo, supongamos que el parámetro **appendfsync** de la instancia 1 no es compatible con la instancia 2. Todavía puede seleccionar este parámetro, pero el sistema muestra un mensaje en **Paso 7** indicando que la instancia 2 no admite este parámetro. Después del envío, el comando de modificación no se entregará a la instancia 2. El rango de valores de parámetro es la intersección de los rangos de valores de parámetro de las instancias seleccionadas. Por ejemplo, si el rango de valores es de 0 a 50,000, por ejemplo 1, y de 1000 a 50,000, por ejemplo 2, el rango de valores que se muestra en la página es de 1000 a 50,000.
- Si no desea modificar una instancia seleccionada, haga clic en  junto a ella.

Paso 6 Seleccione el parámetro que se va a modificar e introduzca un nuevo valor en la columna **New Value**.

Tabla 7-3 y **Tabla 7-4** describen los parámetros. En la mayoría de los casos, puede conservar los valores predeterminados.

Tabla 7-3 Parámetros de configuración de instancia de DCS Redis

Parámetro	Descripción	Rango de valores	Valor predeterminado
timeout	La cantidad máxima de tiempo (en segundos) que se puede permitir que una conexión entre un cliente y la instancia de DCS permanezca inactiva antes de que finalice la conexión. Un ajuste de **0** significa que esta función está deshabilitada.	0–7200 segundos	0

Parámetro	Descripción	Rango de valores	Valor predeterminado
appendfsync	Controla la frecuencia con la que fsync () transfiere datos almacenados en la memoria caché al disco. Tenga en cuenta que algunos SO realizarán una transferencia de datos completa, pero otros solo hacen un intento de "mejor esfuerzo". Hay tres configuraciones: no: nunca se llama a fsync(). El SO descargará los datos cuando esté listo. Este modo ofrece el máximo rendimiento. always: fsync() se llama después de cada escritura en el AOF. Este modo es muy lento, pero también muy seguro. everysec: se llama a fsync() una vez por segundo. Este modo proporciona un compromiso entre seguridad y rendimiento.	● no ● always ● everysec	everysec
appendonly	Indica si se deben registrar todas las modificaciones de la instancia. Por defecto, los datos se escriben en discos de forma asíncrona en Redis. Si esta función está deshabilitada, los datos generados recientemente podrían perderse en el caso de un corte de energía. Opciones: yes: Los registros están habilitados, es decir, la persistencia está habilitada. no: Los registros están deshabilitados, es decir, la persistencia está deshabilitada.	● yes ● no	yes

Parámetro	Descripción	Rango de valores	Valor predeterminado
client-output-buffer-limit-slave-soft-seconds	Número de segundos que el búfer de salida permanece por encima de **client-output-buffer-slave-soft-limit** antes de que el cliente se desconecte.	0–60	60
client-output-buffer-slave-hard-limit	Límite invariable (en bytes) en el búfer de salida de los clientes de réplica. Una vez que el búfer de salida excede el límite invariable, el cliente se desconecta inmediatamente.	0–17,179,869,184	1,717,986,918
client-output-buffer-slave-soft-limit	Límite flexible (en bytes) en el búfer de salida de los clientes de réplica. Una vez que el búfer de salida excede el límite flexible y permanece continuamente por encima del límite durante el período especificado por el parámetro **client-output-buffer-limit-slave-soft-seconds**, el cliente se desconecta.	0–17,179,869,184	1,717,986,918

Parámetro	Descripción	Rango de valores	Valor predeterminado
maxmemory-policy	La política aplicada cuando se alcanza el límite maxmemory. Para obtener más información acerca de este parámetro, vea **https://docs.redis.com/latest/rs/databases/memory-performance/eviction-policy/**.	● volatile-lru ● allkeys-lru ● volatile-random ● allkeys-random ● volatile-ttl ● noeviction	volatile-lru **NOTA** Si la instancia de DCS Redis se creó antes de julio de 2020 y este parámetro no se ha modificado, el valor predeterminado es **noeviction**. Si la instancia se crea después de julio de 2020, el valor predeterminado es **volatile-lru**.
lua-time-limit	Tiempo máximo permitido para la ejecución de un script Lua (en milisegundos)	100–5000	5000
master-read-only	Configura la instancia como de solo lectura. No se podrá ejecutar ninguna operación de escritura.	● yes ● no	no
maxclients	Cantidad máxima de clientes que se puede conectar de forma simultánea a una instancia de DCS.	1000–50,000	10,000
proto-max-bulk-len	Tamaño máximo de una solicitud de elemento único (en bytes).	1,048,576–536,870,912	536,870,912

Parámetro	Descripción	Rango de valores	Valor predeterminado
repl-backlog-size	Tamaño del backlog de replicación (bytes). El backlog es un búfer que acumula datos de réplicas cuando estas se desconectan de la instancia principal. Cuando una réplica se vuelve a conectar, se realiza una sincronización parcial para sincronizar los datos que se perdieron mientras las réplicas estuvieron desconectadas.	16,384–1,073,741,824	1,048,576
repl-backlog-ttl	La cantidad de tiempo, en segundos, antes de que se libere el búfer de backlog, calculada a partir de la última que se desconectó una réplica. El valor **0** indica que el backlog nunca se libera.	0–604,800	3600
repl-timeout	Fin de tiempo de espera de la replicación (en segundos).	30–3600	60
hash-max-ziplist-entries	Número máximo de hashes que se pueden codificar mediante ziplist, una estructura de datos optimizada para reducir el uso de memoria.	1–10,000	512
hash-max-ziplist-value	El valor más grande permitido para un hash codificado usando ziplist, una estructura de datos especial optimizada para el uso de memoria.	1–10,000	64

Parámetro	Descripción	Rango de valores	Valor predeterminado
set-max-intset-entries	Si un conjunto se compone únicamente de cadenas de caracteres que son números enteros en base 10 dentro del rango de números enteros con signo de 64 bits, el conjunto se codifica usando intset, una estructura de datos optimizada para el uso de memoria.	1–10,000	512
zset-max-ziplist-entries	Número máximo de conjuntos ordenados que se pueden codificar mediante ziplist, una estructura de datos optimizada para reducir el uso de memoria.	1–10,000	128
zset-max-ziplist-value	El valor más grande permitido para un conjunto ordenado codificado usando ziplist, una estructura de datos especial optimizada para el uso de memoria.	1–10,000	64

Parámetro	Descripción	Rango de valores	Valor predeterminado
latency-monitor-threshold	La cantidad mínima de latencia que se registrará como picos de latencia ● Establecer en 0: La supervisión de la latencia está deshabilitada. ● Establecer a más de 0: Todos con al menos este número de ms de latencia se registrarán. Al ejecutar el comando LATENCY, puede realizar operaciones relacionadas con el monitoreo de latencia, como la obtención de datos estadísticos y la configuración y habilitación del monitoreo de latencia. Para obtener más información acerca de la latencia-monitor-threshold, visite **https://redis.io/docs/reference/optimization/latency-monitor/**.	0–86,400,000 ms	0

Parámetro	Descripción	Rango de valores	Valor predeterminado
notify-keyspace-events	Controla qué tipo de notificaciones están habilitadas para los eventos de espacio de claves. Si se configura este parámetro, la función Redis Pub/Sub permitirá a los clientes recibir una notificación de evento cuando se modifique un conjunto de datos Redis. Las instancias de Clúster Proxy no tienen este parámetro.	Se puede usar una combinación de diferentes valores para habilitar notificaciones para varios tipos de eventos. Los valores posibles incluyen: K: Eventos de Keyspace, publicados con el __keyspace@__ prefix E: Eventos de Keyevent, publicados con __keyevent@__ prefix Comandos genéricos (sin un tipo específico) como DEL, EXPIRE y RENAME: $: Comandos de cadena l: Lista de comandos s: Establecer comandos h: Comandos hash z: Comandos de conjunto ordenado x: Eventos expirados (eventos generados cada vez que expira una clave) e: Eventos desalojados (eventos generados cuando una clave es desalojada de maxmemory) Para obtener más información, consulte la siguiente nota.	Ex
slowlog-log-slower-than	La cantidad máxima de tiempo permitido, en microsegundos, para la ejecución de comandos. Si se supera este umbral, el registro de consultas lentas de Redis registrará el comando.	0–1,000,000	10,000

Parámetro	Descripción	Rango de valores	Valor predeterminado
slowlog-max-len	Número máximo permitido de consultas lentas que se pueden registrar. El registro de consultas lento consume memoria, pero puede recuperar esta memoria ejecutando el comando **SLOWLOG RESET**.	0–1000	128

NOTA

1. Para obtener más información sobre los parámetros descritos en **Tabla 7-3**, visite el **sitio web oficial de Redis**.
2. El parámetro **latency-monitor-threshold** se utiliza normalmente para la localización de fallos. Después de localizar fallos basados en la información de latencia recopilada, cambie el valor de **latency-monitor-threshold** a **0** para evitar latencia innecesaria.
3. Más información sobre el parámetro **notify-keyspace-events**:
   - La configuración del parámetro debe contener al menos una K o una E.
   - A es un alias para "g$lshzxe" y no se puede usar junto con ninguno de los caracteres en "g $lshzxe".
   - Por ejemplo, el valor **Kl** significa que Redis notificará a los clientes de Pub/Sub acerca de los eventos de espacio de claves y los comandos de lista. El valor **AKE** significa que Redis notificará a los clientes Pub/Sub sobre todos los eventos.

Tabla 7-4 Parámetros de configuración de instancia de DCS Memcached

Parámetro	Descripción	Rango de valores	Default Value
timeout	La cantidad máxima de tiempo (en segundos) que se puede permitir que una conexión entre un cliente y la instancia de DCS permanezca inactiva antes de que finalice la conexión. Un ajuste de **0** significa que esta función está deshabilitada.	0–7200 segundos	0
maxclients	Cantidad máxima de clientes que se puede conectar de forma simultánea a una instancia de DCS.	1000–10,000	10,000

Parámetro	Descripción	Rango de valores	Default Value
maxmemory-policy	La política aplicada cuando se alcanza el límite maxmemory. Para obtener más información acerca de este parámetro, vea **https://docs.redis.com/latest/rs/databases/memory-performance/eviction-policy/**.	volatile-lru allkeys-lru volatile-random allkeys-random volatile-ttl noeviction	noeviction
reserved-memory-percent	Porcentaje de la memoria máxima disponible que está reservada para procesos en segundo plano, como la persistencia y la replicación de datos.	0–80	30

Paso 7 Haga clic en **Next: Confirm Parameters** para confirmar las instancias y los valores de los parámetros.

NOTA

- En la página de confirmación, puede filtrar las instancias seleccionadas en **Paso 4** por motor de caché, tipo de instancia y estado, y modificar el valor del parámetro de estas instancias.
- Puede establecer valores diferentes para diferentes instancias dentro del rango de valores. Si el valor actual y el nuevo valor de una instancia son iguales, no se generará ningún registro de modificación para la instancia.

Paso 8 Haga clic en **Submit**.

Paso 9 Haga clic en una instancia de DCS. En la página de detalles de la instancia que se muestra, seleccione **Parameters** > **Modification History** para comprobar si el parámetro de configuración se ha modificado correctamente.

----Fin

7.3 Modificación de ventana de Mantenimiento

En la consola DCS, después de crear una instancia DCS, puede modificar la ventana de mantenimiento de la instancia DCS en la página **Basic Information** de la instancia. Durante la ventana de mantenimiento, el personal de O&M puede mantener la instancia.

Prerrequisitos:

Se ha creado una instancia de DCS.

Procedimiento

Paso 1 Inicie sesión en la **consola DCS**..

Paso 2 Haga clic en en la esquina superior izquierda de la consola de gestión y seleccione la región donde se encuentra la instancia.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Paso 4 Haga clic en el nombre de la instancia de DCS deseada.

Paso 5 Haga clic en la ficha **Basic Information**. En el área **Instance Details**, haga clic en el icono  situado junto al parámetro **Maintenance**.

Paso 6 Seleccione una nueva ventana de mantenimiento en la lista desplegable. Haga clic en  para guardar la modificación o en  para descartarla.

La modificación entrará en vigor inmediatamente en la página de ficha **Basic Information**.

----Fin

7.4 Modificación del grupo de seguridad

En la consola DCS, después de crear una instancia DCS, puede modificar el grupo de seguridad de la instancia DCS en la página **Basic Information** de la instancia.

Puede modificar los grupos de seguridad de las instancias de DCS Redis 3.0 pero no puede modificar los de las instancias de DCS Redis 4.0/5.0.

NOTA

DCS for Redis 3.0 ya no se proporciona. Puede utilizar DCS for Redis 4.0 o 5.0 en su lugar.

Prerrequisitos:

Se ha creado una instancia de DCS.

Procedimiento

Paso 1 Inicie sesión en la **consola DCS**.

Paso 2 Haga clic en  en la esquina superior izquierda de la consola de gestión y seleccione la región donde se encuentra la instancia.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Paso 4 Haga clic en el nombre de la instancia de DCS deseada.

Paso 5 Haga clic en la ficha **Basic Information**. En el área **Network**, haga clic en  junto al parámetro **Security Group**.

Paso 6 Seleccione un nuevo grupo de seguridad en la lista desplegable. Haga clic en  para guardar la modificación o en para descartarla.

NOTA

Sólo se pueden seleccionar los grupos de seguridad creados en la lista desplegable. Si necesita crear un grupo de seguridad, siga el procedimiento descrito en **¿Cómo configuro un grupo de seguridad?**

La modificación entrará en vigor inmediatamente en la página de ficha **Basic Information**.

----Fin

7.5 Consulta de tareas del fondo

Después de iniciar determinadas operaciones de instancia, como escalar la instancia y cambiar o restablecer una contraseña, se iniciará una tarea en segundo plano para cada operación. En la consola de DCS, puede ver el estado de la tarea del fondo y borrar la información de la tarea eliminando registros de tareas.

Procedimiento

Paso 1 Inicie sesión en la **consola DCS**.

Paso 2 Haga clic en en la esquina superior izquierda de la consola de gestión y seleccione la región donde se encuentra la instancia.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Filtrar las instancias de DCS para buscar la instancia DCS deseada. Actualmente, puede buscar instancias por nombre, especificación, ID, dirección IP, AZ, estado, tipo de instancia, motor de caché y muchos otros atributos.

Paso 4 Haga clic en el nombre de la instancia de DCS para mostrar más detalles sobre la instancia de DCS.

Paso 5 Haga clic en la ficha **Background Tasks**.

Se muestra una lista de tareas del fondo.

Paso 6 Haga clic en , especifique **Start Date** y **End Date** y haga clic en **OK** para ver las tareas iniciadas en el segmento de tiempo correspondiente.

- Haga clic en para actualizar el estado de la tarea.
- Para borrar el registro de una tarea en segundo plano, elija **Operation** > **Delete**.

  NOTA

  Sólo puede eliminar los registros de las tareas en el estado **Successful** o **Failed**.

----Fin

7.6 Gestión de la lista blanca de direcciones IP

Las instancias DCS Redis 3.0/4.0/5.0 y Memcached se implementan en diferentes modos. Por lo tanto, el método de control de acceso varía.

- Para controlar el acceso a las instancias de la edición profesional de DCS Redis 3.0, Memcached y Redis 6.0, puede utilizar grupos de seguridad. Las listas blancas no son compatibles. Para obtener más información sobre cómo configurar un grupo de seguridad, consulte **¿Cómo configuro un grupo de seguridad?**
- Para controlar el acceso a las instancias de DCS Redis 4.0/5.0, puede utilizar listas blancas. No se admiten grupos de seguridad.

A continuación se describe cómo gestionar listas blancas de una instancia de Redis 4.0/5.0 para permitir el acceso solo desde direcciones IP de la lista blanca. Si no se agregan listas

blancas para la instancia o la función de lista blanca está deshabilitada, todas las direcciones IP que pueden comunicarse con la VPC pueden acceder a la instancia.

Creación de un grupo de listas blancas

Paso 1 Inicie sesión en la **consola DCS**.

Paso 2 Haga clic en en la esquina superior izquierda de la consola de gestión y seleccione una región.

NOTA

Seleccione la misma región que su servicio de aplicación.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Paso 4 Haga clic en el nombre de una instancia de DCS.

Paso 5 Seleccione **Instance Configuration** > **Whitelist**. En la página mostrada, haga clic en **Create Whitelist Group**.

Paso 6 En el cuadro de diálogo **Create Whitelist Group**, especifique **Group Name** y **IP Address/Range**.

Tabla 7-5 Parámetros de la lista blanca

Parámetro	Descripción	Ejemplo
Group Name	Nombre del grupo de la lista blanca de la instancia. Se puede crear un máximo de cuatro grupos de listas blancas para cada instancia.	Prueba DCS
IP Address/Range	Se puede agregar un máximo de 20 direcciones IP o intervalos de direcciones IP a una instancia. Separe las múltiples direcciones IP o los rangos de direcciones IP con comas. Direcciones IP y rango de direcciones IP no compatibles: 0.0.0.0 y 0.0.0/0.	10.10.10.1,10.10.10.10

Paso 7 Haz clic en **OK**.

Un grupo de lista blanca se habilita automáticamente para la instancia una vez creada. Solo las direcciones IP que se encuentren dentro de la lista blanca pueden acceder a la instancia.

NOTA

- En la lista de grupos de lista blanca, haga clic en **Edit** para modificar las direcciones IP o los intervalos de direcciones IP de un grupo y haga clic en **Delete** para eliminar un grupo de lista blanca.
- Después de habilitar la lista blanca, puede hacer clic en **Disable Whitelist** encima de la lista de grupos de listas blancas para permitir que todas las direcciones IP conectadas a la VPC accedan a la instancia.

----Fin

7.7 Gestión de etiquetas

Las etiquetas facilitan la identificación y gestión de instancias de DCS.

Puede agregar etiquetas a una instancia al crearla o agregar, modificar o eliminar etiquetas en la página de detalles de una instancia creada. Cada instancia puede tener un máximo de 20 etiquetas.

Una etiqueta consiste en una clave de etiqueta y un valor de etiqueta. **Tabla 7-6** enumera los requisitos de valor y clave de etiqueta.

Tabla 7-6 Clave de etiquetas y requisitos de valor

Parámetro	Requerimientos
Clave de la etiqueta	● No se puede dejar en blanco. ● Debe ser único para la misma instancia. ● Consta de un máximo de 128 caracteres. ● Puede contener letras de cualquier idioma, dígitos, espacios y caracteres especiales _.: = + - @ ● No puede comenzar ni finalizar con un espacio. ● No se puede iniciar con **_sys_**.
Valor de la etiqueta	● Consta de un máximo de 255 caracteres. ● Puede contener letras de cualquier idioma, dígitos, espacios y caracteres especiales _.: / = + - @ ● No puede comenzar ni finalizar con un espacio.

Procedimiento

Paso 1 Inicie sesión en la **consola DCS**.

Paso 2 Haga clic en en la esquina superior izquierda de la consola de gestión y seleccione una región.

NOTA

Seleccione la misma región que su servicio de aplicación.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Paso 4 Haga clic en el nombre de la instancia de DCS deseada para ir a la página de detalles.

Paso 5 Seleccione **Instance Configuration** > **Tags**.

Paso 6 Realice las siguientes operaciones según sea necesario:

- Agregue una etiqueta
  a. Haga clic en **Add/Edit Tag**.
     Si ha creado etiquetas predefinidas, seleccione un par predefinido de clave y valor de etiqueta. Para ver o crear etiquetas predefinidas, haga clic en **View predefined tags**. A continuación, se le dirigirá a la consola TMS.
     También puede crear nuevas etiquetas especificando **Tag key** y **Tag value**.
  b. Haz clic en **OK**.
- Modificar una etiqueta
  Haga clic en **Add/Edit Tag**. En el cuadro de diálogo **Add/Edit Tag** etiqueta que se muestra, elimine la clave deseada, vuelva a agregarla, introduzca un nuevo valor de etiqueta y haga clic en **Add**.
- Eliminar una etiqueta
  En la fila que contiene la etiqueta deseada, haga clic en **Delete**. En la caja de diálogo que aparece, haga clic en **Yes**.

----Fin

7.8 Gestión de fragmentos y réplicas

En esta sección se describe cómo consultar las particiones y réplicas de una instancia de DCS Redis 4.0/5.0 principal/en standby, de clúster o de separación de lectura/escritura, y cómo promover manualmente una réplica al principal.

Las instancias de DCS Redis 3.0 y las de nodo único de DCS Redis 4.0/5.0/6.0 no admiten esta función.

- De forma predeterminada, una instancia principal/standby o de separación de lectura/escritura solo tiene una partición con un principal y una réplica. Puede ver la información de particiones en la página **Shards and Replicas**. Para cambiar manualmente los roles principal y réplica, consulte **Realización de una conmutación principal/en standby**.
- Una instancia Clúster Proxy o Clúster Redis tiene múltiples particiones. Cada partición tiene un principal y una réplica. En la página **Shards and Replicas**, puede ver la información de particiones y cambiar manualmente los roles principal y réplica.

  NOTA

  - Para obtener detalles sobre el número de fragmentos para diferentes especificaciones de instancia, consulte **Instancias de Clúster Redis de DCS para Redis 4.0 y 5.0** e **Instancias de Clúster Proxy de DCS para Redis 4.0 y 5.0**.
  - Puede agregar fragmentos a una instancia de clúster haciendo referencia a **Modificación de las especificaciones**.

Procedimiento

Paso 1 Inicie sesión en la **consola DCS**.

Paso 2 Haga clic en en la esquina superior izquierda de la consola de gestión y seleccione una región.

NOTA

Seleccione la misma región que su servicio de aplicación.

Paso 3 En el panel de navegación, elija **Cache Manager**.

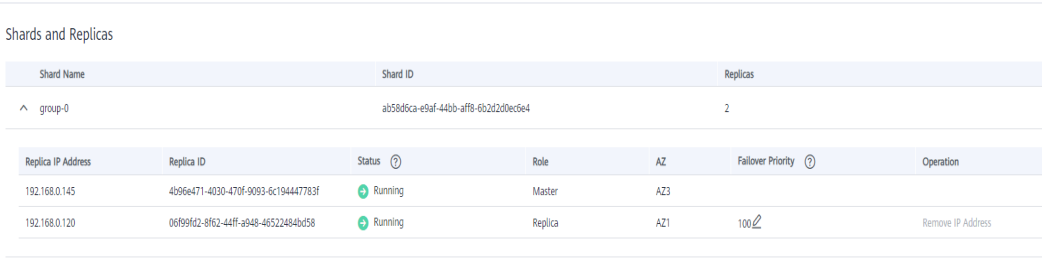
Paso 4 Haga clic en una instancia.

Paso 5 Haga clic en la ficha **Shards and Replicas**.

La página muestra todas las particiones de la instancia y la lista de réplicas de cada partición.

Paso 6 Haga clic en para mostrar todas las réplicas de una partición.

Figura 7-1 Listas de particiones y réplicas



Paso 7 Haga clic en **Promote to Master** en la fila que contiene otra réplica cuyo rol es **Replica**.

Paso 8 Haga clic en **Yes**.

----Fin

7.9 Análisis de caché

7.9.1 Análisis de claves grandes y claves con mucho uso

Al realizar análisis de claves grandes y claves con mucho uso, tendrá una imagen de las claves que ocupan un espacio grande y las claves a las que se accede con más frecuencia.

Notas sobre el análisis de clave grande:

- Todas las instancias de DCS Redis admiten análisis de las claves grandes.
- Durante el análisis de claves grandes, todas las claves serán atravesadas. Cuanto mayor sea el número de claves, más tiempo tardará el análisis.
- Realice el análisis de claves grandes durante las horas de menor actividad y evite los períodos de copias de respaldo automáticas.
- Para una instancia principal/en standby o de clúster, el análisis de clave grande se realiza en el nodo en standby, por lo que el impacto en la instancia es menor. Para una instancia

de nodo único, el análisis de clave grande se realiza en el único nodo de la instancia y reducirá el rendimiento de acceso a la instancia hasta en un 10%. Por lo tanto, realice análisis de claves grandes en instancias de nodo único durante las horas fuera de pico.

