

Comprender la propiedad del servicio NSO

Contenido

[Introducción](#)

[Prerequisites](#)

[Requirements](#)

[Componentes Utilizados](#)

[Conceptos básicos](#)

[Dispositivo propio](#)

[Propiedad en acción](#)

[Escaparate: solo propiedad del servicio](#)

[Escaparate: solo el dispositivo es propiedad](#)

[Escaparate: propiedad del dispositivo + propiedad del servicio](#)

[Reconciliar](#)

[Propósito](#)

[Transferencia de propiedad a servicios](#)

[Quitar indicadores de valor originales](#)

[Corregir la propiedad del servicio](#)

[Reconciliar como volver a implementar](#)

[Reconciliar el funcionamiento interno](#)

[Problemas relacionados con la propiedad del servicio](#)

[Error de reconciliación](#)

[El servicio está eliminando la configuración no perteneciente al servicio](#)

[Escaparate: el servicio está eliminando la configuración que no pertenece al servicio](#)

[Keep-non-service-config vs Discard-non-service-config](#)

[No hay creación](#)

[Fusionar](#)

Introducción

Este documento describe el concepto general, los riesgos comunes y las soluciones de propiedad de servicios en Cisco® Network Service Orchestrator (NSO).

Prerequisites

Requirements

Este documento se aplica a todas las versiones de NSO disponibles actualmente, incluido NSO 6. El comportamiento descrito sólo es aplicable a las configuraciones de NSO que utilizan una combinación de servicios y configuración que no es de servicio. Aunque los comandos específicos utilizados en los ejemplos de este documento sólo se aplican al controlador de

elementos de red (NED) utilizado, la lógica subyacente se aplica a cualquier dispositivo administrado por NSO.

Componentes Utilizados

- NSO 6.1.6
- NED: test-ned 1.0 , un Netconf NED personalizado que utiliza este modelo Yang.

NED yang model:

```
module test-ned{
  namespace "http://example.org/ned/service-ownership";
  prefix ownership;
  import ietf-inet-types{ prefix inet;}

  list interface {
    key interface-name;
    leaf interface-name{
      type string;
    }

    leaf ip-address {
      type inet:ipv4-address;
    }

    leaf description {
      type string;
    }
  }
}
```

- Servicio: example-service 1.0: un servicio personalizado que utiliza este modelo y plantilla yang

```
module example-service {
  namespace "http://com/example/example-service";
  prefix example-service;

  import ietf-inet-types {
    prefix inet;
  }
  import tailf-ncs {
    prefix ncs;
  }

  list example-service {
    key name;

    uses ncs:service-data;
    ncs:servicepoint "example-service";

    leaf name {
      type string;
    }
  }
}
```

```
leaf-list device {  
  type leafref {  
    path "/ncs:devices/ncs:device/ncs:name";  
  }  
}  
leaf ipaddress {  
  type inet:ipv4-address;  
}  
}  
}
```

{/device}

FE1

{/ipaddress}

La información que contiene este documento se creó a partir de los dispositivos en un ambiente de laboratorio específico. Todos los dispositivos que se utilizan en este documento se pusieron en funcionamiento con una configuración verificada (predeterminada). Si tiene una red en vivo, asegúrese de entender el posible impacto de cualquier comando.

Conceptos básicos

El objetivo de la propiedad del servicio es permitir que NSO realice un seguimiento de la configuración relacionada con cada servicio. Cuando se elimina un servicio, NSO necesita eliminar la configuración relacionada con él y utiliza la propiedad del servicio para determinar qué configuración se debe eliminar. Cuando la configuración pertenece a más de un servicio, la eliminación de uno de los servicios simplemente elimina esa referencia de propiedad. La configuración en sí permanece en la base de datos de NSO (CDB) y en los dispositivos de la red.

La propiedad se muestra a través de refcounts y backpointers. Un refcount muestra cuántas entidades poseen esa parte de la configuración. Un refcount es igual a la cantidad de instancias de servicio más 1 si la configuración también es propiedad