- Se conservan un máximo de 100 registros de análisis de claves grandes (20 para Strings y 80 para Lists/Sets/Zsets/Hashes) para cada instancia. Cuando se alcance este límite, se eliminarán los registros más antiguos para dejar espacio para los nuevos registros. También puede eliminar manualmente los registros que ya no necesita.

Notas sobre el análisis de claves con mucho uso:

- Solo las instancias de DCS Redis 4.0/5.0 admiten el análisis de claves con mucho uso, y el parámetro **maxmemory-policy** de las instancias debe establecerse en **allkeys-lfu** o **volatile-lfu**.
- Durante el análisis de las claves con mucho uso, todas las claves serán atravesadas. Cuanto mayor sea el número de claves, más tiempo tardará el análisis.
- Realice un análisis de las claves con mucho uso poco después de las horas pico para garantizar la precisión de los resultados del análisis.
- El análisis de las claves con mucho uso se realiza en el nodo principal de cada instancia y reducirá el rendimiento de acceso a la instancia hasta en un 10%.
- Se conserva un máximo de 100 registros de análisis para cada instancia. Cuando se alcance este límite, se eliminarán los registros más antiguos para dejar espacio para los nuevos registros. También puede eliminar manualmente los registros que ya no necesita.

NOTA

Realice análisis de las claves grandes y las claves con mucho uso durante las horas fuera de pico para evitar el uso del 100% de la CPU.

Procedimiento para el análisis de claves grandes

Paso 1 Inicie sesión en la **consola DCS**.

Paso 2 Haga clic en  en la esquina superior izquierda de la consola de gestión y seleccione una región.

NOTA

Seleccione la misma región que su servicio de aplicación.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Paso 4 Haga clic en el nombre de una instancia de DCS Redis.

Paso 5 Elija **Analysis and Diagnosis** > **Cache Analysis**.

Paso 6 En la página de ficha **Big Key Analysis**, puede iniciar manualmente un análisis de clave grande o programar un análisis automático diario.

Paso 7 Una vez finalizada una tarea de análisis, haga clic en **View** para ver los resultados del análisis.

Puede ver los resultados del análisis de diferentes tipos de datos.

NOTA

La consola muestra un máximo de 20 registros de análisis de claves grandes para cadenas y 80 para Lists, Sets, Zsets, y Hashes.

----Fin

Procedimiento para el análisis de claves con mucho uso

Paso 1 Inicie sesión en la **consola DCS**.

Paso 2 Haga clic en en la esquina superior izquierda de la consola de gestión y seleccione una región.

NOTA

Seleccione la misma región que su servicio de aplicación.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Paso 4 Haga clic en el nombre de una instancia de DCS Redis.

Paso 5 Elija **Analysis and Diagnosis** > **Cache Analysis**.

Paso 6 En la página de ficha **Hot Key Analysis**, puede iniciar manualmente un análisis de clave con mucho uso o programar un análisis automático diario.

NOTA

Si la instancia se creó antes de julio de 2020, el valor predeterminado del parámetro **maxmemory-policy** es **noeviction**. Antes de iniciar el análisis de claves con mucho uso, establezca este parámetro en **allkeys-lfu** o **volatile-lfu**. El valor predeterminado del parámetro **maxmemory-policy** de una instancia creada a partir de julio de 2020 es **volatile-lru**. Para realizar un análisis de teclas de acceso rápido, establezca este parámetro en **allkeys-lfu** o **volatile-lfu** en la página **Instance Configuration** > **Parameters**. Para obtener más información sobre **allkeys-lfu** y **volatile-lfu**, consulte **¿Qué es la política de desalojo de datos predeterminada?**

Paso 7 Una vez finalizada una tarea de análisis, haga clic en **View** para ver los resultados del análisis.

Se muestran los resultados del análisis de claves con mucho uso.

NOTA

La consola muestra un máximo de 100 registros de análisis de claves con mucho uso para cada instancia.

Tabla 7-7 Resultados del análisis de claves con mucho uso

Parámetro	Descripción
Key	Nombre de una clave con mucho uso.
Type	Tipo de una clave con mucho uso, que puede ser string, hash, list, set, o sorted set.
Size	Tamaño del valor de la clave con mucho uso.

Parámetro	Descripción
FREQ	Refleja la frecuencia de acceso de una clave dentro de un período de tiempo específico (normalmente 1 minuto). FREQ es el contador de frecuencia de acceso logarítmico. El valor máximo de FREQ es 255, lo que indica 1 millón de solicitudes de acceso. Después de que FREQ alcance 255, ya no se incrementará incluso si las solicitudes de acceso continúan aumentando. FREQ disminuirá en 1 por cada minuto durante el cual no se acceda a la clave.
Shard	La partición donde se encuentra la clave con mucho uso. **NOTA** Este parámetro sólo está disponible para instancias de clúster.
Database	Base de datos donde se encuentra una clave con mucho uso.

----Fin

FAQ sobre las claves grandes y las claves con mucho uso

- **¿Por qué la capacidad o el rendimiento de una partición de una instancia de Clúster Redis está sobrecargada cuando eso de la instancia está todavía por debajo del cuello de botella?**
- **¿Cuál es el impacto de una clave grande?**
- **¿Cuál es el impacto de una clave con mucho uso?**
- **¿Cómo evito las claves grandes y las claves con mucho uso?**
- **¿Cómo analizo las claves con mucho uso de una instancia de DCS Redis 3.0?**

7.9.2 Escaneo de claves caducadas

Hay dos formas de eliminar una clave en Redis.

- Utilice el comando DEL para eliminar directamente una clave.
- Utilice comandos como EXPIRE para establecer un tiempo de espera en una clave. Después de que transcurra el tiempo de espera, la clave se vuelve inaccesible pero no se elimina inmediatamente porque Redis es principalmente de un solo subproceso. Redis utiliza las siguientes estrategias para liberar la memoria utilizada por las claves caducadas:
  - Borrado libre perezoso: La estrategia de borrado se controla en el bucle principal de eventos de E/S. Antes de ejecutar una orden de lectura/escritura, se llama a una función para comprobar si la clave a la que se accede ha expirado. Si ha caducado, se eliminará y se devolverá una respuesta indicando que la clave no existe. Si la clave no ha caducado, se reanuda la ejecución del comando.
  - Eliminación programada: Una función de evento de tiempo se ejecuta en ciertos intervalos. Cada vez que se ejecuta la función, se comprueba una colección aleatoria de claves y se eliminan las claves caducadas.

> **NOTA**
>
> Para evitar bloqueos prolongados en el subproceso principal de Redis, no todas las claves se comprueban en cada evento de tiempo. En su lugar, una colección aleatoria de claves se comprueba cada vez. Como resultado, la memoria utilizada por las claves caducadas no se puede liberar rápidamente.

DCS integra estas estrategias y le permite liberar periódicamente la memoria utilizada por las claves caducadas. Puede configurar análisis programados en los nodos principals de sus instancias. Todo el espacio de claves se recorre durante las exploraciones, activando Redis para comprobar si las claves han caducado y para eliminar las claves caducadas si las hay.

> **NOTA**
>
> Esta función solo es compatible con las instancias de DCS Redis 4.0 y 5.0.

Procedimiento

Paso 1 Inicie sesión en la **consola DCS**..

Paso 2 Haga clic en en la esquina superior izquierda de la consola de gestión y seleccione una región.

> **NOTA**
>
> Seleccione la misma región que su servicio de aplicación.

Paso 3 En el panel de navegación, elija **Cache Manager**.

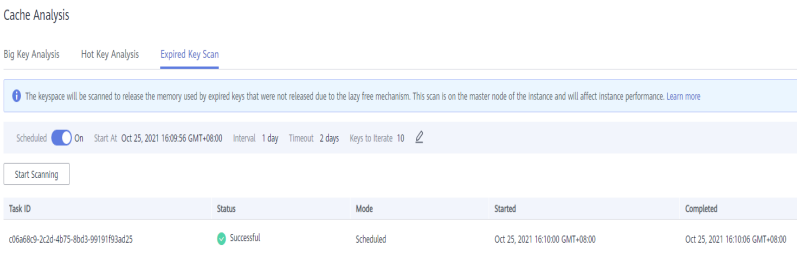
Paso 4 Haga clic en el nombre de una instancia de DCS Redis.

Paso 5 Elija **Analysis and Diagnosis** > **Cache Analysis**.

Paso 6 Haga clic en **Expired Key Scan**. Puede hacer clic en **Start Scanning** para escanear la instancia inmediatamente. También puede configurar una tarea programada para analizar automáticamente la instancia a la hora especificada.

Paso 7 Después de enviar la tarea de escaneo de clave caducada, puede verla en la lista de tareas.

Figura 7-2 Tareas de escaneo de clave caducadas



----Fin

NOTA

Un error de escaneo puede deberse a los siguientes problemas:

- Ocurrió una excepción.
- El escaneo agotó el tiempo de espera porque hay demasiadas claves. En este caso, se han eliminado algunas claves.

Programación de Escaneos Automáticos

Para programar análisis automáticos, haga clic en  junto a **Scheduled**. Defina los parámetros según sea necesario y haga clic en **OK**.

Tabla 7-8 describe los parámetros para programar escaneos automáticos.

Tabla 7-8 Parámetros para programar escaneos automáticos

Parámetro	Descripción	Rango de valores	Valor predeterminado	Notas
Start At	El primer escaneo solo puede comenzar después de la hora actual.	Formato: yyyy/MM/dd hh:mm:ss	-	-

Parámetro	Descripción	Rango de valores	Valor predeterminado	Notas
Interval	Intervalo entre escaneos.	0 a 43,200 (unidad: minuto)	1440	● Si el análisis anterior no se completa cuando llegue la hora de inicio, se omitirá el análisis siguiente. ● Si el análisis anterior se completa dentro de los cinco minutos posteriores a la hora de inicio, el análisis siguiente no se omitirá. **NOTA** Los análisis continuos pueden causar un uso elevado de la CPU. Establezca este parámetro en función del número total de claves en la instancia y el aumento de claves. Para obtener más información, consulte la siguiente descripción de rendimiento y sugerencias de configuración.

Parámetro	Descripción	Rango de valores	Valor predeterminado	Notas
Timeout	Este parámetro se utiliza para evitar el tiempo de espera de análisis debido a razones desconocidas. Si el escaneo se agota debido a razones desconocidas, no se pueden ejecutar las tareas programadas posteriores. Después de que transcurra el tiempo de espera especificado, se devuelve un mensaje de error y se realizará el siguiente análisis.	De 1 a 86,400 (unidad: minuto)	2880	● Configure el tiempo de espera con un valor al menos dos veces mayor que el intervalo. ● Puede establecer un valor basado en el tiempo empleado en los análisis anteriores y el tiempo de espera máximo que se puede tolerar en el escenario de la aplicación.

Parámetro	Descripción	Rango de valores	Valor predeterminado	Notas
Keys to Iterate	El comando **SCAN** se utiliza para iterar las claves de la base de datos actual. La opción **COUNT** se utiliza para permitir al usuario indicar al comando de iteración cuántos elementos deben devolverse del conjunto de datos en cada iteración. Para obtener más información, consulte la **descripción del comando SCAN**. La exploración iterativa puede reducir los riesgos de ralentizar Redis cuando se analizan un gran número de claves a la vez.	De 10 a 1000	10	Por ejemplo, si hay 10 millones de claves en Redis y el número de claves a iterar se establece en 1000, se completará un análisis completo después de las iteraciones de 10,000.

Rendimiento

- El comando SCAN se ejecuta en el plano de datos cada 5 ms, es decir, 200 veces por segundo. Si **Keys to Iterate** se establecen en 10, 100 o 1000, se analizan 2000, 20,000 o 200,000 claves por segundo.
- Cuanto mayor sea el número de claves escaneadas por segundo, mayor será el uso de la CPU.

Prueba de referencia

Se escanea una instancia principal/en standby. Hay 10 millones de claves que no caducarán y 5 millones de claves que caducarán. El tiempo de caducidad es de 1 a 10 segundos.

- Eliminación natural: los registros de 10,000 se eliminan por segundo. Se tarda 8 minutos en eliminar 5 millones de claves caducadas. El uso de la CPU es de aproximadamente 5%.
- **Keys to Iterate** configuradas en **10**: El escaneo tarda 125 minutos (15 millones/2000/60 segundos) y el uso de la CPU es de aproximadamente 8%.
- **Keys to Iterate** establecidas en **100**: El escaneo dura 12.5 minutos (15 millones/ 20,000/60 segundos) y el uso de CPU es de aproximadamente 20%.
- **Keys to Iterate** configuradas en **1000**: El escaneo tarda 1.25 minutos (15 millones/ 200,000/60 segundos) y el uso de la CPU es de aproximadamente el 25%.

Sugerencias de configuración

- Puede configurar el número de claves que se escanearán y el intervalo de escaneo en función del número total de claves y el aumento del número de claves en la instancia.
- En la prueba de referencia con 15 millones de teclas y **Keys to Iterate** establecidas en 10, el escaneo dura aproximadamente 125 minutos. En este caso, ajuste el intervalo a más de 4 horas.
- Si desea acelerar el escaneo, establezca **Keys to Iterate** en 100. Se tarda aproximadamente 12.5 minutos en completar el escaneo. Por lo tanto, ajuste el intervalo a más de 30 minutos.
- Cuanto mayor sea el número de teclas a iterar, más rápido será el escaneo y mayor será el uso de la CPU. Hay una compensación entre el tiempo y el uso de la CPU.
- Si el número de claves caducadas no aumenta rápidamente, puede escanear las claves caducadas una vez al día.

  NOTA

  Comience a escanear durante las horas no pico. Establezca el intervalo en un día y el tiempo de espera en dos días.

7.10 Observación de consultas lentas de Redis

Redis registra las consultas que exceden un tiempo de ejecución especificado. Puede ver los registros lentos en la consola DCS para identificar problemas de rendimiento.

Para obtener más información sobre los comandos, visite el **sitio web oficial de Redis**.

Configure consultas lentas con los siguientes parámetros:

- **slowlog-log-slower-than**: El tiempo máximo permitido, en microsegundos, para la ejecución del comando. Si se supera este umbral, Redis registrará el comando. El valor predeterminado es "10,000". Es decir, si la ejecución del comando supera los 10 ms, el comando se registrará.
- **slowlog-max-len**: El número máximo permitido de consultas lentas que se pueden registrar. El valor predeterminado es **128**. Es decir, si el número de consultas lentas excede de 128, el registro más antiguo se eliminará para dejar espacio a los nuevos.

Para obtener más información sobre los parámetros de configuración, consulte **Modificación de parámetros de configuración de una instancia**.

NOTA

Puede ver las consultas lentas de una instancia de Clúster Proxy de DCS compatible con Redis 3.0 solo si la instancia se crea después del 14 de octubre de 2019.The upgrade adds the slow query function without affecting services.

Observación de consultas lentas en la consola

Paso 1 Inicie sesión en la **consola DCS**.

Paso 2 Haga clic en en la esquina superior izquierda de la consola de gestión y seleccione una región.

NOTA

Seleccione la misma región que su servicio de aplicación.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Paso 4 Haga clic en el nombre de una instancia de DCS.

Paso 5 Elija **Analysis and Diagnosis** > **Slow Queries**.

Paso 6 Seleccione una fecha de inicio y una fecha de finalización para ver las consultas lentas dentro del período especificado.

NOTA

- Para obtener más información sobre los comandos, visite el **sitio web oficial de Redis**.
- Actualmente, puede ver consultas lentas en los últimos siete días.

Figura 7-3 Consultas lentas de una instancia

----Fin

7.11 Consulta de registros de ejecución de Redis

Puede crear archivos de registro de ejecución en la consola de DCS para recopilar registros de ejecución de instancias de DCS Redis dentro de un período especificado. Después de recopilar los registros, puede descargar los archivos de registro para ver los registros.

NOTA

Esta función es compatible con instancias de DCS Redis 4.0 y posteriores.

Procedimiento

Paso 1 Inicie sesión en la **consola DCS**.

Paso 2 Haga clic en en la esquina superior izquierda de la consola de gestión y seleccione la región donde se encuentra la instancia.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Paso 4 Haga clic en una instancia de DCS.

Paso 5 Haga clic en la ficha **Run Logs**.

Paso 6 Haga clic en **Create Log File** y especifique las condiciones de recopilación.

Si la instancia es el tipo principal/en standby o de clúster, puede especificar la partición y la réplica cuyos registros de ejecución desea recopilar. Si la instancia es el tipo de nodo único, se recopilarán los registros del único nodo de la instancia.

----Fin

7.12 Diagnóstico de una instancia

Caso

Si se produce una falla o un problema de rendimiento, puede solicitar a DCS que diagnostique su instancia para obtener información sobre la causa y el impacto del problema y cómo manejarlo.

Restricciones

- Las instancias de DCS Redis 3.0 y Memcached no admiten diagnósticos.

Procedimiento

Paso 1 Inicie sesión en la **consola DCS**..

Paso 2 Haga clic en en la esquina superior izquierda de la consola de gestión y seleccione la región donde se encuentra la instancia.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Paso 4 Haga clic en el nombre de una instancia de DCS Redis.

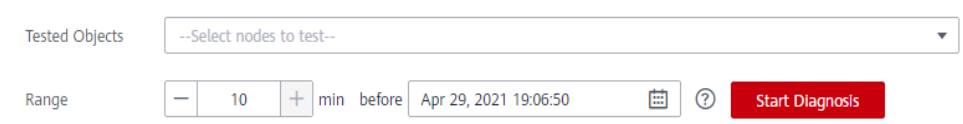
Paso 5 Elija **Analysis and Diagnosis** > **Instance Diagnosis**.

Paso 6 Especifique el objeto probado y el intervalo de tiempo y haga clic en **Start Diagnosis**.

- **Tested Object**: Puede seleccionar un solo nodo o todos los nodos.
- **Range**: Puede especificar hasta 10 minutos antes de un punto en el tiempo en los últimos 7 días.

  En la siguiente figura, se diagnosticarán los datos de instancia entre 18:56:50 y 19:06:50 del 7 de enero de 2021.

  Figura 7-4 Especificación del objeto probado y el intervalo de tiempo



Paso 7 Una vez completado el diagnóstico, puede ver el resultado en la lista **Test History**. Si el resultado es anormal, haga clic en **View Report** para obtener más detalles.

En el informe, puede ver la causa y el impacto de los elementos anormales y sugerencias para su manejo.

----Fin

8 Copia de seguridad y restauración de instancias

8.1 Descripción general

En la consola de DCS, puede realizar copias de seguridad y restaurar instancias de DCS.

Importancia de la copia de seguridad de instancia de DCS

Existe una pequeña posibilidad de que existan datos sucios en una instancia DCS debido a excepciones del sistema de servicio o problemas en la carga de datos desde archivos de persistencia. Además, algunos sistemas exigen no solo una alta fiabilidad, sino también seguridad de los datos, restauración de datos e incluso almacenamiento de datos permanente.

Actualmente, los datos en instancias DCS se pueden realizar copias de seguridad en OBS. Si una instancia de DCS se vuelve defectuosa, los datos de la instancia se pueden restaurar desde la copia de seguridad para que la continuidad del servicio no se vea afectada.

Modos de copia de seguridad

Las instancias de DCS admiten los siguientes modos de copia de seguridad:

- Copia de seguridad automatizada

  Puede crear una política de copia de seguridad programada en la consola DCS. A continuación, los datos de las instancias de DCS elegidas se respaldarán automáticamente a la hora programada.

  Puede elegir los días de la semana en los que se ejecutará la copia de seguridad programada. Los datos de copia de seguridad se conservarán durante un máximo de siete días. Los datos de copia de seguridad de más de siete días se eliminarán automáticamente.

  El propósito principal de las copias de seguridad automatizadas es crear réplicas de datos completas de instancias DCS para que la instancia se pueda restaurar rápidamente si es necesario.

- Copia de respaldo manual

  Las solicitudes de copia de seguridad también se pueden emitir manualmente. A continuación, los datos de las instancias de DCS elegidas se respaldarán

permanentemente en OBS. Los datos de copia de seguridad se pueden eliminar manualmente.

Antes de realizar operaciones de alto riesgo, como el mantenimiento o la actualización del sistema, realice una copia de seguridad de los datos de instancia de DCS.

Información adicional sobre la copia de seguridad de datos

- Tipo de instancia
  - Redis: Solo las instancias principal/en standby, de Clúster Proxy, de Clúster Redis y de separación de lectura/escritura pueden ser respaldadas y restauradas, mientras que las instancias de nodo único no pueden. Sin embargo, puede exportar datos de una instancia de nodo único a un archivo RDB mediante redis-cli. Para obtener más información, consulte **¿Cómo exporto datos de instancia de DCS Redis?**
  - Memcached: Solo se pueden hacer copias de seguridad y restaurar las instancias principal/en standby, mientras que las instancias de nodo único no.
- Mecanismo de copia de seguridad

  DCS for Redis 3.0 persiste los datos en archivos AOF. DCS for Redis 4.0 y 5.0 persisten los datos en archivos RDB o AOF en modo de copia de seguridad manual, y en archivos RDB en modo de copia de seguridad automática.

  Para exportar archivos de copia de seguridad RDB de instancias DCS Redis 3.0, ejecute el comando **redis-cli -h** *{redis_address}* **-p 6379 [-a** *{password}***] --rdb** *{output.rdb}* en redis-cli.

  NOTA

  - DCS for Redis 3.0 ya no se proporciona. Puede utilizar DCS for Redis 4.0 o 5.0 en su lugar.
  - Para una instancia de DCS Redis 3.0 de un solo nodo en la que se puede ejecutar el comando SYNC, puede ejecutar este comando para exportar el archivo RDB. Para una instancia de Clúster Proxy DCS Redis 3.0, el comando SYNC no se puede ejecutar debido a la arquitectura. Por lo tanto, el archivo RDB no se puede exportar.

  Las tareas de copia de seguridad se ejecutan en nodos de caché en standby. Se realiza una copia de seguridad de los datos de instancia DCS comprimiendo y almacenando los archivos de persistencia de datos desde el nodo de caché en standby a OBS.

  DCS comprueba las políticas de copia de seguridad de instancia una vez por hora. Si una política de copia de seguridad coincide, DCS ejecuta una tarea de copia de seguridad para la instancia de DCS correspondiente.
- Impacto en las instancias de DCS durante el backup

  Las tareas de copia de seguridad se ejecutan en nodos de caché en standby, sin incurrir en ningún tiempo de inactividad.

  En caso de sincronización de datos completos o carga de instancia pesada, se tarda unos minutos en completar la sincronización de datos. Si la copia de seguridad de instancia comienza antes de que se complete la sincronización de datos, los datos de copia de seguridad estarán ligeramente detrás de los datos en el nodo de caché principal.

  Durante la copia de seguridad de la instancia, el nodo de caché en standby deja de persistir los últimos cambios en los archivos de disco. Si se escriben nuevos datos en el nodo de caché principal durante la copia de seguridad, el archivo de copia de seguridad no contendrá los nuevos datos.
- Tiempo de respaldo

  Es aconsejable realizar copias de seguridad de los datos de instancia durante períodos fuera de pico.

- Almacenamiento y precios de los archivos de copia de seguridad

  Los archivos de copia de seguridad se almacenan en OBS.

  DCS proporciona el servicio de copia de seguridad de forma gratuita, pero se incurrirán cargos OBS por la cantidad y el período en que se consuma el espacio de almacenamiento.

- Manejo de excepciones en copias de seguridad programadas

  Si se activa una tarea de copia de seguridad programada mientras la instancia de DCS se está reiniciando o se está escalando, la tarea de copia de seguridad programada se ejecutará en el siguiente ciclo.

  Si la copia de seguridad de una instancia de DCS falla o la copia de seguridad se pospone porque hay otra tarea en curso, DCS intentará hacer una copia de seguridad de la instancia en el siguiente ciclo. Se permite un máximo de tres reintentos en un solo día.

- Período de retención de datos de backup

  Los archivos de copia de seguridad programados se conservan durante un máximo de siete días. Puede configurar el período de retención. Al final del período de retención, la mayoría de los archivos de copia de seguridad de la instancia de DCS se eliminarán automáticamente, pero se conservará al menos un archivo de copia de seguridad.

  Los archivos de copia de seguridad manuales se conservan de forma permanente y deben eliminarse manualmente.

Restauración de datos

- Proceso de restauración de datos

  a. Puede iniciar una solicitud de restauración de datos mediante la consola DCS.
  b. DCS obtiene el archivo de copia de seguridad de OBS.
  c. Se suspende la lectura/escritura en la instancia de DCS.
  d. El archivo de persistencia de datos original del nodo de caché principal se sustituye por el archivo de copia de seguridad.
  e. El nuevo archivo de persistencia de datos (es decir, el archivo de copia de seguridad) se vuelve a cargar.
  f. Los datos se restauran y la instancia de DCS comienza a proporcionar un servicio de lectura/escritura de nuevo.

- Impacto en los sistemas de servicio

  Las tareas de restauración se ejecutan en los nodos de caché principal. Durante la restauración, los datos no se pueden escribir ni leer de instancias.

- Manejo de excepciones de restauración de datos

  Si un archivo de copia de seguridad está dañado, DCS intentará reparar el archivo de copia de seguridad mientras restaura los datos de instancia. Si el archivo de copia de seguridad se corrige correctamente, la restauración continúa. Si el archivo de copia de seguridad no se puede arreglar, la instancia principal/en standby de DCS se cambiará de nuevo al estado en el que estaba antes de la restauración de los datos.

8.2 Configuración de la política de copia de seguridad

En la consola DCS, puede configurar una política de copia de seguridad automática. A continuación, el sistema realiza una copia de seguridad de los datos de sus instancias de acuerdo con la política de copia de seguridad.

Si no se requiere una copia de seguridad automática, deshabilite la función de copia de seguridad automática en la política de copia de seguridad.

Prerrequisitos:

Una instancia de DCS principal/en standby, de clúster o de separación de lectura/escritura se encuentra en el estado **Running**.

Procedimiento

Paso 1 Inicie sesión en la **consola DCS**..

Paso 2 Haga clic en  en la esquina superior izquierda de la consola de gestión y seleccione la región donde se encuentra la instancia.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Filtrar las instancias de DCS para buscar la instancia DCS deseada. Actualmente, puede buscar instancias por nombre, especificación, ID, dirección IP, AZ, estado, tipo de instancia, motor de caché y muchos otros atributos.

Paso 4 Haga clic en el nombre de la instancia de DCS deseada para ir a la página de detalles.

Paso 5 En la página de detalles de la instancia, haga clic en **Backups & Restorations**.

Paso 6 Deslice  a la derecha para habilitar la copia de seguridad automática. Se mostrarán las políticas de copia de seguridad.

Tabla 8-1 Parámetros en una política de copia de seguridad

Parámetro	Descripción
Backup Schedule	El día de la semana en la que se realiza una copia de seguridad automática de los datos de la instancia de DCS elegida. Puede seleccionar uno o varios días de una semana.
Retention Period (days)	El número de días que se conservan los datos de copia de seguridad automática. Los datos de copia de seguridad se eliminarán permanentemente al final del período de retención y no se podrán restaurar. Rango de valores: 1–7.

Parámetro	Descripción
Start Time	La hora en la que se inicia la copia de seguridad automática. Valor: la hora completa entre las 00:00 y las 23:00 DCS comprueba las políticas de copia de seguridad una vez cada hora. Si ha llegado la hora de inicio de la copia de seguridad en una política de copia de seguridad, se realiza una copia de seguridad de los datos de la instancia correspondiente. **NOTA** La copia de seguridad de instancias tarda de 5 a 30 minutos. Los datos agregados o modificados durante el proceso de copia de seguridad no serán respaldados. Para reducir el impacto de la copia de seguridad en los servicios, se recomienda que se realice una copia de seguridad de los datos durante los períodos fuera de pico. Solo se pueden realizar copias de seguridad de las instancias en el estado **Running**.

Paso 7 Haz clic en **OK**.

----Fin

8.3 Copia de seguridad manual de una instancia de DCS

Puede realizar una copia de seguridad manual de los datos en instancias de DCS de manera oportuna. En esta sección se describe cómo realizar una copia de seguridad manual de los datos en instancias principal/en standby mediante la consola DCS.

De forma predeterminada, los datos de copia de seguridad manualmente se conservan permanentemente. Si los datos de copia de seguridad ya no están en uso, puede eliminarlos manualmente.

Prerrequisitos:

Una instancia de DCS principal/en standby, de clúster o de separación de lectura/escritura se encuentra en el estado **Running**.

Procedimiento

Paso 1 Inicie sesión en la **consola DCS**.

Paso 2 Haga clic en  en la esquina superior izquierda de la consola de gestión y seleccione la región donde se encuentra la instancia.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Filtrar las instancias de DCS para buscar la instancia DCS deseada. Actualmente, puede buscar instancias por nombre, especificación, ID, dirección IP, AZ, estado, tipo de instancia, motor de caché y muchos otros atributos.

Paso 4 Haga clic en el nombre de la instancia de DCS deseada para ir a la página de detalles.

Paso 5 En la página de detalles de la instancia, haga clic en **Backups & Restorations**.

Paso 6 Haga clic en **Create Backup**.

Paso 7 Seleccione un formato de archivo de copia de seguridad.

Solo las instancias de DCS Redis 4.0/5.0 admiten la selección de formato de archivo de copia de seguridad.

Paso 8 En el cuadro de diálogo **Create Backup**, haga clic en **OK**.

La información del cuadro de texto **Description** no puede superar los 128 bytes.

NOTA

La copia de seguridad de instancias tarda de 10 a 15 minutos. Los datos agregados o modificados durante el proceso de copia de seguridad no serán respaldados.

----Fin

8.4 Restauración de una instancia de DCS

En la consola de DCS, puede restaurar los datos de copia de seguridad en una instancia de DCS elegida.

Prerrequisitos:

- Una instancia de DCS principal/en standby, de clúster o de separación de lectura/escritura se encuentra en el estado **Running**.
- Se ha ejecutado una tarea de copia de seguridad para realizar una copia de seguridad de los datos de la instancia que se va a restaurar y la tarea de copia de seguridad se ha realizado correctamente.

Procedimiento

Paso 1 Inicie sesión en la **consola DCS**.

Paso 2 Haga clic en en la esquina superior izquierda de la consola de gestión y seleccione la región donde se encuentra la instancia.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Filtrar las instancias de DCS para buscar la instancia DCS deseada. Actualmente, puede buscar instancias por nombre, especificación, ID, dirección IP, AZ, estado, tipo de instancia, motor de caché y muchos otros atributos.

Paso 4 Haga clic en el nombre de la instancia de DCS deseada para ir a la página de detalles.

Paso 5 En la página de detalles de la instancia, haga clic en **Backups & Restorations**.

A continuación, se muestra una lista de tareas de copia de seguridad históricas.

Paso 6 Haga clic en **Restore** en la fila que contiene la tarea de copia de seguridad elegida.

Paso 7 Haga clic en **OK** para iniciar la restauración de instancias.

La información del cuadro de texto **Description** no puede superar los 128 bytes.

Puede ver los resultados de todas las tareas de restauración en la página **Restoration History**. Los registros no se pueden eliminar.

NOTA

La restauración de instancias tarda de 1 a 30 minutos.

Mientras se restaura, las instancias de DCS no aceptan solicitudes de operación de datos de los clientes porque los datos de copia de seguridad sobrescriben los datos existentes.

----Fin

8.5 Descarga de un archivo de copia de seguridad RDB o AOF

Los datos respaldados automáticamente se pueden conservar durante un máximo de 7 días. La copia de seguridad manual de los datos no es gratuita y ocupa espacio en OBS. Debido a estas limitaciones, se recomienda descargar los archivos de copia de seguridad RDB y AOF y guardarlos permanentemente en el host local.

Esta función solo es compatible con instancias principal/en standby, y no con instancias de nodo único. Para exportar los datos de una instancia de nodo único a un archivo RDB, puede utilizar redis-cli. Para obtener más información, consulte **¿Cómo exporto datos de instancia de DCS Redis?**

Para exportar los datos de una instancia principal/en standby o de clúster, haga lo siguiente:

- Redis 3.0: Exporte los datos de instancia a archivos AOF mediante la consola DCS o a archivos RDB ejecutando **redis-cli -h** *{redis_address}* **-p 6379 [-a** *{password}***] --rdb** *{output.rdb}* mediante redis-cli.

  NOTA

  DCS for Redis 3.0 ya no se proporciona. Puede utilizar DCS for Redis 4.0 o 5.0 en su lugar.

- Redis 4.0 y 5.0: Exporte los datos de instancia a archivos AOF o RDB mediante la consola DCS.

Prerrequisitos:

Se ha realizado una copia de seguridad de la instancia y la copia de seguridad sigue siendo válida.

Procedimiento

Paso 1 Inicie sesión en la **consola DCS**.

Paso 2 Haga clic en en la esquina superior izquierda de la consola de gestión y seleccione la región donde se encuentra la instancia.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Filtrar las instancias de DCS para buscar la instancia DCS deseada. Actualmente, puede buscar por nombre, especificación, ID, dirección IP, AZ, estado, tipo de instancia, motor de caché, proyecto de empresa, modo de facturación y etiquetas.

Paso 4 Haga clic en el nombre de la instancia de DCS para mostrar más detalles sobre la instancia de DCS.

Paso 5 En la página de detalles de la instancia, haga clic en **Backups & Restorations**.

A continuación, se muestra una lista de tareas de copia de seguridad históricas.

Paso 6 Haga clic en **Download** en la fila que contiene la tarea de copia de seguridad elegida.

Paso 7 En el cuadro de diálogo **Download Backup File** que se muestra, seleccione cualquiera de los dos métodos de descarga siguientes.

Métodos de descarga:

- Por URL
  a. Paso 1: Establece el período de validez de la URL y haz clic en **Query**.
  b. Paso 2: Descargue el archivo de copia de seguridad utilizando la lista de URL.

  NOTA

  Si elige copiar URLs, use comillas para citar las URL cuando ejecute el comando **wget** en Linux. Por ejemplo:

  wget 'https://obsEndpoint.com:443/redisdemo.rdb?parm01=value01&parm02=value02'

  Esto se debe a que la URL contiene el carácter especial y (&), que confundirá el comando **wget**. Citar la URL facilita la identificación de la misma.
- Por OBS

  Siga el procedimiento mostrado.

----Fin

9 Migración de datos de instancia

9.1 Descripción general de la migración de datos

La consola DCS admite la migración en línea (total o incrementalmente) y la migración de backup (mediante la importación de archivos de backup) con operaciones intuitivas.

- La migración de copia de seguridad es adecuada cuando las instancias de origen y destino de Redis no están conectadas y la instancia de origen de Redis no admite los comandos SYNC y PSYNC. Para migrar datos, importe los archivos de copia de seguridad a OBS, y DCS leerá los datos de OBS y migrará los datos a la instancia de DCS Redis de destino. Alternativamente, puede importar los archivos de copia de seguridad directamente a la instancia de DCS.
- La migración en línea es adecuada cuando la instancia Redis de origen admite los comandos SYNC y PSYNC. Los datos de la instancia de Redis de origen se pueden migrar de forma completa o incremental a la instancia de destino.

  Durante la migración en línea, el comando PSYNC se entrega a la dirección de origen. Para obtener más información sobre cómo funciona esto, consulte la **explicación de la replicación**. Este comando provocará una operación de bifurcación en el extremo de origen, lo que afecta a la latencia. Para obtener más información sobre el alcance del impacto, consulte el **sitio web oficial de Redis**.

NOTA

Actualmente, la función de migración de datos es gratuita en la OBT. Se le notificará cuando comience a cobrarse la migración de datos.

Para obtener más información acerca de las herramientas y esquemas de migración, consulte **Herramientas y esquemas de migración**.

Tabla 9-1 Modos de migración de datos DCS

Modo de migración	Fuente	Target: DCS		
		Single-Node and Master/Standby	Proxy Cluster	Redis Cluster

Importación de archivos de copia de respaldo	OBS bucket: AOF files **NOTA** Los archivos AOF exportados desde instancias de Huawei Cloud Redis 4.0/5.0 y otras instancias con compresión RDB activada no se pueden importar.	√	√	×
	OBS bucket: RDB files	√	√	√
Migración de datos en línea	DCS for Redis: single-node or master/standby	√	√	√
	DCS for Redis: Proxy Cluster **NOTA** Las instancias de Proxy Cluster DCS Redis 3.0 no se pueden usar como origen, mientras que las instancias de Proxy Cluster DCS Redis 4.0 o 5.0 sí.	√	√	√
	DCS for Redis: Redis Cluster	√	√	√
	Self-hosted single-node or master/standby Redis	√	√	√
	Self-hosted proxy-based cluster Redis	√	√	√
	Self-hosted Redis Cluster	√	√	√
	Other Redis: single-node or master/standby	×	×	×
	Other Redis: proxy-based cluster	×	×	×
	Other Redis: Redis Cluster	×	×	×

NOTA

- **DCS for Redis** se refiere a instancias de Redis proporcionadas por DCS
- **Self-hosted Redis** se refiere a Redis autohospedado en la nube, de otros proveedores de nube o en centros de datos locales.
- **Other cloud Redis** se refiere a los servicios de Redis proporcionados por otros proveedores de nube.
- √ apoyado. ×: No soportado.
- Puede migrar datos en línea de forma completa o incremental desde **other cloud Redis** a **DCS for Redis** si están conectados y los comandos **SYNC** y **PSYNC** se pueden ejecutar en el Redis de origen. Sin embargo, algunas instancias proporcionadas por otros proveedores de nube pueden no ser migrados en línea. En este caso, migre datos a través de la importación de copias de seguridad o utilice otros esquemas de migración. Para obtener más información, consulte **Herramientas de migración y herramientas de Schemes**.

9.2 Importación de archivos de copia de seguridad desde un bucket de OBS

Caso

Utilice la consola DCS para migrar datos de Redis desde Redis de otra nube o Redis autohospedado a Huawei Cloud DCS for Redis.

Simplemente descargue los datos de Redis de origen y luego cargue los datos a un bucket OBS en la misma región que la instancia de DCS Redis de destino. Después de crear una tarea de migración en la consola DCS, DCS leerá los datos del bucket OBS y los datos se migrarán a la instancia de destino.

Los archivos .aof, .rbb, .zip y .tar.gz se pueden cargar en buckets de OBS. Puede cargar directamente archivos .aof y .rdb o comprimirlos en archivos .zip o .tar.gz antes de cargarlos.

Prerrequisitos

- El bucket de OBS debe estar en la misma región que la instancia de DCS Redis de destino.
- Los archivos de datos que se van a cargar deben estar en formato .aof, .rdb, .zip o .tar.gz.
- Para migrar datos desde una instancia de Redis de nodo único o principal/en standby de otra nube, cree una tarea de copia de seguridad y descargue el archivo de copia de seguridad.
- Para migrar datos desde una instancia de Redis de clúster de otra nube, descargue todos los archivos de copia de seguridad, cargue todos ellos en el bucket OBS y seleccione todos ellos para la migración. Cada archivo de copia de seguridad contiene datos para una partición de la instancia.
- Los archivos de copia de seguridad .rdb de Redis 5.0 autohospedados no se pueden importar. Los archivos de copia de seguridad .rdb de Redis 3.0 o 4.0 autohospedados se pueden exportar usando redis-cli. Los archivos .rdb de otra nube Redis solo se pueden exportar creando tareas de copia de seguridad y no se pueden exportar ejecutando comandos en redis-cli.
- Las instancias de Clúster Redis solo admiten archivos .rdb.

Paso 1: Preparar la instancia de DCS Redis de destino

- Si una instancia de DCS Redis de destino no está disponible, cree una primera. Para obtener más información, consulte **Compra de una instancia de DCS Redis**.
- Si ya tiene una instancia de DCS Redis, no es necesario crear una de nuevo, pero debe borrar los datos de instancia antes de la migración. Para obtener más información, consulte **Borrado de datos de la instancia de DCS**.

Puede utilizar una instancia de DCS Redis 3.0, 4.0 o 5.0 como instancia de destino.

Paso 2: Crear un bucket de OBS y cargar archivos de copia de seguridad

Paso 1 Cargue los archivos de datos de copia de seguridad en el bucket de OBS usando OBS Browser+.

Si el archivo de copia de seguridad que se va a cargar es menor que 5 GB, vaya al paso **Paso 2** para cargar el archivo usando la consola OBS.

Si el archivo de copia de seguridad que se va a cargar es mayor que 5 GB, siga las **instrucciones** proporcionadas por OBS.

Paso 2 En la consola OBS, cargue los archivos de datos de copia de seguridad en el bucket OBS.

Si los archivos de copia de seguridad son menores a 5 GB, realice los siguientes pasos:

1. Cree un bucket de OBS.

   Al crear un bucket de OBS, preste atención a la configuración de los siguientes parámetros. Para obtener más información sobre cómo establecer otros parámetros, consulte **Creación de un Bucket** en la *Guía del usuario de OBS*.

   a. Región:

      El bucket de OBS debe estar en la misma región que la instancia de DCS Redis de destino.

   b. **Storage Class**: seleccione **Standard** o **Infrequent Access**.

      No seleccione **Archive**. De lo contrario, la tarea de migración no se ejecutará.
2. En la lista de buckets, haga clic en el bucket creado en **Paso 2.1**.
3. En el panel de navegación, elija **Objects**.
4. En la página de la ficha **Objects**, haga clic en **Upload Object**.
5. Especifique **Storage Class**.

   No seleccione **Archive**. De lo contrario, la tarea de migración no se ejecutará.
6. Sube los objetos.

   Arrastre archivos o carpetas al área **Upload Object** o haga clic en **add file**.

   Se pueden cargar 100 archivos como máximo a la vez. El tamaño total no puede superar los 5 GB.

Figura 9-1 Carga de objetos por lotes

7. (Opcional) Seleccione **KMS encryption** para cifrar los archivos cargados.
8. Haga clic en **Upload**.

----Fin

Paso 3: Crear una tarea de migración

Paso 1 Iniciar sesión en la consola de DCS.

Paso 2 En el panel de navegación, elija **Data Migration**.

Paso 3 Haga clic en **Create Backup Import Task**.

Paso 4 Introduzca el nombre y la descripción de la tarea.

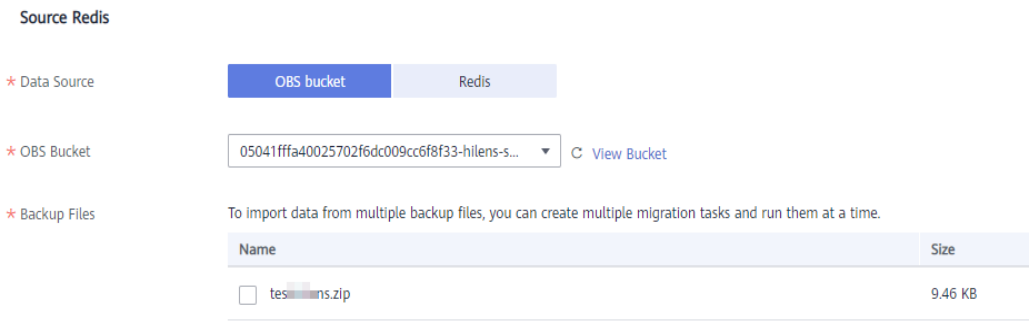
Paso 5 En el área **Source Redis**, seleccione **OBS Bucket** para **Data Source** y, a continuación, seleccione el bucket OBS en el que ha cargado los archivos de copia de seguridad.

En la tabla **Backup Files**, se muestran los archivos que ha subido.

NOTA

Puede cargar archivos en el formato .aof, .rdb, .zip o .tar.gz.

Figura 9-2 Especificación de la información del archivo de copia de seguridad



Paso 6 Seleccione los archivos de copia de seguridad cuyos datos se van a migrar.

Paso 7 Seleccione la instancia de Redis de destino preparada en **Paso 1: Preparar la instancia de DCS Redis de destino**. Si la instancia de Redis de destino tiene una contraseña, introduzca la contraseña y pruebe la conexión para comprobar si la contraseña es correcta.

Paso 8 Haga clic en **Next**.

Paso 9 Confirme los detalles de la tarea de migración y haga clic en **Submit**.

Vuelva a la lista de tareas de migración de datos. Una vez que la migración se realiza correctamente, el estado de la tarea cambia a **Successful**.

----Fin

9.3 Importación de archivos de copia de seguridad desde Redis

Caso

Puede migrar los datos de backup de Redis a instancias principal/en standby o de clúster de DCS Redis.

Simplemente realice una copia de seguridad de sus datos de Redis, cree una tarea de migración en la consola de DCS y, a continuación, importe la copia de seguridad en una instancia de DCS Redis.

Prerrequisitos:

Se ha creado una instancia de DCS Redis de destino como destino para la migración. La instancia de origen tiene datos escritos y se ha realizado una copia de seguridad.

Paso 1: Obtener el nombre de instancia de origen

Obtenga el nombre de la instancia de Redis de origen.

Paso 2: Preparar la instancia de DCS Redis de destino

- Si una instancia de DCS Redis no está disponible, cree una primera. Para más detalles, consulte **Compra de una instancia de DCS Redis**.
- Si hay disponible una instancia de DCS Redis, no es necesario crear una nueva. Sin embargo, debe borrar los datos de instancia antes de la migración. Para más detalles, consulte **Borrado de datos de la instancia de DCS**.

Puede utilizar una instancia de DCS Redis 3.0, 4.0 o 5.0 como instancia de destino.

Paso 3: Crear una tarea de migración

Paso 1 Iniciar sesión en la consola de DCS.

Paso 2 En el panel de navegación, elija **Data Migration**. Se muestra la lista de tareas de migración.

Paso 3 Haga clic en **Create Backup Import Task**.

Paso 4 Introduzca el nombre y la descripción de la tarea.

Paso 5 Establezca **Data Source** en **Redis**.

Paso 6 Para **Source Redis Instance**, seleccione la instancia preparada en **Paso 1: Obtener el nombre de instancia de origen**.

Paso 7 Seleccione la tarea de copia de seguridad cuyos datos se van a migrar.

Paso 8 Seleccione la instancia de destino creada en **Paso 2: Preparar la instancia de DCS Redis de destino**.

Paso 9 Introduzca la contraseña de la instancia de destino. Haga clic en **Test Connection** para verificar la contraseña.

Paso 10 Haga clic en **Next**.

Paso 11 Confirme los detalles de la tarea de migración y haga clic en **Submit**.

Vuelva a la lista de tareas de migración de datos. Una vez que la migración se realiza correctamente, el estado de la tarea cambia a **Successful**.

----Fin

9.4 Migración en línea

Caso

Si las instancias de origen y de destino están interconectadas y la instancia de origen admite los comandos SYNC y PSYNC, los datos se pueden migrar en línea completa o incrementalmente desde el origen al destino.

ATENCIÓN

- Si los comandos SYNC y PSYNC están deshabilitados en la instancia de Redis de origen, habilítelos antes de realizar la migración en línea. De lo contrario, la migración no se podrá realizar. Si utiliza una instancia de Huawei Cloud DCS Redis para la migración en línea, el comando SYNC se activa automáticamente.
- No puede utilizar redes públicas para la migración en línea.
- Durante la migración en línea, se recomienda establecer **repl-timeout** en la instancia de origen en 300s y **client-output-buffer-limit** en 20% de la memoria máxima de la instancia.

NOTA

Durante la migración en línea, los resultados de los comandos FLUSHDB y FLUSHALL ejecutados en el origen no se sincronizarán con el destino.

Impacto en los servicios

Durante la migración en línea, los datos se sincronizan en su totalidad con una nueva réplica. Por lo tanto, realice la migración en línea durante las horas de baja demanda.

Prerrequisitos:

- Antes de migrar datos, lea **Herramientas y esquemas de migración** para obtener información sobre la función de migración de datos de DCS y seleccione una instancia de destino adecuada.
- De forma predeterminada, una instancia de Clúster Proxy solo tiene una base de datos (DB0). Antes de migrar datos desde una instancia de nodo único o principal/en standby a una instancia de Clúster Proxy, compruebe si existen datos en las DB distintas de DB0. En caso afirmativo, habilite multi-DB para la instancia Clúster Proxy haciendo referencia a **Activación de Multi-DB**.
- De forma predeterminada, una instancia de Clúster Redis solo tiene una base de datos (DB0). Antes de migrar datos desde una instancia de nodo único o principal/en standby a una instancia de Clúster Redis, compruebe si existen datos en bases de datos distintas de DB0. Para garantizar que la migración tenga éxito, mueva todos los datos a DB0 haciendo referencia a **Migración en línea con Rump**.

Paso 1: Obtener información acerca de la instancia de origen Redis

- Si el origen es una instancia de Cloud Redis, obtenga su nombre.
- Si el origen es Redis autohospedado, obtenga su dirección IP o nombre de dominio y número de puerto.

Paso 2: Preparar la instancia de DCS Redis de destino

- Si una instancia de DCS Redis de destino no está disponible, cree una primera. Para obtener más información, consulte **Compra de una instancia de DCS Redis**.
- Si ya tiene una instancia de DCS Redis, no es necesario crear una de nuevo, pero debe borrar los datos de instancia antes de la migración. Para obtener más información, consulte **Borrado de datos de la instancia de DCS**.

  Si los datos de la instancia de destino no se borran antes de la migración y las instancias de origen y destino contienen la misma clave, la clave de la instancia de destino se sobrescribirá después de la migración.

Paso 3: Comprobar la red

Paso 1 Compruebe si la instancia de Redis de origen, la instancia de Redis de destino y la tarea de migración están configuradas con la misma VPC.

En caso afirmativo, vaya a **Paso 4: Crear una tarea de migración en línea**. En caso negativo, vaya a **Paso 2**.

Paso 2 Compruebe si las VPC configuradas para la instancia de Redis de origen, la instancia de Redis de destino y la tarea de migración están conectadas para asegurarse de que el recurso de VM de la tarea de migración puede acceder a las instancias de Redis de origen y destino.

En caso afirmativo, vaya a **Paso 4: Crear una tarea de migración en línea**. En caso negativo, vaya a **Paso 3**.

Paso 3 Realice las siguientes operaciones para establecer la red.

- Si las instancias de origen y destino de Redis están en la misma región, cree una conexión de interconexión de VPC haciendo referencia a **Interconexión de VPC**.
- Si las instancias de origen y destino de Redis se encuentran en diferentes regiones, cree una conexión a la nube consultando **Introducción a Cloud Connect**.

- Si las instancias de origen y destino de Redis están en nubes diferentes, cree una conexión consultando la **documentación de Direct Connect**.

----Fin

Paso 4: Crear una tarea de migración en línea

Paso 1 Iniciar sesión en la consola de DCS.

Paso 2 En el panel de navegación, elija **Data Migration**.

Paso 3 Haga clic en **Create Online Migration Task**.

Paso 4 Introduzca el nombre y la descripción de la tarea.

Paso 5 Configure la VPC, la subred y el grupo de seguridad para la tarea de migración.

La VPC, la subred y el grupo de seguridad facilitan la migración. Asegúrese de que los recursos de migración puedan acceder a las instancias de Redis de origen y destino.

----Fin

Paso 5: Configurar la tarea de migración en línea

Paso 1 En la página de ficha **Online Migration**, haga clic en **Configure** en la fila que contiene la tarea de migración en línea que acaba de crear.

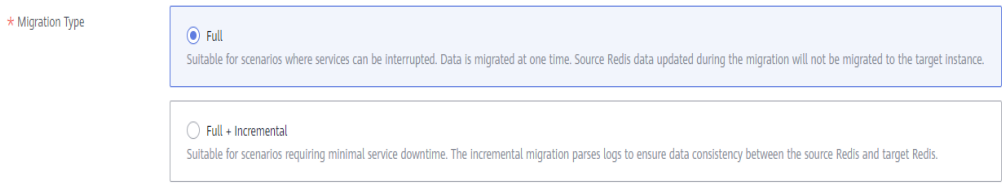
Paso 2 Seleccione un tipo de migración.

Los tipos de migración admitidos son **Full** y **Full + Incremental**, que se describen en **Tabla 9-2**.

Tabla 9-2 Descripción del tipo de migración

Tipo de migración	Descripción
Full	Adecuado para escenarios en los que los servicios pueden ser interrumpidos. Los datos se migran al mismo tiempo. Los datos de instancia de origen actualizados durante la migración no se migrarán a la instancia de destino.
Full + incremental	Adecuado para escenarios que requieren un mínimo tiempo de inactividad del servicio. La migración incremental analiza los registros para garantizar la coherencia de los datos entre las instancias de origen y destino. Una vez que se inicie la migración, seguirá **Migrating** hasta que haga clic en **Stop** en la columna **Operation**. Después de detener la migración, los datos de la instancia de origen no se perderán, pero los datos no se escribirán en la instancia de destino. Cuando la red de transmisión es estable, el retardo de la migración incremental es en segundos. El retardo real depende de la calidad de transmisión del enlace de red.

Figura 9-3 Selección del tipo de migración



Paso 3 Configurar Redis de origen y Redis de destino.

1. **Source Redis Type**: seleccione **Redis in the cloud** o **Self-hosted Redis** según sea necesario.
   - **Redis in the cloud**: una instancia de Huawei Cloud DCS Redis que se encuentra en la misma VPC que la tarea de migración
   - **Self-hosted Redis**: Redis autohospedado en Huawei Cloud, en otra nube o en centros de datos locales. Si selecciona esta opción, introduzca las direcciones de Redis.
2. Si la instancia está protegida con contraseña, puede hacer clic en **Test Connection** para comprobar si la contraseña de la instancia es correcta y si la red está conectada.

Paso 4 Para **Target Redis Instance**, seleccione la instancia de Redis de DCS preparada en **Paso 2: Preparar la instancia de DCS Redis de destino**.

Si la instancia está protegida con contraseña, puede hacer clic en **Test Connection** para comprobar si la contraseña de la instancia cumple los requisitos.

NOTA

Si las instancias de Redis de origen y destino están conectadas pero se encuentran en diferentes regiones de Huawei Cloud, solo puede seleccionar **Self-hosted Redis** para **Target Redis Type** e introducir las direcciones de instancia, independientemente de si la instancia de Redis de destino está autohospedada o en la nube.

Paso 5 Confirme los detalles de la tarea de migración y haga clic en **Submit**.

Vuelva a la lista de tareas de migración de datos. Una vez que la migración se realiza correctamente, el estado de la tarea cambia a **Successful**.

NOTA

Una vez que se inicia la migración incremental, seguirá **Migrating** hasta que haga clic en **Stop**.

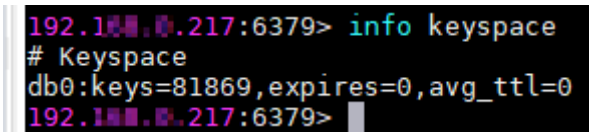
Si la migración falla, haga clic en la tarea de migración y compruebe el registro en la página **Migration Logs**.

----Fin

Verificación de la migración

Una vez completada la migración, utilice redis-cli para conectar las instancias de Redis de origen y destino para comprobar la integridad de los datos.

1. Conéctese al Redis de origen y al Redis de destino.
2. Ejecute el comando **info keyspace** para comprobar los valores de **keys** y **expires**.



3. Calcular la diferencia entre los valores de **keys** y **expires**. del Redis de origen y el Redis de destino. Si las diferencias son las mismas, los datos están completos y la migración se realiza correctamente.

Durante la migración completa, los datos de origen de Redis actualizados durante la migración no se migrarán a la instancia de destino.

9.5 Conmutación de IP

Caso

Actualmente, no se puede cambiar el tipo de instancia cuando se utiliza la función de modificación de especificación. Para modificar las especificaciones de instancia mientras se cambia el tipo de instancia, puede realizar el cambio de IP después de la migración de datos. Al cambiar las direcciones IP, también puede cambiar la arquitectura AZ y CPU utilizada por una instancia.

- Una vez completada la migración de datos en línea, puede cambiar las direcciones IP.
- Las direcciones IP se pueden revertir según sea necesario después de la conmutación.

NOTA

- La conmutación IP solo es compatible con las instancias de DCS Redis 4.0 y 5.0.
- La conmutación IP solo se admite cuando las instancias de origen y destino son instancias de Redis en la nube.

Prerrequisitos:

- Obtenga información sobre las instancias de origen y destino. Para obtener más información sobre cómo preparar una instancia de destino, consulte **Paso 2: Preparar la instancia de DCS Redis de destino**.
- Asegúrese de que las instancias de origen y destino puedan comunicarse entre sí. Para más detalles, consulte **Paso 3: Comprobar la red**.
- Las instancias de destino y origen deben utilizar el mismo puerto.
- La conmutación IP sólo se puede realizar cuando se cumplen las siguientes condiciones:
  - La conmutación IP depende de la función de migración de datos. Por lo tanto, las instancias de origen y destino deben admitir la función de migración de datos. Para más detalles, consulte **Tabla 9-1**.
  - Tanto las instancias de origen como de destino son instancias de Redis en la nube.
  - **Tabla 9-3** enumera los escenarios de conmutación IP admitidos.

Tabla 9-3 Escenarios de conmutación de IP

Origen	Objetivo
Nodo único, principal/en standby o separación de lectura/escritura	Nodo único, principal/en standby, separación de lectura/escritura o Clúster Proxy
Clúster Proxy	Nodo único, principal/en standby, separación de lectura/escritura o Clúster Proxy

Precauciones para la conmutación de IP

1. La migración en línea se detendrá durante la conmutación.
2. Las instancias serán de solo lectura durante un minuto y se desconectarán durante varios segundos durante la conmutación.
3. Si la aplicación no puede volver a conectarse a Redis o manejar excepciones, es posible que tenga que reiniciar la aplicación después de la conmutación de IP.
4. Si las instancias de origen y destino se encuentran en diferentes subredes, la información de la subred se actualizará después de la conmutación.
5. Si la fuente es una instancia principal/en standby, la dirección IP del nodo en standby no se conmutará. Asegúrese de que sus aplicaciones no utilicen esta dirección IP.
6. Si sus aplicaciones usan un nombre de dominio para conectarse a Redis, el nombre de dominio se usará para la instancia de origen. Seleccione **Yes** para **Switch Domain Name**.
7. Asegúrese de que las contraseñas de las instancias de origen y de destino sean las mismas. Si son diferentes, la verificación fallará después de la conmutación.
8. Si se configura una lista blanca para la instancia de origen, asegúrese de que la misma lista blanca está configurada para la instancia de destino antes de cambiar las direcciones IP.

Conmutación de direcciones IP

Paso 1 Inicie sesión en la **consola DCS**..

Paso 2 Haga clic en en la esquina superior izquierda de la consola de gestión y seleccione la región donde se encuentra la instancia.

Paso 3 En el panel de navegación, elija **Data Migration**.

Paso 4 Haga clic en **Create Online Migration Task**.

Paso 5 Introduzca el nombre y la descripción de la tarea.

Paso 6 Configure la VPC, la subred y el grupo de seguridad para la tarea de migración.

La VPC, la subred y el grupo de seguridad facilitan la migración. Asegúrese de que los recursos de migración puedan acceder a las instancias de Redis de origen y destino.

Paso 7 Configure la tarea de migración haciendo referencia a **Configuración de la tarea de migración en línea**. Establezca **Migration Type** en **Full + Incremental**.

Paso 8 En la página **Online Migration**, cuando el estado de la tarea de migración cambie a **Incremental migration in progress**, elija **More** > **Switch IP** en la columna **Operation**.

Paso 9 En el cuadro de diálogo **Switch IP**, seleccione si desea cambiar el nombre de dominio.

NOTA

- Si se utiliza un nombre de dominio, cámbielo o debe modificar el nombre de dominio en el cliente.
- Si no se utiliza ningún nombre de dominio, se actualizará el DNS de las instancias.

Paso 10 Haz clic en **OK**. La tarea de conmutación de direcciones IP se envía correctamente. Cuando el estado de la tarea de migración cambia a **IP switched**, se completa el cambio de dirección IP.

----Fin

Retroceder las direcciones IP

Si desea cambiar la dirección IP de la instancia a la dirección IP original, realice las siguientes operaciones:

Paso 1 Inicie sesión en la **consola DCS**..

Paso 2 Haga clic en  en la esquina superior izquierda de la consola de gestión y seleccione la región donde se encuentra la instancia.

Paso 3 En el panel de navegación, elija **Data Migration**.

Paso 4 En la página **Online Migration**, busque la fila que contiene la tarea de migración en el estado **IP switched**, elija **More** > **Roll Back IP**.

Paso 5 En la ventana de confirmación, haga clic en **Yes**. La tarea de reversión de direcciones IP se envía correctamente. Cuando el estado de la tarea cambia a **IP rolled back**, se completa la reversión.

----Fin

10 Plantillas de parámetros

10.1 Consulta de plantillas de parámetros

Esta sección describe cómo ver las plantillas de parámetro en la consola DCS.

Procedimiento

Paso 1 Iniciar sesión en la consola de DCS.

Paso 2 Haga clic en  en la esquina superior izquierda de la consola de gestión y seleccione la región donde se encuentra la instancia.

Paso 3 En el panel de navegación, elija **Parameter Templates**.

Paso 4 Elija la ficha **Default Templates** o **Custom Templates**.

Paso 5 Ver plantillas de parámetros

Actualmente, puede introducir una palabra clave en el cuadro de búsqueda para buscar una plantilla de parámetro por su nombre.

Paso 6 Haga clic en una plantilla de parámetro. Se muestran los parámetros contenidos en la plantilla. Para obtener detalles sobre los parámetros, consulte **Tabla 10-1**.

Tabla 10-1 Parámetros de configuración de instancia de DCS Redis

Parámetro	Descripción	Rango de valores	Valor predeterminado
timeout	La cantidad máxima de tiempo (en segundos) que se puede permitir que una conexión entre un cliente y la instancia de DCS permanezca inactiva antes de que finalice la conexión. Un ajuste de **0** significa que esta función está deshabilitada.	0–7200 segundos	0
appendfsync	Controla la frecuencia con la que fsync () transfiere datos almacenados en la memoria caché al disco. Tenga en cuenta que algunos SO realizarán una transferencia de datos completa, pero otros solo hacen un intento de "mejor esfuerzo". Hay tres configuraciones: no: nunca se llama a fsync(). El SO descargará los datos cuando esté listo. Este modo ofrece el máximo rendimiento. always: fsync() se llama después de cada escritura en el AOF. Este modo es muy lento, pero también muy seguro. everysec: se llama a fsync() una vez por segundo. Este modo proporciona un compromiso entre seguridad y rendimiento.	● no ● always ● everysec	everysec

Parámetro	Descripción	Rango de valores	Valor predeterminado
appendonly	Indica si se deben registrar todas las modificaciones de la instancia. Por defecto, los datos se escriben en discos de forma asíncrona en Redis. Si esta función está deshabilitada, los datos generados recientemente podrían perderse en el caso de un corte de energía. Opciones: yes: Los registros están habilitados, es decir, la persistencia está habilitada. no: Los registros están deshabilitados, es decir, la persistencia está deshabilitada.	● yes ● no	yes
client-output-buffer-limit-slave-soft-seconds	Número de segundos que el búfer de salida permanece por encima de **client-output-buffer-slave-soft-limit** antes de que el cliente se desconecte.	0–60	60
client-output-buffer-slave-hard-limit	Límite invariable (en bytes) en el búfer de salida de los clientes de réplica. Una vez que el búfer de salida excede el límite invariable, el cliente se desconecta inmediatamente.	0–17,179,869,184	1,717,986,918
client-output-buffer-slave-soft-limit	Límite flexible (en bytes) en el búfer de salida de los clientes de réplica. Una vez que el búfer de salida excede el límite flexible y permanece continuamente por encima del límite durante el período especificado por el parámetro **client-output-buffer-limit-slave-soft-seconds**, el cliente se desconecta.	0–17,179,869,184	1,717,986,918

Parámetro	Descripción	Rango de valores	Valor predeterminado
maxmemory-policy	La política aplicada cuando se alcanza el límite maxmemory. Para obtener más información acerca de este parámetro, vea **https://docs.redis.com/latest/rs/databases/memory-performance/eviction-policy/**.	● volatile-lru ● allkeys-lru ● volatile-random ● allkeys-random ● volatile-ttl ● noeviction	volatile-lru **NOTA** Si la instancia de DCS Redis se creó antes de julio de 2020 y este parámetro no se ha modificado, el valor predeterminado es **noeviction**. Si la instancia se crea después de julio de 2020, el valor predeterminado es **volatile-lru**.
lua-time-limit	Tiempo máximo permitido para la ejecución de un script Lua (en milisegundos)	100–5000	5000
master-read-only	Configura la instancia como de solo lectura. No se podrá ejecutar ninguna operación de escritura.	● yes ● no	no
maxclients	Cantidad máxima de clientes que se puede conectar de forma simultánea a una instancia de DCS.	1000–50,000	10,000
proto-max-bulk-len	Tamaño máximo de una solicitud de elemento único (en bytes).	1,048,576–536,870,912	536,870,912

Parámetro	Descripción	Rango de valores	Valor predeterminado
repl-backlog-size	Tamaño del backlog de replicación (bytes). El backlog es un búfer que acumula datos de réplicas cuando estas se desconectan de la instancia principal. Cuando una réplica se vuelve a conectar, se realiza una sincronización parcial para sincronizar los datos que se perdieron mientras las réplicas estuvieron desconectadas.	16,384–1,073,741,824	1,048,576
repl-backlog-ttl	La cantidad de tiempo, en segundos, antes de que se libere el búfer de backlog, calculada a partir de la última que se desconectó una réplica. El valor **0** indica que el backlog nunca se libera.	0–604,800	3600
repl-timeout	Fin de tiempo de espera de la replicación (en segundos).	30–3600	60
hash-max-ziplist-entries	Número máximo de hashes que se pueden codificar mediante ziplist, una estructura de datos optimizada para reducir el uso de memoria.	1–10,000	512
hash-max-ziplist-value	El valor más grande permitido para un hash codificado usando ziplist, una estructura de datos especial optimizada para el uso de memoria.	1–10,000	64

Parámetro	Descripción	Rango de valores	Valor predeterminado
set-max-intset-entries	Si un conjunto se compone únicamente de cadenas de caracteres que son números enteros en base 10 dentro del rango de números enteros con signo de 64 bits, el conjunto se codifica usando intset, una estructura de datos optimizada para el uso de memoria.	1–10,000	512
zset-max-ziplist-entries	Número máximo de conjuntos ordenados que se pueden codificar mediante ziplist, una estructura de datos optimizada para reducir el uso de memoria.	1–10,000	128
zset-max-ziplist-value	El valor más grande permitido para un conjunto ordenado codificado usando ziplist, una estructura de datos especial optimizada para el uso de memoria.	1–10,000	64

Parámetro	Descripción	Rango de valores	Valor predeterminado
latency-monitor-threshold	La cantidad mínima de latencia que se registrará como picos de latencia ● Establecer en 0: La supervisión de la latencia está deshabilitada. ● Establecer a más de 0: Todos con al menos este número de ms de latencia se registrarán. Al ejecutar el comando LATENCY, puede realizar operaciones relacionadas con el monitoreo de latencia, como la obtención de datos estadísticos y la configuración y habilitación del monitoreo de latencia. Para obtener más información acerca de la latencia-monitor-threshold, visite **https://redis.io/docs/reference/optimization/latency-monitor/**.	0–86,400,000 ms	0

Parámetro	Descripción	Rango de valores	Valor predeterminado
notify-keyspace-events	Controla qué tipo de notificaciones están habilitadas para los eventos de espacio de claves. Si se configura este parámetro, la función Redis Pub/Sub permitirá a los clientes recibir una notificación de evento cuando se modifique un conjunto de datos Redis. Las instancias de Clúster Proxy no tienen este parámetro.	Se puede usar una combinación de diferentes valores para habilitar notificaciones para varios tipos de eventos. Los valores posibles incluyen: K: Eventos de Keyspace, publicados con el __keyspace@__ prefix E: Eventos de Keyevent, publicados con __keyevent@__ prefix Comandos genéricos (sin un tipo específico) como DEL, EXPIRE y RENAME: $: Comandos de cadena l: Lista de comandos s: Establecer comandos h: Comandos hash z: Comandos de conjunto ordenado x: Eventos expirados (eventos generados cada vez que expira una clave) e: Eventos desalojados (eventos generados cuando una clave es desalojada de maxmemory) Para obtener más información, consulte la siguiente nota.	Ex
slowlog-log-slower-than	La cantidad máxima de tiempo permitido, en microsegundos, para la ejecución de comandos. Si se supera este umbral, el registro de consultas lentas de Redis registrará el comando.	0–1,000,000	10,000

Parámetro	Descripción	Rango de valores	Valor predeterminado
slowlog-max-len	Número máximo permitido de consultas lentas que se pueden registrar. El registro de consultas lento consume memoria, pero puede recuperar esta memoria ejecutando el comando **SLOWLOG RESET**.	0–1000	128

NOTA

1. Los valores predeterminados y los rangos de valores de los parámetros **maxclients**, **reserved-memory-percent**, **client-output-buffer-slave-soft-limit**, y **client-output-buffer-slave-hard-limit** están relacionados con las especificaciones de instancia. Por lo tanto, estos parámetros no se muestran en la plantilla de parámetro.
2. Para obtener más información acerca de los parámetros descritos en **Tabla 10-1**, visite **https://redis.io/topics/memory-optimization**.

----Fin

10.2 Creación de una plantilla de parámetros personalizada

Puede crear las plantillas de parámetros personalizadas para diferentes versiones del motor de caché y tipos de instancia según los requisitos del servicio.

Procedimiento

Paso 1 Iniciar sesión en la consola de DCS.

Paso 2 Haga clic en  en la esquina superior izquierda de la consola de gestión y seleccione la región donde se encuentra la instancia.

Paso 3 En el panel de navegación, elija **Parameter Templates**.

Paso 4 Haga clic en la ficha **Default Templates** o **Custom Templates** para crear una plantilla basada en una plantilla predeterminada o una plantilla personalizada existente.

- Si selecciona **Default Templates**, haga clic en **Customize** en la columna **Operation** de la fila que contiene la versión del motor de caché deseada.
- Si selecciona **Custom Templates**, haga clic en **Copy** en la columna **Operation** de la fila que contiene la plantilla personalizada deseada.

Paso 5 Especifique **Template Name** y **Description**.

> **NOTA**
>
> El nombre de la plantilla puede contener de 4 a 64 caracteres y debe comenzar con una letra o un dígito. Solo se permiten letras, dígitos, guiones medios (-), guiones bajos (_) y puntos (.). La descripción puede estar vacía.

Paso 6 Seleccione **Modifiable parameters**.

Actualmente, puede introducir una palabra clave en el cuadro de búsqueda para buscar un parámetro por nombre de parámetro.

Paso 7 En la fila que contiene el parámetro que se va a modificar, introduzca un valor en la columna **Assigned Value**.

Tabla 10-2 describe los parámetros. En la mayoría de los casos, se conservan los valores predeterminados.

Tabla 10-2 Parámetros de configuración de instancia de DCS Redis

Parámetro	Descripción	Rango de valores	Valor predeterminado
timeout	La cantidad máxima de tiempo (en segundos) que se puede permitir que una conexión entre un cliente y la instancia de DCS permanezca inactiva antes de que finalice la conexión. Un ajuste de **0** significa que esta función está deshabilitada.	0–7200 segundos	0

Parámetro	Descripción	Rango de valores	Valor predeterminado
appendfsync	Controla la frecuencia con la que fsync () transfiere datos almacenados en la memoria caché al disco. Tenga en cuenta que algunos SO realizarán una transferencia de datos completa, pero otros solo hacen un intento de "mejor esfuerzo". Hay tres configuraciones: no: nunca se llama a fsync(). El SO descargará los datos cuando esté listo. Este modo ofrece el máximo rendimiento. always: fsync() se llama después de cada escritura en el AOF. Este modo es muy lento, pero también muy seguro. everysec: se llama a fsync() una vez por segundo. Este modo proporciona un compromiso entre seguridad y rendimiento.	● no ● always ● everysec	everysec
appendonly	Indica si se deben registrar todas las modificaciones de la instancia. Por defecto, los datos se escriben en discos de forma asíncrona en Redis. Si esta función está deshabilitada, los datos generados recientemente podrían perderse en el caso de un corte de energía. Opciones: yes: Los registros están habilitados, es decir, la persistencia está habilitada. no: Los registros están deshabilitados, es decir, la persistencia está deshabilitada.	● yes ● no	yes

Parámetro	Descripción	Rango de valores	Valor predeterminado
client-output-buffer-limit-slave-soft-seconds	Número de segundos que el búfer de salida permanece por encima de **client-output-buffer-slave-soft-limit** antes de que el cliente se desconecte.	0–60	60
client-output-buffer-slave-hard-limit	Límite invariable (en bytes) en el búfer de salida de los clientes de réplica. Una vez que el búfer de salida excede el límite invariable, el cliente se desconecta inmediatamente.	0–17,179,869,184	1,717,986,918
client-output-buffer-slave-soft-limit	Límite flexible (en bytes) en el búfer de salida de los clientes de réplica. Una vez que el búfer de salida excede el límite flexible y permanece continuamente por encima del límite durante el período especificado por el parámetro **client-output-buffer-limit-slave-soft-seconds**, el cliente se desconecta.	0–17,179,869,184	1,717,986,918

Parámetro	Descripción	Rango de valores	Valor predeterminado
maxmemory-policy	La política aplicada cuando se alcanza el límite maxmemory. Para obtener más información acerca de este parámetro, vea **https://docs.redis.com/latest/rs/databases/memory-performance/eviction-policy/**.	● volatile-lru ● allkeys-lru ● volatile-random ● allkeys-random ● volatile-ttl ● noeviction	volatile-lru **NOTA** Si la instancia de DCS Redis se creó antes de julio de 2020 y este parámetro no se ha modificado, el valor predeterminado es **noeviction**. Si la instancia se crea después de julio de 2020, el valor predeterminado es **volatile-lru**.
lua-time-limit	Tiempo máximo permitido para la ejecución de un script Lua (en milisegundos)	100–5000	5000
master-read-only	Configura la instancia como de solo lectura. No se podrá ejecutar ninguna operación de escritura.	● yes ● no	no
maxclients	Cantidad máxima de clientes que se puede conectar de forma simultánea a una instancia de DCS.	1000–50,000	10,000
proto-max-bulk-len	Tamaño máximo de una solicitud de elemento único (en bytes).	1,048,576–536,870,912	536,870,912

Parámetro	Descripción	Rango de valores	Valor predeterminado
repl-backlog-size	Tamaño del backlog de replicación (bytes). El backlog es un búfer que acumula datos de réplicas cuando estas se desconectan de la instancia principal. Cuando una réplica se vuelve a conectar, se realiza una sincronización parcial para sincronizar los datos que se perdieron mientras las réplicas estuvieron desconectadas.	16,384–1,073,741,824	1,048,576
repl-backlog-ttl	La cantidad de tiempo, en segundos, antes de que se libere el búfer de backlog, calculada a partir de la última que se desconectó una réplica. El valor **0** indica que el backlog nunca se libera.	0–604,800	3600
repl-timeout	Fin de tiempo de espera de la replicación (en segundos).	30–3600	60
hash-max-ziplist-entries	Número máximo de hashes que se pueden codificar mediante ziplist, una estructura de datos optimizada para reducir el uso de memoria.	1–10,000	512
hash-max-ziplist-value	El valor más grande permitido para un hash codificado usando ziplist, una estructura de datos especial optimizada para el uso de memoria.	1–10,000	64

Parámetro	Descripción	Rango de valores	Valor predeterminado
set-max-intset-entries	Si un conjunto se compone únicamente de cadenas de caracteres que son números enteros en base 10 dentro del rango de números enteros con signo de 64 bits, el conjunto se codifica usando intset, una estructura de datos optimizada para el uso de memoria.	1–10,000	512
zset-max-ziplist-entries	Número máximo de conjuntos ordenados que se pueden codificar mediante ziplist, una estructura de datos optimizada para reducir el uso de memoria.	1–10,000	128
zset-max-ziplist-value	El valor más grande permitido para un conjunto ordenado codificado usando ziplist, una estructura de datos especial optimizada para el uso de memoria.	1–10,000	64

Parámetro	Descripción	Rango de valores	Valor predeterminado
latency-monitor-threshold	La cantidad mínima de latencia que se registrará como picos de latencia ● Establecer en 0: La supervisión de la latencia está deshabilitada. ● Establecer a más de 0: Todos con al menos este número de ms de latencia se registrarán. Al ejecutar el comando LATENCY, puede realizar operaciones relacionadas con el monitoreo de latencia, como la obtención de datos estadísticos y la configuración y habilitación del monitoreo de latencia. Para obtener más información acerca de la latencia-monitor-threshold, visite **https://redis.io/docs/reference/optimization/latency-monitor/**.	0–86,400,000 ms	0

Parámetro	Descripción	Rango de valores	Valor predeterminado
notify-keyspace-events	Controla qué tipo de notificaciones están habilitadas para los eventos de espacio de claves. Si se configura este parámetro, la función Redis Pub/Sub permitirá a los clientes recibir una notificación de evento cuando se modifique un conjunto de datos Redis. Las instancias de Clúster Proxy no tienen este parámetro.	Se puede usar una combinación de diferentes valores para habilitar notificaciones para varios tipos de eventos. Los valores posibles incluyen: K: Eventos de Keyspace, publicados con el __keyspace@__ prefix E: Eventos de Keyevent, publicados con __keyevent@__ prefix Comandos genéricos (sin un tipo específico) como DEL, EXPIRE y RENAME: $: Comandos de cadena l: Lista de comandos s: Establecer comandos h: Comandos hash z: Comandos de conjunto ordenado x: Eventos expirados (eventos generados cada vez que expira una clave) e: Eventos desalojados (eventos generados cuando una clave es desalojada de maxmemory) Para obtener más información, consulte la siguiente nota.	Ex
slowlog-log-slower-than	La cantidad máxima de tiempo permitido, en microsegundos, para la ejecución de comandos. Si se supera este umbral, el registro de consultas lentas de Redis registrará el comando.	0–1,000,000	10,000

Parámetro	Descripción	Rango de valores	Valor predeterminado
slowlog-max-len	Número máximo permitido de consultas lentas que se pueden registrar. El registro de consultas lento consume memoria, pero puede recuperar esta memoria ejecutando el comando **SLOWLOG RESET**.	0–1000	128

NOTA

1. Los valores predeterminados y los rangos de valores de los parámetros **maxclients**, **reserved-memory-percent**, **client-output-buffer-slave-soft-limit**, y **client-output-buffer-slave-hard-limit** están relacionados con las especificaciones de instancia. Por lo tanto, estos parámetros no se pueden modificar.
2. Para obtener más información acerca de los parámetros descritos en **Tabla 10-2**, visite **https://redis.io/topics/memory-optimization**.
3. El parámetro **latency-monitor-threshold** se utiliza normalmente para la localización de fallos. Después de localizar fallos basados en la información de latencia recopilada, cambie el valor de **latency-monitor-threshold** a **0** para evitar latencia innecesaria.
4. Más información sobre el parámetro **notify-keyspace-events**:
   - La configuración del parámetro debe contener al menos una K o una E.
   - A es un alias para "g$lshzxe" y no se puede usar junto con ninguno de los caracteres en "g $lshzxe".
   - Por ejemplo, el valor **Kl** significa que Redis notificará a los clientes de Pub/Sub acerca de los eventos de espacio de claves y los comandos de lista. El valor **AKE** significa que Redis notificará a los clientes Pub/Sub sobre todos los eventos.

Paso 8 Haz clic en **OK**.

----Fin

10.3 Modificación de una plantilla de parámetros personalizada

Puede modificar el nombre, la descripción y los parámetros de una plantilla de parámetros personalizada según los requisitos de servicio.

Procedimiento

Paso 1 Iniciar sesión en la consola de DCS.

Paso 2 Haga clic en  en la esquina superior izquierda de la consola de gestión y seleccione la región donde se encuentra la instancia.

Paso 3 En el panel de navegación, elija **Parameter Templates**.

Paso 4 Elija la ficha **Custom Templates**.

Paso 5 Puede modificar una plantilla de parámetros personalizada de cualquiera de las siguientes maneras:

- Haga clic en **Edit** en la columna **Operation**.
  a. Cambie el nombre o modifique la descripción de una plantilla.
  b. En el área **Parameters**, seleccione **Modifiable parameters**. En la fila que contiene el parámetro que se va a modificar, introduzca un valor en la columna **Assigned Value**. **Tabla 10-3** describe los parámetros. En la mayoría de los casos, se conservan los valores predeterminados.
  c. Haz clic en **OK**.
- Haga clic en el nombre de una plantilla personalizada. En la página mostrada, modifique los parámetros.
  a. Seleccione **Modifiable parameters**. Introduzca una palabra clave en el cuadro de búsqueda para buscar un parámetro por nombre de parámetro.
  b. Haga clic en **Modify**.
  c. En la fila que contiene el parámetro que se va a modificar, introduzca un valor en la columna **Assigned Value**. **Tabla 10-3** describe los parámetros. En la mayoría de los casos, se conservan los valores predeterminados.
  d. Haga clic en **Save**.

Tabla 10-3 Parámetros de configuración de instancia de DCS Redis

Parámetro	Descripción	Rango de valores	Valor predeterminado
timeout	La cantidad máxima de tiempo (en segundos) que se puede permitir que una conexión entre un cliente y la instancia de DCS permanezca inactiva antes de que finalice la conexión. Un ajuste de **0** significa que esta función está deshabilitada.	0–7200 segundos	0

Parámetro	Descripción	Rango de valores	Valor predeterminado
appendfsync	Controla la frecuencia con la que fsync () transfiere datos almacenados en la memoria caché al disco. Tenga en cuenta que algunos SO realizarán una transferencia de datos completa, pero otros solo hacen un intento de "mejor esfuerzo". Hay tres configuraciones: no: nunca se llama a fsync(). El SO descargará los datos cuando esté listo. Este modo ofrece el máximo rendimiento. always: fsync() se llama después de cada escritura en el AOF. Este modo es muy lento, pero también muy seguro. everysec: se llama a fsync() una vez por segundo. Este modo proporciona un compromiso entre seguridad y rendimiento.	● no ● always ● everysec	everysec
appendonly	Indica si se deben registrar todas las modificaciones de la instancia. Por defecto, los datos se escriben en discos de forma asíncrona en Redis. Si esta función está deshabilitada, los datos generados recientemente podrían perderse en el caso de un corte de energía. Opciones: yes: Los registros están habilitados, es decir, la persistencia está habilitada. no: Los registros están deshabilitados, es decir, la persistencia está deshabilitada.	● yes ● no	yes

Parámetro	Descripción	Rango de valores	Valor predeterminado
client-output-buffer-limit-slave-soft-seconds	Número de segundos que el búfer de salida permanece por encima de **client-output-buffer-slave-soft-limit** antes de que el cliente se desconecte.	0–60	60
client-output-buffer-slave-hard-limit	Límite invariable (en bytes) en el búfer de salida de los clientes de réplica. Una vez que el búfer de salida excede el límite invariable, el cliente se desconecta inmediatamente.	0–17,179,869,184	1,717,986,918
client-output-buffer-slave-soft-limit	Límite flexible (en bytes) en el búfer de salida de los clientes de réplica. Una vez que el búfer de salida excede el límite flexible y permanece continuamente por encima del límite durante el período especificado por el parámetro **client-output-buffer-limit-slave-soft-seconds**, el cliente se desconecta.	0–17,179,869,184	1,717,986,918

Parámetro	Descripción	Rango de valores	Valor predeterminado
maxmemory-policy	La política aplicada cuando se alcanza el límite maxmemory. Para obtener más información acerca de este parámetro, vea **https://docs.redis.com/latest/rs/databases/memory-performance/eviction-policy/**.	● volatile-lru ● allkeys-lru ● volatile-random ● allkeys-random ● volatile-ttl ● noeviction	volatile-lru **NOTA** Si la instancia de DCS Redis se creó antes de julio de 2020 y este parámetro no se ha modificado, el valor predeterminado es **noeviction**. Si la instancia se crea después de julio de 2020, el valor predeterminado es **volatile-lru**.
lua-time-limit	Tiempo máximo permitido para la ejecución de un script Lua (en milisegundos)	100–5000	5000
master-read-only	Configura la instancia como de solo lectura. No se podrá ejecutar ninguna operación de escritura.	● yes ● no	no
maxclients	Cantidad máxima de clientes que se puede conectar de forma simultánea a una instancia de DCS.	1000–50,000	10,000
proto-max-bulk-len	Tamaño máximo de una solicitud de elemento único (en bytes).	1,048,576–536,870,912	536,870,912

Parámetro	Descripción	Rango de valores	Valor predeterminado
repl-backlog-size	Tamaño del backlog de replicación (bytes). El backlog es un búfer que acumula datos de réplicas cuando estas se desconectan de la instancia principal. Cuando una réplica se vuelve a conectar, se realiza una sincronización parcial para sincronizar los datos que se perdieron mientras las réplicas estuvieron desconectadas.	16,384–1,073,741,824	1,048,576
repl-backlog-ttl	La cantidad de tiempo, en segundos, antes de que se libere el búfer de backlog, calculada a partir de la última que se desconectó una réplica. El valor **0** indica que el backlog nunca se libera.	0–604,800	3600
repl-timeout	Fin de tiempo de espera de la replicación (en segundos).	30–3600	60
hash-max-ziplist-entries	Número máximo de hashes que se pueden codificar mediante ziplist, una estructura de datos optimizada para reducir el uso de memoria.	1–10,000	512
hash-max-ziplist-value	El valor más grande permitido para un hash codificado usando ziplist, una estructura de datos especial optimizada para el uso de memoria.	1–10,000	64

Parámetro	Descripción	Rango de valores	Valor predeterminado
set-max-intset-entries	Si un conjunto se compone únicamente de cadenas de caracteres que son números enteros en base 10 dentro del rango de números enteros con signo de 64 bits, el conjunto se codifica usando intset, una estructura de datos optimizada para el uso de memoria.	1–10,000	512
zset-max-ziplist-entries	Número máximo de conjuntos ordenados que se pueden codificar mediante ziplist, una estructura de datos optimizada para reducir el uso de memoria.	1–10,000	128
zset-max-ziplist-value	El valor más grande permitido para un conjunto ordenado codificado usando ziplist, una estructura de datos especial optimizada para el uso de memoria.	1–10,000	64

Parámetro	Descripción	Rango de valores	Valor predeterminado
latency-monitor-threshold	La cantidad mínima de latencia que se registrará como picos de latencia ● Establecer en 0: La supervisión de la latencia está deshabilitada. ● Establecer a más de 0: Todos con al menos este número de ms de latencia se registrarán. Al ejecutar el comando LATENCY, puede realizar operaciones relacionadas con el monitoreo de latencia, como la obtención de datos estadísticos y la configuración y habilitación del monitoreo de latencia. Para obtener más información acerca de la latencia-monitor-threshold, visite **https://redis.io/docs/reference/optimization/latency-monitor/**.	0–86,400,000 ms	0

Parámetro	Descripción	Rango de valores	Valor predeterminado
notify-keyspace-events	Controla qué tipo de notificaciones están habilitadas para los eventos de espacio de claves. Si se configura este parámetro, la función Redis Pub/Sub permitirá a los clientes recibir una notificación de evento cuando se modifique un conjunto de datos Redis. Las instancias de Clúster Proxy no tienen este parámetro.	Se puede usar una combinación de diferentes valores para habilitar notificaciones para varios tipos de eventos. Los valores posibles incluyen: K: Eventos de Keyspace, publicados con el __keyspace@__ prefix E: Eventos de Keyevent, publicados con __keyevent@__ prefix Comandos genéricos (sin un tipo específico) como DEL, EXPIRE y RENAME: $: Comandos de cadena l: Lista de comandos s: Establecer comandos h: Comandos hash z: Comandos de conjunto ordenado x: Eventos expirados (eventos generados cada vez que expira una clave) e: Eventos desalojados (eventos generados cuando una clave es desalojada de maxmemory) Para obtener más información, consulte la siguiente nota.	Ex
slowlog-log-slower-than	La cantidad máxima de tiempo permitido, en microsegundos, para la ejecución de comandos. Si se supera este umbral, el registro de consultas lentas de Redis registrará el comando.	0–1,000,000	10,000

Parámetro	Descripción	Rango de valores	Valor predeterminado
slowlog-max-len	Número máximo permitido de consultas lentas que se pueden registrar. El registro de consultas lento consume memoria, pero puede recuperar esta memoria ejecutando el comando **SLOWLOG RESET**.	0–1000	128

NOTA

1. Los valores predeterminados y los rangos de valores de los parámetros **maxclients**, **reserved-memory-percent**, **client-output-buffer-slave-soft-limit**, y **client-output-buffer-slave-hard-limit** están relacionados con las especificaciones de instancia. Por lo tanto, estos parámetros no se pueden modificar.
2. Para obtener más información acerca de los parámetros descritos en **Tabla 10-3**, visite **https://redis.io/topics/memory-optimization**.
3. El parámetro **latency-monitor-threshold** se utiliza normalmente para la localización de fallos. Después de localizar fallos basados en la información de latencia recopilada, cambie el valor de **latency-monitor-threshold** a **0** para evitar latencia innecesaria.
4. Más información sobre el parámetro **notify-keyspace-events**:
   - La configuración del parámetro debe contener al menos una K o una E.
   - A es un alias para "g$lshzxe" y no se puede usar junto con ninguno de los caracteres en "g $lshzxe".
   - Por ejemplo, el valor **Kl** significa que Redis notificará a los clientes de Pub/Sub acerca de los eventos de espacio de claves y los comandos de lista. El valor **AKE** significa que Redis notificará a los clientes Pub/Sub sobre todos los eventos.

----Fin

10.4 Eliminación de una plantilla de parámetros personalizada

En esta sección se describe cómo eliminar una plantilla de parámetros personalizada.

Procedimiento

Paso 1 Iniciar sesión en la consola de DCS.

Paso 2 Haga clic en en la esquina superior izquierda de la consola de gestión y seleccione la región donde se encuentra la instancia.

Paso 3 En el panel de navegación, elija **Parameter Templates**.

Paso 4 Elija la ficha **Custom Templates**.

Paso 5 Haga clic en **Delete** en la columna **Operation**.

Paso 6 Haga clic en **Yes**.

----Fin

11 Gestión de contraseñas

11.1 Contraseñas de instancia de DCS

Las contraseñas se pueden configurar para controlar el acceso a sus instancias DCS, lo que garantiza la seguridad de sus datos.

Puede establecer una contraseña durante o después de la creación de una instancia. Para obtener más información sobre cómo establecer una contraseña después de crear una instancia, consulte **Restablecer contraseñas de instancia**.

Puede elegir si desea habilitar el acceso sin contraseña en función de su seguridad y conveniencia.

Escenarios que requieren contraseñas

- Para una instancia de DCS que se utiliza en la red en vivo o que contiene información importante, se recomienda establecer una contraseña.
- Para una instancia de DCS con acceso público habilitado, se debe establecer una contraseña para garantizar la seguridad de los datos.

Para obtener más información sobre cómo acceder a una instancia con una contraseña, consulte **Acceso a una instancia de DCS**.

Usar contraseñas de forma segura

1. Oculte la contraseña cuando utilice redis-cli.

   Si se utiliza la opción **-a <password>** en redis-cli en Linux, la contraseña es propensa a fugas porque se registra y se mantiene en el historial. Se recomienda no utilizar la opción **-a <password>** al ejecutar comandos en redis-cli. Después de conectarse a Redis, ejecute el comando **auth** para completar la autenticación, como se muestra en el siguiente ejemplo:

   ```
   $ redis-cli -h 192.168.0.148 -p 6379
   redis 192.168.0.148:6379>auth yourPassword
   OK
   redis 192.168.0.148:6379>
   ```

2. Utilice la autenticación de contraseña interactiva o cambie entre usuarios con diferentes permisos.

Si la secuencia de comandos implica acceso a instancia de DCS, utilice la autenticación de contraseña interactiva. Para habilitar la ejecución automática de scripts, administre el script como otro usuario y autorice la ejecución usando sudo.

3. Utilice un módulo de encriptación en su aplicación para cifrar la contraseña.

11.2 Cambio de contraseñas de instancia

En la consola DCS, puede cambiar la contraseña necesaria para acceder a su instancia DCS.

NOTA

- No puede cambiar la contraseña de una instancia de DCS en modo sin contraseña.
- La instancia de DCS para la que desea cambiar la contraseña está en el estado **Running**.
- La nueva contraseña entra en vigor inmediatamente en el servidor sin necesidad de reiniciar. El cliente debe volver a conectarse al servidor con la nueva contraseña después de cerrar una conexión pconnect. (La contraseña antigua aún se puede usar antes de la desconexión.)

Prerrequisitos:

Se ha creado una instancia de DCS.

Procedimiento

Paso 1 Inicie sesión en la **consola DCS**.

Paso 2 Haga clic en en la esquina superior izquierda de la consola de gestión y seleccione la región donde se encuentra la instancia.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Paso 4 Elija **More** > **Change Password** en la fila que contiene la instancia elegida.

Paso 5 En el cuadro de diálogo que se muestra, establezca **Old Password**, **New Password**, y **Confirm Password**.

NOTA

Después de 5 intentos de contraseña incorrectos consecutivos, la cuenta para acceder a la instancia de DCS elegida se bloqueará durante 5 minutos. Las contraseñas no se pueden cambiar durante el período de bloqueo.

La contraseña debe cumplir los siguientes requisitos:

- No se puede dejar en blanco.
- No puede ser la misma que la contraseña anterior.
- Puede tener de 8 a 64 caracteres.
- Contiene al menos tres de los siguientes tipos de caracteres:
  - Letras en minúscula
  - Letras en mayúscula
  - Dígitos
  - caracteres especiales (`~!@#$^&*()-_=+\|{}:,<.>/?)

Paso 6 En el cuadro de diálogo **Change Password**, haga clic en **OK** para confirmar el cambio de contraseña.

----Fin

11.3 Reajuste de la contraseña de instancia

En la consola DCS, puede configurar una nueva contraseña si olvida la contraseña de la instancia.

NOTA

- Para una instancia de DCS Redis o Memcached, puede cambiarla de modo contraseña a modo sin contraseña o de modo sin contraseña a modo contraseña restableciendo su contraseña. Para más detalles, consulte **Cambio de la configuración de contraseña para instancias de DCS Memcached**.
- La instancia de DCS para la que desea restablecer la contraseña está en el estado **Running** .

Prerrequisitos:

Se ha creado una instancia de DCS.

Procedimiento

Paso 1 Inicie sesión en la **consola DCS**.

Paso 2 Haga clic en  en la esquina superior izquierda de la consola de gestión y seleccione la región donde se encuentra la instancia.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Paso 4 Elija **More** > **Reset Password** en la fila que contiene la instancia elegida.

Paso 5 En el cuadro de diálogo que se muestra, establezca **New Password** y **Confirm Password**.

NOTA

La contraseña debe cumplir los siguientes requisitos:

- No se puede dejar en blanco.
- Puede tener de 8 a 64 caracteres.
- Contiene al menos tres de los siguientes tipos de caracteres:
  - Letras en minúscula
  - Letras en mayúscula
  - Dígitos
  - caracteres especiales (`~!@#$^&*()-_=+\|{};,<.>/?)

Paso 6 Haz clic en **OK**.

NOTA

El sistema mostrará un mensaje de éxito solo después de que la contraseña se restablezca correctamente en todos los nodos. Si el restablecimiento falla, la instancia se reiniciará y se restaurará la contraseña de la instancia de caché.

----Fin

11.4 Cambio de la configuración de contraseña para instancias de DCS Redis

Caso

Se puede acceder a instancias de DCS Redis con o sin contraseñas. Después de crear una instancia, puede cambiar su configuración de contraseña en los siguientes escenarios:

- Para habilitar el acceso público para una instancia de DCS Redis 3.0, debe cambiar la instancia al modo protegido con contraseña antes de habilitar el acceso público.
- Para acceder a una instancia de DCS Redis en modo sin contraseña, puede habilitar el acceso sin contraseña para borrar la contraseña existente de la instancia.

NOTA

- Para cambiar la configuración de contraseña, la instancia de DCS Redis debe estar en el estado **Running**.
- El acceso sin contraseña puede comprometer la seguridad. Puede establecer una contraseña mediante la función de restablecimiento de contraseña.
- Por motivos de seguridad, el acceso sin contraseña debe estar deshabilitado cuando el acceso público está habilitado.

Procedimiento

Paso 1 Inicie sesión en la **consola DCS**.

Paso 2 Haga clic en en la esquina superior izquierda de la consola de gestión y seleccione la región donde se encuentra la instancia.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Paso 4 Para cambiar la configuración de contraseña de una instancia de DCS Redis, elija **Operation** > **More** > **Reset Password** en la fila que contiene la instancia elegida.

Paso 5 En el cuadro de diálogo **Reset Password**, realice cualquiera de las siguientes operaciones según sea necesario:

- Desde protegido con contraseña hasta sin contraseña:

  Cambie la opción de **Password-Free Access** y haga clic en **OK**.
- De sin contraseña a protegido con contraseña:

  Introduzca una contraseña, confírmela y haga clic en **OK**.

----Fin

11.5 Cambio de la configuración de contraseña para instancias de DCS Memcached

Caso

Se puede acceder a instancias de DCS Memcached con o sin contraseñas. Después de crear una instancia, puede cambiar su configuración de contraseña en los siguientes escenarios:

- Si desea acceder a una instancia de DCS Memcached protegida con contraseña sin el nombre de usuario y la contraseña, puede habilitar el acceso sin contraseña para borrar el nombre de usuario y la contraseña de la instancia.

  El protocolo de texto Memcached no admite la autenticación de nombre de usuario y contraseña. Para acceder a una instancia de Memcached de DCS mediante el protocolo de texto de Memcached, debe habilitar el acceso sin contraseña a la instancia.
- Si desea acceder a una instancia de DCS Memcached sin contraseña mediante un nombre de usuario y una contraseña, puede establecer una contraseña para la instancia mediante la función de restablecimiento de contraseña.

Procedimiento

Paso 1 Inicie sesión en la **consola DCS**.

Paso 2 Haga clic en  en la esquina superior izquierda de la consola de gestión y seleccione la región donde se encuentra la instancia.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Paso 4 Para habilitar el acceso sin contraseña a una instancia de DCS Memcached, elija **Operation** > **More** > **Reset Password** en la fila que contiene la instancia elegida.

Paso 5 En el cuadro de diálogo **Reset Password**, realice cualquiera de las siguientes operaciones según sea necesario:

- Desde protegido con contraseña hasta sin contraseña:

  Cambie la opción de **Password-Free Access** y haga clic en **OK**.
- De sin contraseña a protegido con contraseña:

  Introduzca una contraseña, confírmela y haga clic en **OK**.

----Fin

12 Cuotas

¿Qué es la cuota?

Una cuota es un límite en la cantidad o capacidad de un determinado tipo de recursos de servicio que puede usar, por ejemplo, el número máximo de instancias de DCS que puede crear y la cantidad máxima de memoria que puede usar.

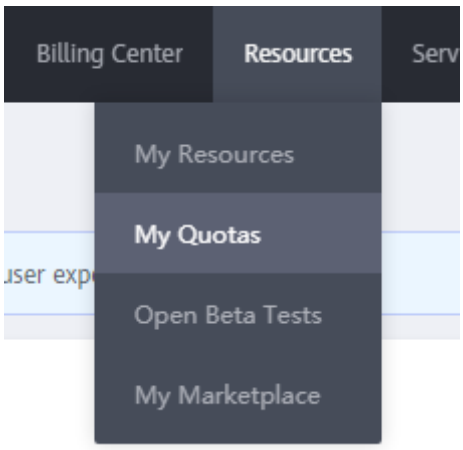
Si una cuota no puede satisfacer sus necesidades, solicite una cuota más alta.

¿Cómo puedo ver mi cuota?

1. Inicie sesión en la consola de gestión.
2. Haga clic en en la esquina superior izquierda de la consola de gestión y seleccione la región donde se encuentra la instancia.
3. En la esquina superior derecha de la página, seleccione **Resources** > **My Quotas**.

   Se muestra la página **Service Quota**.

   Figura 12-1 Mis cuotas

4. En la página **Service Quota**, vea las cuotas usadas y totales de recursos.

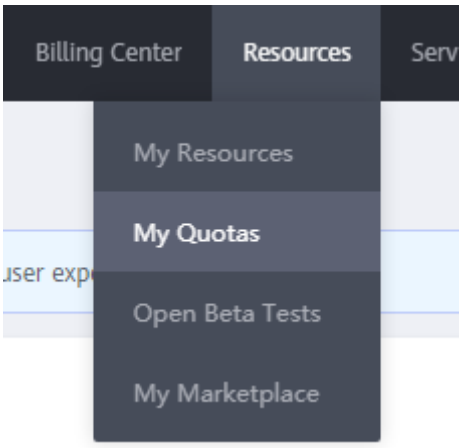
   Si una cuota no puede satisfacer sus necesidades, solicite una cuota más alta realizando las siguientes operaciones.

¿Cómo puedo aumentar mi cuota?

1. Inicie sesión en la consola de gestión.
2. En la esquina superior derecha de la página, seleccione **Resources** > **My Quotas**.

   Se muestra la página **Service Quota**.

   Figura 12-2 Mis cuotas

3. Haga clic en **Increase Quota**.
4. En la página **Create Service Ticket**, establezca los parámetros.

   En el área **Problem Description**, introduzca la cuota requerida y el motivo del ajuste de cuota.
5. Lea los acuerdos y confirme que está de acuerdo con ellos y, a continuación, haga clic en **Submit**.

13 Monitoreo

Cloud Eye es una plataforma de monitoreo segura y escalable. Supervisa las métricas de DCS y envía notificaciones si se activan alarmas o ocurren eventos.

13.1 Métricas de DCS

Introducción

En esta sección se describen las métricas de DCS notificadas a Cloud Eye, así como sus espacios de nombres y dimensiones. Puedes usar la consola de Cloud Eye o llamar a **las API** para consultar las métricas y alarmas de DCS.

Se supervisan los diferentes tipos de instancias en diferentes dimensiones.

Tabla 13-1 Control de dimensiones para diferentes tipos de instancia

Tipo de instancia	Monitoreo de instancias	Monitoreo del servidor Redis	Monitoreo de proxy
Nodo único	Se admite La supervisión de la dimensión de instancia se realiza en el servidor Redis.	N/A	N/A
Principal/En standby	Se admite Se monitorea el nodo principal.	Se admite Se supervisan los nodos principal y en standby.	N/A
Clúster Proxy	Se admite Los datos de monitoreo son los datos agregados del nodo principal.	Se admite Cada partición se monitorea.	Se admite Cada proxy se monitorea.

Tipo de instancia	Monitoreo de instancias	Monitoreo del servidor Redis	Monitoreo de proxy
Clúster Redis	Se admite Los datos de monitoreo son los datos agregados del nodo principal.	Se admite Cada partición se monitorea.	N/A

Namespace (espacio de nombres)

SYS.DCS

Métricas de instancias de DCS Redis 3.0

NOTA

- DCS for Redis 3.0 ya no se proporciona. Puede utilizar DCS for Redis 4.0 o 5.0 en su lugar.
- **Dimensiones** muestra las dimensiones de la métrica.

Tabla 13-2 Métricas de instancias de DCS Redis 3.0

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
cpu_usage	CPU Usage	Uso máximo de la CPU del objeto monitorizado entre múltiples valores de muestreo en un período de monitoreo Unidad: % Para una instancia de nodo único o principal/en standby, esta métrica indica el uso de CPU del nodo principal. Para una instancia de Clúster Proxy, esta métrica indica el valor promedio de todos los proxy.	0–100%	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
memory_usage	Memory Usage	Consumo de memoria del objeto monitoreado Unidad: % **AVISO** El uso de memoria no incluye el uso de memoria reservada.	0–100%	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
net_in_throughput	Network Input Throughput	Rendimiento de entrada por segundo en un puerto Unidad: byte/s	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
net_out_throughput	Network Output Throughput	Rendimiento de salida por segundo en un puerto Unidad: byte/s	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
connected_clients	Connected Clients	Número de clientes conectados (excluyendo los de nodos esclavos)	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
client_longest_out_list	Client Longest Output List	Lista de salida más larga entre las conexiones de cliente actuales	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
client_biggest_in_buf	Client Biggest Input Buf	Longitud máxima de los datos de entrada entre las conexiones actuales del cliente Unidad: byte	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
blocked_clients	Blocked Clients	Número de clientes suspendidos por operaciones en bloque como BLPOP, BRPOP y BRPOPLPUSH	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
used_memory	Used Memory	Número de bytes utilizados por el servidor Redis Unidad: byte	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
used_memory_rss	Used Memory RSS	Memoria de tamaño de conjunto residente (RSS) que ha utilizado el servidor Redis, que es la memoria que realmente reside en la memoria, incluida toda la memoria de pila y pila, pero no la memoria intercambiada Unidad: byte	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
used_memory_peak	Used Memory Peak	Memoria máxima consumida por Redis desde la última vez que se inició el servidor de Redis Unidad: byte	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
used_memory_lua	Used Memory Lua	Número de bytes utilizados por el motor Lua Unidad: byte	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
memory_frag_ratio	Memory Fragmentation Ratio	Fragmentación de memoria actual, que es la relación entre **used_memory_rss**/ **used_memory**.	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
total_connections_received	New Connections	Número de conexiones recibidas durante el período de supervisión	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
total_commands_processed	Commands Processed	Número de comandos procesados durante el período de supervisión	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
instantaneous_ops	Ops per Second	Cantidad de comandos procesados por segundo	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
total_net_input_bytes	Network Input Bytes	Número de bytes recibidos durante el período de supervisión Unidad: byte	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
total_net_output_bytes	Network Output Bytes	Número de bytes enviados durante el período de seguimiento Unidad: byte	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
instantaneous_input_kbps	Input Flow	Tráfico de entrada instantáneo Unidad: KB/s	≥ 0 KB/s	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
instantaneous_output_kbps	Output Flow	Tráfico de salida instantáneo Unidad: KB/s	≥ 0 KB/s	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
rejected_connections	Rejected Connections	Número de conexiones que han excedido los clientes máximos y han sido rechazadas durante el período de supervisión	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
expired_keys	Expired Keys	Número de claves que han caducado y se han eliminado durante el período de supervisión	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
evicted_keys	Evicted Keys	Número de claves que han sido desalojadas y eliminadas durante el período de supervisión	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
keyspace_hits	Keyspace Hits	Número de búsquedas exitosas de claves en el diccionario principal durante el período de supervisión	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
keyspace_misses	Keyspace Misses	Número de búsquedas fallidas de claves en el diccionario principal durante el período de supervisión	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
pubsub_channels	PubSub Channels	Número de canales Pub/Sub	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
pubsub_patterns	PubSub Patterns	Número de patrones Pub/Sub	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
keyspace_hits_perc	Hit Rate	Relación entre el número de aciertos de caché de Redis y el número de búsquedas. Cálculo: keyspace_hits/(keyspace_hits + keyspace_misses) Unidad: %	0–100%	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
command_max_delay	Maximum Command Latency	Latencia máxima de los comandos Unidad: ms	≥ 0 ms	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
auth_errors	Authentication Failures	Cantidad de autenticaciones con error	≥ 0	Instancia DCS Redis de nodo unico o principal/en standby	1 minuto
is_slow_log_exist	Slow Query Logs	Existencia de registros de consultas lentos en la instancia	• 1: sí • 0: no	Instancia DCS Redis de nodo unico o principal/en standby	1 minuto
keys	Keys	Número de claves en Redis	≥ 0	Instancia DCS Redis de nodo unico o principal/en standby	1 minuto

Métricas de instancias de DCS Redis 4.0/5.0

NOTA

Dimensiones muestra las dimensiones de la métrica.

Tabla 13-3 Métricas de instancias de DCS Redis 4.0/5.0

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
cpu_usage	CPU Usage	Uso máximo de la CPU del objeto monitorizado entre múltiples valores de muestreo en un período de monitoreo Unidad: %	0–100%	Instancia DCS Redis de nodo unico o principal/en standby	1 minuto
cpu_avg_usage	Average CPU Usage	Uso promedio de la CPU del objeto monitorizado de múltiples valores de muestreo en un período de monitoreo Unidad: %	0–100%	Instancia DCS Redis de nodo unico o principal/en standby	1 minuto
command_max_delay	Maximum Command Latency	Latencia máxima de los comandos Unidad: ms	≥ 0 ms	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
total_connections_received	New Connections	Número de conexiones recibidas durante el período de supervisión	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
is_slow_log_exist	Slow Query Logs	Existencia de registros de consultas lentos en la instancia	● 1: sí ● 0: no	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
memory_usage	Memory Usage	Consumo de memoria del objeto monitoreado Unidad: %	0–100%	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
expires	Keys With an Expiration	Número de claves con una caducidad en Redis	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
keyspace_hits_perc	Hit Rate	Relación entre el número de aciertos de caché de Redis y el número de búsquedas. Cálculo: keyspace_hits/(keyspace_hits + keyspace_misses) Unidad: %	0–100%	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
used_memory	Used Memory	Número total de bytes utilizados por el servidor Redis Unidad: byte	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
used_memory_dataset	Used Memory Dataset	Memoria de conjunto de datos que ha utilizado el servidor Redis Unidad: byte	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
used_memory_dataset_perc	Used Memory Dataset Ratio	Porcentaje de memoria de datos que Redis ha utilizado con respecto al total de memoria utilizada Unidad: %	0–100%	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
used_memory_rss	Used Memory RSS	Memoria de tamaño de conjunto residente (RSS) que ha utilizado el servidor Redis, que es la memoria que realmente reside en la memoria, incluida toda la memoria de pila y pila, pero no la memoria intercambiada Unidad: byte	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
instantaneous_ops	Ops per Second	Cantidad de comandos procesados por segundo	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
keyspace_misses	Keyspace Misses	Número de búsquedas fallidas de claves en el diccionario principal durante el período de supervisión	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
keys	Keys	Número de claves en Redis	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
blocked_clients	Blocked Clients	Número de clientes suspendidos por operaciones en bloque	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
connected_clients	Connected Clients	Número de clientes conectados (excluyendo los de nodos esclavos)	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
del	DEL	Número de comandos **DEL** procesados por segundo	0–500,000	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
evicted_keys	Evicted Keys	Número de claves que han sido desalojadas y eliminadas durante el período de supervisión	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
expire	EXPIRE	Número de comandos **EXPIRE** procesados por segundo	0–500,000	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
expired_keys	Expired Keys	Número de claves que han caducado y se han eliminado durante el período de supervisión	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
get	GET	Número de comandos **GET** procesados por segundo	0–500,000	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
hdel	HDEL	Número de comandos **HDEL** procesados por segundo	0–500,000	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
hget	HGET	Número de comandos **HGET** procesados por segundo	0–500,000	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
hmget	HMGET	Número de comandos **HMGET** procesados por segundo	0–500,000	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
hmset	HMSET	Número de comandos **HMSET** procesados por segundo	0–500,000	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
hset	HSET	Número de comandos **HSET** procesados por segundo	0–500,000	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
instantaneous_input_kbps	Input Flow	Tráfico de entrada instantáneo Unidad: KB/s	≥ 0 KB/s	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
instantaneous_output_kbps	Output Flow	Tráfico de salida instantáneo Unidad: KB/s	≥ 0 KB/s	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
memory_frag_ratio	Memory Fragmentation Ratio	Relación entre la memoria usada RSS y la memoria usada	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minute
mget	MGET	Número de comandos **MGET** procesados por segundo	0–500,000	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
mset	MSET	Número de comandos **MSET** procesados por segundo	0–500,000	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
pubsub_channels	PubSub Channels	Número de canales Pub/Sub	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
pubsub_patterns	PubSub Patterns	Número de patrones Pub/Sub	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
set	SET	Número de comandos **SET** procesados por segundo	0–500,000	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
used_memory_lua	Used Memory Lua	Número de bytes utilizados por el motor Lua Unidad: byte	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
used_memory_peak	Used Memory Peak	Memoria máxima consumida por Redis desde la última vez que se inició el servidor de Redis Unidad: byte	≥ 0	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
sadd	Sadd	Cantidad de comandos **SADD** procesados por segundo Unidad: vez/s	0–500,000	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
smembers	Smembers	Cantidad de comandos **SMEMBERS** procesados por segundo Unidad: vez/s	0–500,000	Instancia DCS Redis de nodo unico, principal/en standby o de clúster	1 minuto
rx_controlled	Flow Control Times	Número de tiempos de control de flujo durante el período de supervisión Si el valor es mayor que 0, el ancho de banda usado excede el límite superior y se activa el control de flujo. Unidad: Vez	≥ 0	Instancia de Clúster Redis	1 minuto
bandwidth_usage	Bandwidth Usage	Porcentaje del ancho de banda utilizado al límite máximo de ancho de banda	0–200%	Instancia de Clúster Redis	1 minuto
command_max_rt	Maximum Latency	Retardo máximo desde cuando el nodo recibe comandos hasta cuando responde Unidad: us	≥ 0	Instancia de nodo único de DCS Redis 4.0/5.0/6.0	1 minuto
command_avg_rt	Average Latency	Retraso promedio desde cuando el nodo recibe comandos hasta cuando responde Unidad: us	≥ 0	Instancia de nodo único de DCS Redis 4.0/5.0/6.0	1 minuto

Métricas de servidor Redis de instancias de DCS para Redis

NOTA

- Para las instancias de Clúster Proxy, el monitoreo cubre los servidores y proxy de Redis. Para las instancias de Clúster Redis de DCS Redis 4.0 y 5.0 y las instancias principal/en standby, el monitoreo solo cubre los servidores de Redis.
- **Dimensiones** muestra las dimensiones de la métrica.

Tabla 13-4 Métricas de Redis Server

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
cpu_usage	CPU Usage	Uso máximo de la CPU del objeto monitorizado entre múltiples valores de muestreo en un período de monitoreo Unidad: %	0–100%	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis	1 minuto
cpu_avg_usage	Average CPU Usage	Uso promedio de la CPU del objeto monitorizado de múltiples valores de muestreo en un período de monitoreo Unidad: %	0–100%	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis 4.0 o 5.0 o una instancia principal/en standby de DCS Redis 6.0	1 minuto
memory_usage	Memory Usage	Consumo de memoria del objeto monitoreado Unidad: %	0–100%	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis	1 minuto

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
connected_clients	Connected Clients	Número de clientes conectados (excluyendo los de nodos esclavos)	≥ 0	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis	1 minuto
client_longest_out_list	Client Longest Output List	Lista de salida más larga entre las conexiones de cliente actuales	≥ 0	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis 3.0 o 4.0	1 minuto
client_biggest_in_buf	Client Biggest Input Buf	Longitud máxima de los datos de entrada entre las conexiones actuales del cliente Unidad: byte	≥ 0	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis 3.0 o 4.0	1 minuto
blocked_clients	Blocked Clients	Número de clientes suspendidos por operaciones en bloque como BLPOP, BRPOP y BRPOPLPUSH	≥ 0	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis	1 minuto
used_memory	Used Memory	Número total de bytes utilizados por el servidor Redis Unidad: byte	≥ 0	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis	1 minuto

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
used_memory_rss	Used Memory RSS	Memoria RSS que el servidor Redis ha utilizado, que incluye toda la memoria de pila y pila, pero no la memoria intercambiada Unidad: byte	≥ 0	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis	1 minuto
used_memory_peak	Used Memory Peak	Memoria máxima consumida por Redis desde la última vez que se inició el servidor de Redis Unidad: byte	≥ 0	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis	1 minuto
used_memory_lua	Used Memory Lua	Número de bytes utilizados por el motor Lua Unidad: byte	≥ 0	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis	1 minuto
memory_frag_ratio	Memory Fragmentation Ratio	Fragmentación de memoria actual, que es la relación entre **used_memory_rss**/ **used_memory**.	≥ 0	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis	1 minuto
total_connections_received	New Connections	Número de conexiones recibidas durante el período de supervisión	≥ 0	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis	1 minuto

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
total_commands_processed	Commands Processed	Número de comandos procesados durante el período de supervisión	≥ 0	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis	1 minuto
instantaneous_ops	Ops per Second	Cantidad de comandos procesados por segundo	≥ 0	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis	1 minuto
total_net_input_bytes	Network Input Bytes	Número de bytes recibidos durante el período de supervisión Unidad: byte	≥ 0	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis	1 minuto
total_net_output_bytes	Network Output Bytes	Número de bytes enviados durante el período de seguimiento Unidad: byte	≥ 0	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis	1 minuto
instantaneous_input_kbps	Input Flow	Tráfico de entrada instantáneo Unidad: KB/s	≥ 0 KB/s	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis	1 minuto
instantaneous_output_kbps	Output Flow	Tráfico de salida instantáneo Unidad: KB/s	≥ 0 KB/s	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis	1 minuto

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
rejected_connections	Rejected Connections	Número de conexiones que han excedido los clientes máximos y han sido rechazadas durante el período de supervisión	≥ 0	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis	1 minuto
expired_keys	Expired Keys	Número de claves que han caducado y se han eliminado durante el período de supervisión	≥ 0	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis	1 minuto
evicted_keys	Evicted Keys	Número de claves que han sido desalojadas y eliminadas durante el período de supervisión	≥ 0	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis	1 minuto
pubsub_channels	PubSub Channels	Número de canales Pub/Sub	≥ 0	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis	1 minuto
pubsub_patterns	PubSub Patterns	Número de patrones Pub/Sub	≥ 0	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis 3.0, 4.0, or 5.0	1 minuto

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
keyspace_hits_perc	Hit Rate	Relación entre el número de aciertos de caché de Redis y el número de búsquedas. Cálculo: keyspace_hits/(keyspace_hits + keyspace_misses) Unidad: %	0–100%	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis	1 minuto
command_max_delay	Maximum Command Latency	Latencia máxima de los comandos Unidad: ms	≥ 0 ms	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis	1 minuto
is_slow_log_exist	Slow Query Logs	Existencia de registros de consultas lentos en el nodo	• 1: sí • 0: no	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis	1 minuto
keys	Keys	Número de claves en Redis	≥ 0	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis	1 minuto

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
sadd	Sadd	Cantidad de comandos **SADD** procesados por segundo Unidad: vez/s	0–500,000	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis 4.0 o 5.0 o una instancia principal/en standby de DCS Redis 6.0	1 minuto
smembers	Smembers	Cantidad de comandos **SMEMBERS** procesados por segundo Unidad: vez/s	0–500,000	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis 4.0 o 5.0 o una instancia principal/en standby de DCS Redis 6.0	1 minuto
ms_repl_offset	Replication Gap	Intervalo de sincronización de datos entre la instancia principal y la réplica	-	Réplica Redis Server de una instancia principal/en standby o de clúster de DCS Redis 4.0 o 5.0 o una instancia principal/en standby de DCS Redis 6.0	1 minuto

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
del	DEL	Número de comandos **DEL** procesados por segundo Unidad: vez/s	0–500,000	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis 4.0 o 5.0 o una instancia principal/en standby de DCS Redis 6.0	1 minuto
expire	EXPIRE	Número de comandos **EXPIRE** procesados por segundo Unidad: vez/s	0–500,000	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis 4.0 o 5.0 o una instancia principal/en standby de DCS Redis 6.0	1 minuto
get	GET	Número de comandos **GET** procesados por segundo Unidad: vez/s	0–500,000	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis 4.0 o 5.0 o una instancia principal/en standby de DCS Redis 6.0	1 minuto

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
hdel	HDEL	Número de comandos **HDEL** procesados por segundo Unidad: vez/s	0–500,000	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis 4.0 o 5.0 o una instancia principal/en standby de DCS Redis 6.0	1 minuto
hget	HGET	Número de comandos **HGET** procesados por segundo Unidad: vez/s	0–500,000	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis 4.0 o 5.0 o una instancia principal/en standby de DCS Redis 6.0	1 minuto
hmget	HMGET	Número de comandos **HMGET** procesados por segundo Unidad: vez/s	0–500,000	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis 4.0 o 5.0 o una instancia principal/en standby de DCS Redis 6.0	1 minuto

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
hmset	HMSET	Número de comandos **HMSET** procesados por segundo Unidad: vez/s	0–500,000	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis 4.0 o 5.0 o una instancia principal/en standby de DCS Redis 6.0	1 minuto
hset	HSET	Número de comandos **HSET** procesados por segundo Unidad: vez/s	0–500,000	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis 4.0 o 5.0 o una instancia principal/en standby de DCS Redis 6.0	1 minuto
mget	MGET	Número de comandos **MGET** procesados por segundo Unidad: vez/s	0–500,000	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis 4.0 o 5.0 o una instancia principal/en standby de DCS Redis 6.0	1 minuto

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
mset	MSET	Número de comandos **MSET** procesados por segundo Unidad: vez/s	0–500,000	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis 4.0 o 5.0 o una instancia principal/en standby de DCS Redis 6.0	1 minuto
set	SET	Número de comandos **SET** procesados por segundo Unidad: vez/s	0–500,000	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis 4.0 o 5.0 o una instancia principal/en standby de DCS Redis 6.0	1 minuto
rx_controlled	Flow Control Times	Número de tiempos de control de flujo durante el período de supervisión Si el valor es mayor que 0, el ancho de banda usado excede el límite superior y se activa el control de flujo. Unidad: Vez	≥ 0	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis 4.0 o 5.0 o una instancia principal/en standby de DCS Redis 6.0	1 minuto

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
bandwidth_usage	Bandwidth Usage	Porcentaje del ancho de banda utilizado al límite máximo de ancho de banda	0–200%	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis 4.0 o 5.0 o una instancia principal/en standby de DCS Redis 6.0	1 minuto
connections_usage	Connection Usage	Porcentaje del número actual de conexiones al número máximo permitido de conexiones Unidad: %	0–100%	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis 4.0 o 5.0 o una instancia principal/en standby de DCS Redis 6.0	1 minuto
command_max_rt	Maximum Latency	Retardo máximo desde cuando el nodo recibe comandos hasta cuando responde Unidad: us	≥ 0	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis 4.0 o 5.0 o una instancia principal/en standby de DCS Redis 6.0	1 minuto

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
command_avg_rt	Average Latency	Retraso promedio desde cuando el nodo recibe comandos hasta cuando responde Unidad: us	≥ 0	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis 4.0 o 5.0 o una instancia principal/en standby de DCS Redis 6.0	1 minuto
sync_full	Full Sync Times	Número total de sincronizaciones completas desde que se inició por última vez el servidor Redis	≥ 0	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis 4.0 o 5.0 o una instancia principal/en standby de DCS Redis 6.0	1 minuto
slow_log_counts	Slow Queries	Número de veces que se producen consultas lentas dentro de un período de supervisión	≥ 0	Servidor Redis de una instancia principal/en standby o de clúster de DCS Redis 4.0 o 5.0 o una instancia principal/en standby de DCS Redis 6.0	1 minuto

Métricas de proxy

NOTA

Dimensiones muestra las dimensiones de la métrica.

Tabla 13-5 Métricas de proxy de instancias de Clúster Proxy de DCS Redis 3.0

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
cpu_usage	CPU Usage	Uso máximo de la CPU del objeto monitorizado entre múltiples valores de muestreo en un período de monitoreo Unidad: %	0–100%	Proxy en una instancia de Clúster Proxy de DCS Redis 3.0	1 minuto
memory_usage	Memory Usage	Consumo de memoria del objeto monitoreado Unidad: %	0–100%	Proxy en una instancia de Clúster Proxy de DCS Redis 3.0	1 minuto
p_connected_clients	Connected Clients	Cantidad de clientes conectados	≥ 0	Proxy en una instancia de Clúster Proxy de DCS Redis 3.0	1 minuto
max_rxpck_per_sec	Max. NIC Data Packet Receive Rate	Número máximo de paquetes de datos recibidos por la NIC proxy por segundo durante el período de supervisión Unidad: paquetes/segundo	0–10,000,000	Proxy en una instancia de Clúster Proxy de DCS Redis 3.0	1 minuto
max_txpck_per_sec	Max. NIC Data Packet Transmit Rate	Número máximo de paquetes de datos transmitidos por la NIC proxy por segundo durante el período de supervisión Unidad: paquetes/segundo	0–10,000,000	Proxy en una instancia de Clúster Proxy de DCS Redis 3.0	1 minuto

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
max_rxkB_per_sec	Maximum Inbound Bandwidth	El mayor volumen de datos recibidos por la NIC proxy por segundo Unidad: KB/s	≥ 0 KB/s	Proxy en una instancia de Clúster Proxy de DCS Redis 3.0	1 minuto
max_txkB_per_sec	Maximum Outbound Bandwidth	El mayor volumen de datos transmitidos por la NIC proxy por segundo Unidad: KB/s	≥ 0 KB/s	Proxy en una instancia de Clúster Proxy de DCS Redis 3.0	1 minuto
avg_rxpck_per_sec	Average NIC Data Packet Receive Rate	Número promedio de paquetes de datos recibidos por la NIC proxy por segundo durante el período de supervisión Unidad: paquetes/segundo	0–10,000,000	Proxy en una instancia de Clúster Proxy de DCS Redis 3.0	1 minuto
avg_txpck_per_sec	Average NIC Data Packet Transmit Rate	Número promedio de paquetes de datos transmitidos por la NIC proxy por segundo durante el período de supervisión Unidad: paquetes/segundo	0–10,000,000	Proxy en una instancia de Clúster Proxy de DCS Redis 3.0	1 minuto
avg_rxkB_per_sec	Average Inbound Bandwidth	Volumen promedio de datos recibidos por la NIC proxy por segundo Unidad: KB/s	≥ 0 KB/s	Proxy en una instancia de Clúster Proxy de DCS Redis 3.0	1 minuto

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
avg_txkB_per_sec	Average Outbound Bandwidth	Volumen promedio de datos transmitidos por la NIC proxy por segundo Unidad: KB/s	≥ 0 KB/s	Proxy en una instancia de Clúster Proxy de DCS Redis 3.0	1 minuto

Tabla 13-6 Métricas de proxy de instancias de Clúster Proxy de DCS Redis 4.0 o 5.0

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
node_status	Proxy Status	Indicación de que si el proxy es normal.	● 0: Normal ● 1: Anormal	Proxy en una instancia de Clúster Proxy de DCS Redis 4.0 o 5.0	1 minuto
cpu_usage	CPU Usage	Uso máximo de la CPU del objeto monitorizado entre múltiples valores de muestreo en un período de monitoreo Unidad: %	0–100%	Proxy en una instancia de Clúster Proxy de DCS Redis 4.0 o 5.0	1 minuto
memory_usage	Memory Usage	Consumo de memoria del objeto monitoreado Unidad: %	0–100%	Proxy en una instancia de Clúster Proxy de DCS Redis 4.0 o 5.0	1 minuto

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
p_connected_clients	Connected Clients	Cantidad de clientes conectados	≥ 0	Proxy en una instancia de Clúster Proxy de DCS Redis 4.0 o 5.0	1 minuto
instantaneous_ops	Ops per Second	Cantidad de comandos procesados por segundo	≥ 0	Proxy en una instancia de Clúster Proxy de DCS Redis 4.0 o 5.0	1 minuto
instantaneous_input_kbps	Input Flow	Tráfico de entrada instantáneo Unidad: KB/s	≥ 0 KB/s	Proxy en una instancia de Clúster Proxy de DCS Redis 4.0 o 5.0	1 minuto
instantaneous_output_kbps	Output Flow	Tráfico de salida instantáneo Unidad: KB/s	≥ 0 KB/s	Proxy en una instancia de Clúster Proxy de DCS Redis 4.0 o 5.0	1 minuto
total_net_input_bytes	Network Input Bytes	Número de bytes recibidos durante el período de supervisión Unidad: byte	≥ 0	Proxy en una instancia de Clúster Proxy de DCS Redis 4.0 o 5.0	1 minuto
total_net_output_bytes	Network Output Bytes	Número de bytes enviados durante el período de seguimiento Unidad: byte	≥ 0	Proxy en una instancia de Clúster Proxy de DCS Redis 4.0 o 5.0	1 minuto
connections_usage	Connection Usage	Porcentaje del número actual de conexiones al número máximo permitido de conexiones. Unidad: %	0–100%	Proxy en una instancia de Clúster Proxy de DCS Redis 4.0 o 5.0	1 minuto

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
command_max_rt	Maximum Latency	Retardo máximo desde cuando el nodo recibe comandos hasta cuando responde Unidad: us	≥ 0	Proxy en una instancia de Clúster Proxy de DCS Redis 4.0 o 5.0	1 minuto
command_avg_rt	Average Latency	Retraso promedio desde cuando el nodo recibe comandos hasta cuando responde Unidad: us	≥ 0	Proxy en una instancia de Clúster Proxy de DCS Redis 4.0 o 5.0	1 minuto

Métricas de instancias de DCS Memcached

NOTA

Dimensiones muestra las dimensiones de la métrica.

Tabla 13-7 Métricas de instancias de DCS Memcached

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
cpu_usage	CPU Usage	Uso máximo de la CPU del objeto monitorizado entre múltiples valores de muestreo en un período de monitoreo Unidad: %	0–100%	Instancia de DCS Memcached	1 minuto

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
memory_usage	Memory Usage	Consumo de memoria del objeto monitoreado Unidad: %	0–100%	Instancia de DCS Memcached	1 minuto
net_in_throughput	Network Input Throughput	Rendimiento de entrada por segundo en un puerto Unidad: byte/s	≥ 0	Instancia de DCS Memcached	1 minuto
net_out_throughput	Network Output Throughput	Rendimiento de salida por segundo en un puerto Unidad: byte/s	≥ 0	Instancia de DCS Memcached	1 minuto
mc_connected_clients	Connected Clients	Número de clientes conectados (excluyendo los de nodos esclavos)	≥ 0	Instancia de DCS Memcached	1 minuto
mc_used_memory	Used Memory	Número de bytes utilizados por Memcached Unidad: byte	≥ 0	Instancia de DCS Memcached	1 minuto
mc_used_memory_rss	Used Memory RSS	Memoria RSS utilizada que en realidad reside en la memoria, incluida toda la memoria de pila y pila, pero no la memoria intercambiada Unidad: byte	≥ 0	Instancia de DCS Memcached	1 minuto

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
mc_used_memory_peak	Used Memory Peak	Memoria máxima consumida desde la última vez que se inició el servidor Unidad: byte	≥ 0	Instancia de DCS Memcached	1 minuto
mc_memory_frag_ratio	Memory Fragmentation Ratio	Relación entre la memoria usada RSS y la memoria usada	≥ 0	Instancia de DCS Memcached	1 minuto
mc_connections_received	New Connections	Número de conexiones recibidas durante el período de supervisión	≥ 0	Instancia de DCS Memcached	1 minuto
mc_commands_processed	Commands Processed	Número de comandos procesados durante el período de supervisión	≥ 0	Instancia de DCS Memcached	1 minuto
mc_instantaneous_ops	Ops per Second	Cantidad de comandos procesados por segundo	≥ 0	Instancia de DCS Memcached	1 minuto
mc_net_input_bytes	Network Input Bytes	Número de bytes recibidos durante el período de supervisión Unidad: byte	≥ 0	Instancia de DCS Memcached	1 minuto
mc_net_output_bytes	Network Output Bytes	Número de bytes enviados durante el período de seguimiento Unidad: byte	≥ 0	Instancia de DCS Memcached	1 minuto

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
mc_instantaneous_input_kbps	Input Flow	Tráfico de entrada instantáneo Unidad: KB/s	≥ 0 KB/s	Instancia de DCS Memcached	1 minuto
mc_instantaneous_output_kbps	Output Flow	Tráfico de salida instantáneo Unidad: KB/s	≥ 0 KB/s	Instancia de DCS Memcached	1 minuto
mc_rejected_connections	Rejected Connections	Número de conexiones que han excedido los clientes máximos y han sido rechazadas durante el período de supervisión	≥ 0	Instancia de DCS Memcached	1 minuto
mc_expired_keys	Expired Keys	Número de claves que han caducado y se han eliminado durante el período de supervisión	≥ 0	Instancia de DCS Memcached	1 minuto
mc_evicted_keys	Evicted Keys	Número de claves que han sido desalojadas y eliminadas durante el período de supervisión	≥ 0	Instancia de DCS Memcached	1 minuto
mc_cmd_get	Number of Retrieval Requests	Número de solicitudes de recuperación de datos recibidas	≥ 0	Instancia de DCS Memcached	1 minuto
mc_cmd_set	Number of Storage Requests	Número de solicitudes de almacenamiento de datos recibidas	≥ 0	Instancia de DCS Memcached	1 minuto

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
mc_cmd_flush	Number of Flush Requests	Número de solicitudes de autorización de datos recibidas	≥ 0	Instancia de DCS Memcached	1 minuto
mc_cmd_touch	Number of Touch Requests	Número de solicitudes recibidas para modificar el período de validez de los datos	≥ 0	Instancia de DCS Memcached	1 minuto
mc_get_hits	Number of Retrieval Hits	Número de operaciones de recuperación de datos exitosas	≥ 0	Instancia de DCS Memcached	1 minuto
mc_get_misses	Number of Retrieval Misses	Número de operaciones de recuperación de datos fallidas debido a la inexistencia de claves	≥ 0	Instancia de DCS Memcached	1 minuto
mc_delete_hits	Number of Delete Hits	Número de operaciones de eliminación de datos exitosas	≥ 0	Instancia de DCS Memcached	1 minuto
mc_delete_misses	Number of Delete Misses	Número de operaciones de eliminación de datos fallidas debido a la inexistencia de claves	≥ 0	Instancia de DCS Memcached	1 minuto
mc_incr_hits	Number of Increment Hits	Número de operaciones de incremento exitosas	≥ 0	Instancia de DCS Memcached	1 minuto

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
mc_incr_misses	Number of Increment Misses	Número de operaciones de incremento fallidas debido a la inexistencia de clave	≥ 0	Instancia de DCS Memcached	1 minuto
mc_decr_hits	Number of Decrement Hits	Número de operaciones de reducción exitosas	≥ 0	Instancia de DCS Memcached	1 minuto
mc_decr_misses	Number of Decrement Misses	Número de operaciones de disminución fallidas debido a la inexistencia de claves	≥ 0	Instancia de DCS Memcached	1 minuto
mc_cas_hits	Number of CAS Hits	Número de operaciones CAS exitosas	≥ 0	Instancia de DCS Memcached	1 minuto
mc_cas_misses	Number of CAS Misses	Número de operaciones CAS fallidas debido a la inexistencia de claves	≥ 0	Instancia de DCS Memcached	1 minuto
mc_cas_badval	Number of CAS Values Not Matched	Número de operaciones CAS fallidas debido a la falta de coincidencia de valores de CAS	≥ 0	Instancia de DCS Memcached	1 minuto
mc_touch_hits	Number of Touch Hits	Número de solicitudes de modificación del período de validez de los datos con éxito	≥ 0	Instancia de DCS Memcached	1 minuto

ID de métrica	Nombre de la métrica	Descripción de Métrica	Rango de valores	Objeto monitoreado	Período de monitoreo (datos brutos)
mc_touch_misses	Number of Touch Misses	Número de solicitudes fallidas para modificar el período de validez de los datos debido a la inexistencia clave	≥ 0	Instancia de DCS Memcached	1 minuto
mc_auth_cmds	Authentication Requests	Número de solicitudes de autenticación	≥ 0	Instancia de DCS Memcached	1 minuto
mc_auth_errors	Authentication Failures	Número de solicitudes de autenticación fallidas	≥ 0	Instancia de DCS Memcached	1 minuto
mc_curr_items	Number of Items Stored	Número de elementos de datos almacenados	≥ 0	Instancia de DCS Memcached	1 minuto
mc_command_max_delay	Maximum Command Latency	Latencia máxima de los comandos Unidad: ms	≥ 0 ms	Instancia de DCS Memcached	1 minuto
mc_is_slow_log_exist	Slow Query Logs	Existencia de registros de consultas lentos en la instancia	● 1: sí ● 0: no	Instancia de DCS Memcached	1 minuto
mc_keyspace_hits_perc	Hit Rate	Relación entre el número de aciertos en caché de Memcached y el número de búsquedas Unidad: %	0–100%	Instancia de DCS Memcached	1 minuto

Dimensiones

Clave	Valor
dcs_instance_id	Instancia de DCS Redis
dcs_cluster_redis_node	Servidor de Redis
dcs_cluster_proxy_node	Proxy en una instancia de Clúster Proxy de DCS Redis 3.0
dcs_cluster_proxy2_node	Proxy en una instancia de Clúster Proxy de DCS Redis 4.0 o 5.0
dcs_memcached_instance_id	Instancia de DCS Memcached

13.2 Métricas comunes

En esta sección se describen las métricas de Redis comunes.

Tabla 13-8 Métricas comunes

Métrica	Descripción
CPU Usage	Esta métrica indica el valor máximo en cada período de medición (nivel minuto: cada minuto; segundo nivel: cada 5 segundos). ● Para una instancia de nodo único o principal/en standby, puede ver el uso de la CPU de la instancia. ● Para una instancia de Clúster Proxy, puede ver el uso de la CPU de los servidores Redis y los proxy. ● Para una instancia de Clúster Redis, solo puede ver el uso de CPU de los servidores Redis.
Memory Usage	Esta métrica mide el uso de memoria en cada período de medición (nivel de minutos: cada minuto; segundo nivel: cada 5 segundos). ● Para una instancia de nodo único o principal/en standby, puede ver el uso de memoria de la instancia. ● Para una instancia de Clúster Proxy, puede ver el uso de memoria de la instancia y los proxy. ● Para una instancia de Clúster Redis, solo puede ver el uso de memoria de los servidores Redis. **AVISO** El uso de memoria no incluye el uso de memoria reservada.

Métrica	Descripción
Connected Clients	Esta métrica indica el número de clientes conectados instantáneos, es decir, el número de conexiones simultáneas. Esta métrica no incluye el número de conexiones a los nodos en standby de instancias principal/en standby o de clúster. For details about the maximum allowed number of connections, see the "Max.
Ops per Second	Esta métrica indica el número de operaciones procesadas por segundo. Para obtener detalles sobre el número máximo permitido de operaciones por segundo, consulte la columna "Rendimiento de referencia (QPS) " de diferentes tipos de instancia que se enumeran en **Especificaciones de instancias de DCS**.
Input Flow	Esta métrica indica el tráfico de entrada instantáneo. ● Los datos de supervisión en el nivel de instancia muestran el tráfico de entrada agregado de todos los nodos. ● Los datos de supervisión en el nivel de nodo muestran el tráfico de entrada del nodo actual.
Output Flow	Esta métrica indica el tráfico de salida instantáneo. ● Los datos de supervisión en el nivel de instancia muestran el tráfico de salida agregado de todos los nodos. ● Los datos de supervisión en el nivel de nodo muestran el tráfico de salida del nodo actual.
Bandwidth Usage	Esta métrica indica el porcentaje del ancho de banda utilizado al límite máximo de ancho de banda.
Commands Processed	Esta métrica indica el número de comandos procesados durante el período de supervisión. El período de supervisión predeterminado es de 1 minuto. El período de monitoreo de esta métrica es diferente del de la métrica **Ops per Second**. La métrica **Ops per Second** mide el número instantáneo de comandos procesados. La métrica **Commands Processed** mide el número total de comandos procesados durante el período de supervisión.
Flow Control Times	Esta métrica indica el número de veces que se excede el ancho de banda máximo permitido durante el período de supervisión. Para obtener más información sobre el ancho de banda máximo permitido, consulte la columna "Ancho de banda máximo/ asegurado" de diferentes tipos de instancia enumerados en **Especificaciones de instancias de DCS**.
Slow Queries	Esta métrica indica si existen consultas lentas en la instancia. Para obtener más información sobre la causa de una consulta lenta, consulte **Observación de consultas lentas de Redis**.

13.3 Consulta de Métricas

El servicio Cloud Eye supervisa el rendimiento en ejecución de las instancias de DCS.

Procedimiento

Paso 1 Inicie sesión en la **consola DCS**.

Paso 2 Haga clic en  en la esquina superior izquierda de la consola de gestión y seleccione la región donde se encuentra la instancia.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Paso 4 Haga clic en la instancia deseada.

Paso 5 Elija **Performance Monitoring**. Se muestran todas las métricas de supervisión de la instancia.

NOTA

También puede hacer clic en **View Metric** en la columna **Operation** de la página **Cache Manager**. Serás redirigido a la consola Cloud Eye. Las métricas que se muestran en la consola de Cloud Eye son las mismas que las que se muestran en la página **Performance Monitoring** de la consola de DCS.

----Fin

13.4 Configuración de reglas de alarma para métricas críticas

Esta sección describe las reglas de alarma de algunas métricas y cómo configurarlas. En escenarios reales, configure las reglas de alarma para métricas haciendo referencia a las siguientes políticas de alarma.

Políticas de alarma para instancias de DCS Redis

Tabla 13-9 Métricas de instancia de DCS Redis para configurar reglas de alarma para

Métrica	Rango de valores	Política de alarmas	Límite superior de aproximación	Sugerencia sobre el manejo
CPU Usage	0–100%	Umbral de alarma: > 70% Número de períodos consecutivos: 2 Severidad de la alarma: mayor	No	Considere la ampliación de la capacidad basada en el análisis del servicio. La capacidad de la CPU de una instancia de nodo único o principal/en standby no se puede ampliar. Si necesita una mayor capacidad, use una instancia de clúster en su lugar. Esta métrica solo está disponible para instancias de nodo único, principal/en standby y de Clúster Proxy. Para las instancias de Clúster Redis, esta métrica solo está disponible en el nivel de servidor Redis. Puede ver la métrica en la página de ficha **Redis Server** en la página **Performance Monitoring** de la instancia.
Average CPU Usage	0–100%	Umbral de alarma: > 70% Número de períodos consecutivos: 2 Severidad de la alarma: mayor	No	Considere la ampliación de la capacidad basada en el análisis del servicio. La capacidad de la CPU de una instancia de nodo único o principal/en standby no se puede ampliar. Si necesita una mayor capacidad, use una instancia de clúster en su lugar. Esta métrica solo está disponible para instancias de nodo único, principal/en standby y de Clúster Proxy. Para las instancias de Clúster Redis, esta métrica solo está disponible en el nivel de servidor Redis. Puede ver la métrica en la página de ficha **Redis Server** en la página **Performance Monitoring** de la instancia.

Métrica	Rango de valores	Política de alarmas	Límite superior de aproximación	Sugerencia sobre el manejo
Memory Usage	0–100%	Umbral de alarma: > 70% Número de períodos consecutivos: 2 Severidad de la alarma: crítica	No	Expanda la capacidad de la instancia.
Connected Clients	0–10,000	Umbral de alarma: > 8000 Número de períodos consecutivos: 2 Severidad de la alarma: mayor	No	Optimice el grupo de conexiones en el código de servicio para evitar que el número de conexiones exceda el límite máximo. Configure esta política de alarma en el nivel de instancia para las instancias de nodo único y principal/en standby. Para las instancias de clúster, configure esta política de alarma en el nivel de servidor Redis y Proxy. Para las instancias de nodo único y principal/en standby, el número máximo de conexiones permitidas es de 10,000. Puede ajustar el umbral en función de los requisitos de servicio.
New Connections (Recuento/min)	≥ 0	Umbral de alarma: > 10,000 Número de períodos consecutivos: 2 Severidad de la alarma: menor	-	Compruebe si se utiliza **connect** y si la conexión del cliente es anormal. Utilice conexiones persistentes ("**pconnect**" en la terminología de Redis) para garantizar el rendimiento. Configure esta política de alarma en el nivel de instancia para las instancias de nodo único y principal/en standby. Para las instancias de clúster, configure esta política de alarma en el nivel de servidor Redis y Proxy.

Métrica	Rango de valores	Política de alarmas	Límite superior de aproximación	Sugerencia sobre el manejo
Input Flow	≥ 0	Umbral de alarma: > 80% del ancho de banda asegurado Número de períodos consecutivos: 2 Severidad de la alarma: mayor	Sí	Considere la ampliación de la capacidad basada en el análisis del servicio y el límite de ancho de banda. Configure esta alarma solo para instancias de nodo único y principal/en standby de DCS Redis 3.0 y establezca el umbral de alarma al 80% del ancho de banda garantizado de instancias de DCS Redis 3.0.
Output Flow	≥ 0	Umbral de alarma: > 80% del ancho de banda asegurado Número de períodos consecutivos: 2 Severidad de la alarma: mayor	Sí	Considere la ampliación de la capacidad basada en el análisis del servicio y el límite de ancho de banda. Configure esta alarma solo para instancias de nodo único y principal/en standby de DCS Redis 3.0 y establezca el umbral de alarma al 80% del ancho de banda garantizado de instancias de DCS Redis 3.0.

Políticas de alarma para instancias de DCS Memcached

Tabla 13-10 Métricas de instancia de DCS Memcached para configurar reglas de alarma para

Métrica	Rango de valores	Política de alarmas	Límite superior de aproximación	Sugerencia sobre el manejo
CPU Usage	0–100%	Umbral de alarma: > 70% Número de períodos consecutivos: 2 Severidad de la alarma: mayor	No	Compruebe el servicio para ver si hay un aumento de tráfico. La capacidad de la CPU de una instancia de nodo único o principal/en standby no se puede ampliar. Analice el servicio y considere la posibilidad de dividir el servicio o combine varias instancias en un clúster en el extremo del cliente.
Memory Usage	0–100%	Umbral de alarma: > 65% Número de períodos consecutivos: 2 Severidad de la alarma: menor	No	Considere la posibilidad de ampliar la capacidad de la instancia.
Connected Clients	0–10,000	Umbral de alarma: > 8000 Número de períodos consecutivos: 2 Severidad de la alarma: mayor	No	Optimice el grupo de conexiones en el código de servicio para evitar que el número de conexiones exceda el límite máximo.

Métrica	Rango de valores	Política de alarmas	Límite superior de aproximación	Sugerencia sobre el manejo
New Connections	≥ 0	Umbral de alarma: > 10,000 Número de períodos consecutivos: 2 Severidad de la alarma: menor	-	Compruebe si se utiliza **connect** y si la conexión del cliente es anormal. Utilice conexiones persistentes ("pconnect" en la terminología de Redis) para garantizar el rendimiento.
Input Flow	≥ 0	Umbral de alarma: > 80% del ancho de banda asegurado Número de períodos consecutivos: 2 Severidad de la alarma: mayor	Sí	Considere la ampliación de la capacidad basada en el análisis del servicio y el límite de ancho de banda. Para obtener más información sobre los límites de ancho de banda de diferentes especificaciones de instancia, consulte **Especificaciones de instancia de DCS**.
Output Flow	≥ 0	Umbral de alarma: > 80% del ancho de banda asegurado Número de períodos consecutivos: 2 Severidad de la alarma: mayor	Sí	Considere la ampliación de la capacidad basada en el análisis del servicio y el límite de ancho de banda. Para obtener más información sobre los límites de ancho de banda de diferentes especificaciones de instancia, consulte **Especificaciones de instancia de DCS**.

Métrica	Rango de valores	Política de alarmas	Límite superior de aproximación	Sugerencia sobre el manejo
Authentication Failures	≥ 0	Umbral de alarma: > 0 Número de períodos consecutivos: 1 Severidad de la alarma: crítica	-	Compruebe si la contraseña se ha introducido correctamente.

Politicas de alarma para nodos de servidor Redis de instancias de clúster de DCS Redis

Tabla 13-11 Métricas del servidor Redis para configurar las políticas de alarma para

Métrica	Rango de valores	Política de alarmas	Límite superior de aproximación	Sugerencia sobre el manejo
CPU Usage	0–100%	Umbral de alarma: > 70% Número de períodos consecutivos: 2 Severidad de la alarma: mayor	No	Compruebe el servicio para ver si hay un aumento de tráfico. Compruebe si el uso de la CPU se distribuye uniformemente a los nodos de servidor de Redis. Si el uso de la CPU es alto en varios nodos, considere la expansión de la capacidad. La ampliación de la capacidad de una instancia de clúster reducirá los nodos para compartir la presión de la CPU. Si el uso de CPU es alto en un solo nodo, compruebe si existen las claves de mucho uso. En caso afirmativo, optimice el código de servicio para eliminar las claves de mucho uso.

Métrica	Rango de valores	Política de alarmas	Límite superior de aproximación	Sugerencia sobre el manejo
Average CPU Usage	0–100%	Umbral de alarma: > 70% Número de períodos consecutivos: 2 Severidad de la alarma: mayor	No	Considere la ampliación de la capacidad basada en el análisis del servicio. La capacidad de la CPU de una instancia de nodo único o principal/en standby no se puede ampliar. Si necesita una mayor capacidad, use una instancia de clúster en su lugar. Esta métrica solo está disponible para instancias de nodo único, principal/en standby y de Clúster Proxy. Para las instancias de Clúster Redis, esta métrica solo está disponible en el nivel de servidor Redis. Puede ver la métrica en la página de ficha **Redis Server** en la página **Performance Monitoring** de la instancia.
Memory Usage	0–100%	Umbral de alarma: > 70% Número de períodos consecutivos: 2 Severidad de la alarma: mayor	No	Compruebe el servicio para ver si hay un aumento de tráfico. Compruebe si el uso de memoria se distribuye uniformemente a los nodos del servidor Redis. Si el uso de la memoria es alto en varios nodos, considere la expansión de la capacidad. Si el uso de memoria es alto en un solo nodo, compruebe si existen las claves grandes. En caso afirmativo, optimice el código de servicio para eliminar las claves grandes.
Connected Clients	0–10,000	Umbral de alarma: > 8000 Número de períodos consecutivos: 2 Severidad de la alarma: mayor	No	Compruebe si el número de conexiones está dentro del rango adecuado. En caso afirmativo, ajuste el umbral de alarma.

Métrica	Rango de valores	Política de alarmas	Límite superior de aproximación	Sugerencia sobre el manejo
New Connections	≥ 0	Umbral de alarma: > 10,000 Número de períodos consecutivos: 2 Severidad de la alarma: menor	-	Compruebe si se utiliza **connect**. Para garantizar el rendimiento, utilice conexiones persistentes ("pconnect" en la terminología de Redis).
Slow Query Logs	0–1	Umbral de alarma: > 0 Número de períodos consecutivos: 1 Severidad de la alarma: mayor	-	Utilice la función de consulta lenta de la consola para analizar los comandos lentos.

Métrica	Rango de valores	Política de alarmas	Límite superior de aproximación	Sugerencia sobre el manejo
Bandwidth Usage	0–200%	Umbral de alarma: > 90% Número de períodos consecutivos: 2 Severidad de la alarma: mayor	Sí	Compruebe si el aumento del uso del ancho de banda proviene de los servicios de lectura o de los servicios de escritura basados en el flujo de entrada y salida. Si el uso de ancho de banda de un solo nodo es alto, compruebe si existen claves grandes. Incluso si el uso de ancho de banda excede del 100%, el control de flujo puede no realizarse necesariamente. El control de flujo real está sujeto a la métrica **Flow Control Times**. Incluso si el uso del ancho de banda es inferior al 100%, se puede realizar el control de flujo. El uso del ancho de banda en tiempo real se informa una vez en cada período de informe. La métrica de tiempos de control de flujo se informa cada segundo. Durante un período de informe, el tráfico puede aumentar en segundos y luego retroceder. En el momento en que se informa del uso del ancho de banda, se ha restaurado al nivel normal.
Flow Control Times	≥ 0	Umbral de alarma: > 0 Número de períodos consecutivos: 1 Severidad de la alarma: crítica	Sí	Considere la expansión de la capacidad basada en los límites de especificación, el flujo de entrada y el flujo de salida. **NOTA** Esta métrica solo es compatible con Redis 4.0 y 5.0 y no con Redis 3.0.

Políticas de alarma para nodos proxy de instancias de clúster de DCS Redis

Tabla 13-12 Métricas de proxy para configurar las políticas de alarma para

Métrica	Rango de valores	Política de alarmas	Límite superior de aproximación	Sugerencia sobre el manejo
CPU Usage	0–100%	Umbral de alarma: > 70% Número de períodos consecutivos: 2 Severidad de la alarma: crítica	Sí	Considere la ampliación de la capacidad, que agregará Proxy.
Memory Usage	0–100%	Umbral de alarma: > 70% Número de períodos consecutivos: 2 Severidad de la alarma: crítica	Sí	Considere la ampliación de la capacidad, que agregará Proxy.
Connected Clients	0–30,000	Umbral de alarma: > 20,000 Número de períodos consecutivos: 2 Severidad de la alarma: mayor	No	Optimice el grupo de conexiones en el código de servicio para evitar que el número de conexiones exceda el límite máximo.

Configuración de una regla de alarma para un grupo de recursos

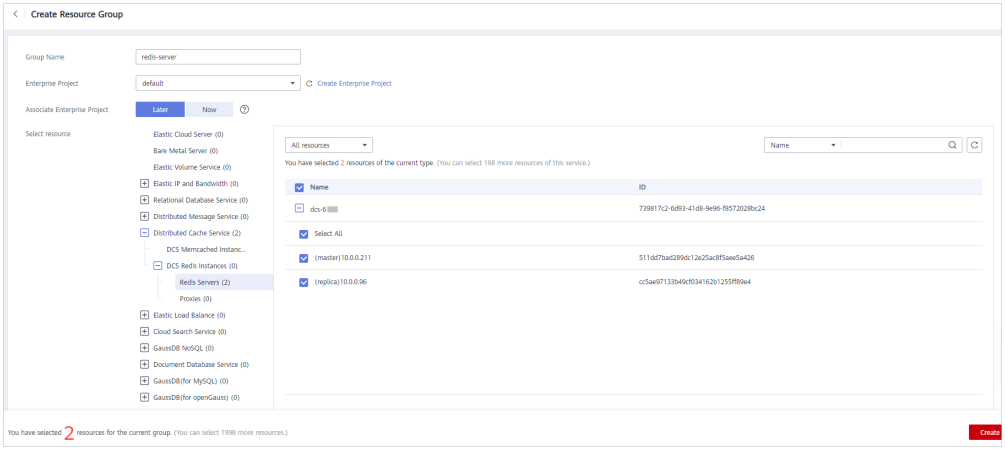
Cloud Eye le permite agregar las instancias de DCS, nodos de servidor Redis y nodos proxy a grupos de recursos y gestionar las instancias y reglas de alarma por grupo para simplificar la operación. Para obtener más información, consulte **Creación de un grupo de recursos**.

Paso 1 Crear un grupo de recursos.

1. Inicie sesión en la consola de Cloud Eye. En el panel de navegación, elija **Resource Groups** y, a continuación, haga clic en **Create Resource Group** en la esquina superior derecha.
2. Introduzca un nombre de grupo y agregue nodos de servidor de Redis al grupo de recursos.

   Puede agregar nodos de servidor de Redis de las instancias diferentes al mismo grupo de recursos.

   Figura 13-1 Creación de un grupo de recursos

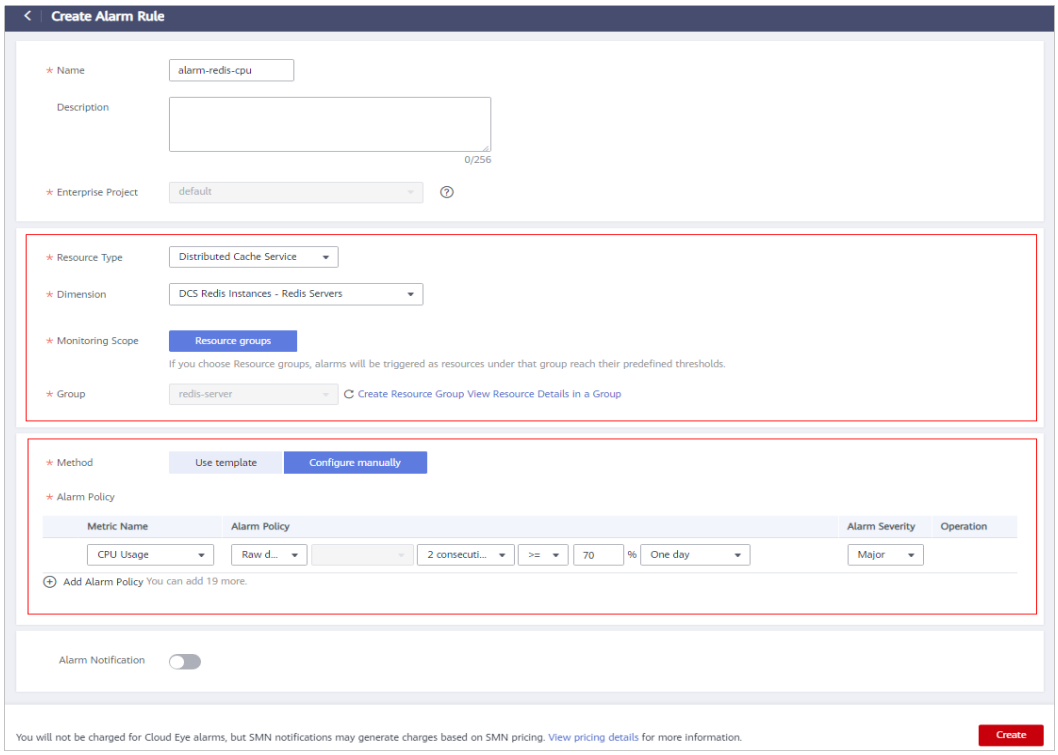


3. Haga clic en **Create**.

Paso 2 En el panel de navegación de la consola de Cloud Eye, elija **Alarm Management** > **Alarm Rules** y, a continuación, haga clic en **Create Alarm Rule** para establecer la información de alarma para el grupo de recursos.

Cree una regla de alarma de uso de CPU para todos los nodos de servidor de Redis del grupo de recursos, como se muestra en la siguiente figura.

Figura 13-2 Creación de una regla de alarma para un grupo de recursos



Paso 3 Haga clic en **Create**.

----Fin

Configuración de una regla de alarma para un recurso específico

En el ejemplo siguiente, se establece una regla de alarma para la métrica **Slow Query Logs** (**is_slow_log_exist**).

Paso 1 Inicie sesión en la **consola DCS**.

Paso 2 Haga clic en en la esquina superior izquierda de la consola de gestión y seleccione una región.

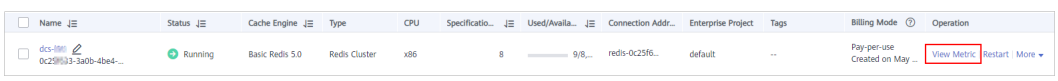
NOTA

Seleccione la misma región que su servicio de aplicación.

Paso 3 En el panel de navegación, elija **Cache Manager**.

Paso 4 En la fila que contiene la instancia de DCS cuyas métricas desea ver, haga clic en **View Metric** en la columna **Operation**.

Figura 13-3 Consulta de métricas de instancia



Paso 5 En la página mostrada, busque la métrica **Slow Query Logs**. Pase el cursor sobre la métrica y haga clic en para crear una regla de alarma para la métrica.

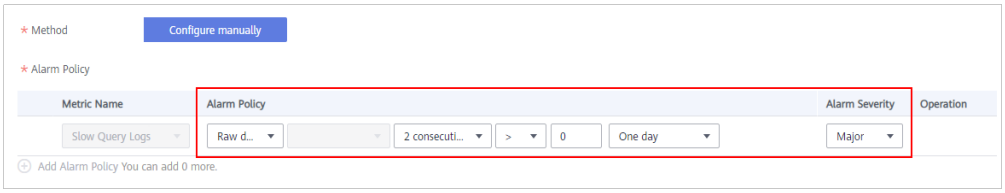
Se muestra la página **Create Alarm Rule**.

Paso 6 Especifique la información de alarma.

1. Establezca el nombre y la descripción de la alarma.
2. Especifique la política de alarma y la gravedad de la alarma.

   Por ejemplo, la política de alarma mostrada en **Figura 13-4** indica que se activará una alarma si existen consultas lentas en el caso durante dos períodos consecutivos. Si no se realizan acciones, la alarma se activará una vez al día, hasta que el valor de esta métrica vuelva a 0.

   Figura 13-4 Configuración del contenido de la alarma

3. Establezca las configuraciones de notificación de alarma. Si habilita **Alarm Notification**, establezca el período de validez, el objeto de notificación y la condición del activador.
4. Haga clic en **Create**.

   NOTA

   Para obtener más información sobre cómo crear reglas de alarma, consulte **Creación de una regla de alarma**.

----Fin

14 Auditoría

14.1 Operaciones registradas por CTS

Con CTS, puede consultar, auditar y revisar las operaciones realizadas en recursos en la nube. Las trazas incluyen las solicitudes de operación enviadas mediante la consola de gestión o las API abiertas, así como los resultados de estas solicitudes.

A continuación se enumeran las operaciones de DCS que pueden ser registradas por CTS.

Tabla 14-1 Operaciones de DCS que pueden ser grabadas por CTS

Operación	Tipo de recurso	Nombre del seguimiento
Creación de una instancia	Redis	createDCSInstance
Envío de una solicitud de creación de instancia	Redis	submitCreateDCSInstanceRequest
Eliminación de varias instancias	Redis	batchDeleteDCSInstance
Eliminación de una instancia	Redis	deleteDCSInstance
Modificación de información de instancia	Redis	modifyDCSInstanceInfo

Operación	Tipo de recurso	Nombre del seguimiento
Modificación de configuraciones de instancia	Redis	modifyDCSInstanceConfig
Cambio de contraseña de instancia	Redis	modifyDCSInstancePassword
Detención de una instancia	Redis	stopDCSInstance
Envío de una solicitud de detención de instancia	Redis	submitStopDCSInstanceRequest
Reinicio de una instancia	Redis	restartDCSInstance
Envío de una solicitud de reinicio de instancia	Redis	submitRestartDCSInstanceRequest
Inicio de una instancia	Redis	startDCSInstance
Envío de una solicitud de inicio de instancia	Redis	submitStartDCSInstanceRequest
Limpieza de datos de instancia	Redis	flushDCSInstance
Detención de varias instancias	Redis	batchStopDCSInstance
Envío de una solicitud para detener instancias por lotes	Redis	submitBatchStopDCSInstanceRequest
Reinicio de instancias por lotes	Redis	batchRestartDCSInstance

Operación	Tipo de recurso	Nombre del seguimiento
Envío de una solicitud para reiniciar instancias por lotes	Redis	submitBatchRestartDCSInstanceRequest
Inicio de varias instancias	Redis	batchStartDCSInstance
Envío de una solicitud para iniciar instancias por lotes	Redis	submitBatchStartDCSInstanceRequest
Restauración de datos de instancia	Redis	restoreDCSInstance
Envío de una solicitud para restaurar datos de instancia	Redis	submitRestoreDCSInstanceRequest
Copia de seguridad de datos de instancia	Redis	backupDCSInstance
Envío de una solicitud para realizar una copia de seguridad de los datos de instancia	Redis	submitBackupDCSInstanceRequest
Eliminación de archivos de copia de seguridad de instancia	Redis	deleteInstanceBackupFile
Eliminación de tareas en segundo plano	Redis	deleteDCSInstanceJobRecord
Modificación de especificaciones de instancia	Redis	modifySpecification

Operación	Tipo de recurso	Nombre del seguimiento
Envío de una solicitud para modificar las especificaciones de instancia	Redis	submitModifySpecificationRequest
Creación de un pedido de suscripción de instancia	Redis	createInstanceOrder
Creación de un pedido para modificar las especificaciones de instancia	Redis	createSpecificationChangeOrder
Actualización de ID de proyecto de empresa	Redis	updateEnterpriseProjectId
Cambio entre nodos principal y en standby	Redis	masterStandbySwitchover
Desactivación del acceso público	Redis	disablePublicNetworkAccess
Activación el acceso público	Redis	enablePublicNetworkAccess
Reajuste de la contraseña de instancia	Redis	resetDCSInstancePassword
Envío de una solicitud para borrar datos de instancia	Redis	submitFlushDCSInstanceRequest
Acceso a la Web CLI	Redis	webCliLogin
Ejecución de comandos en Web CLI	Redis	webCliCommand
Saliendo de Web CLI	Redis	webCliLogout

Operación	Tipo de recurso	Nombre del seguimiento
Migración de datos sin conexión	Redis	offlineMigrate
Cambio del modo de facturación	Redis	billingModeChange

14.2 Consulta de logs de auditoría

Para obtener más información acerca de cómo ver los registros de auditoría, véase **Consulta de seguimientos en tiempo real**.

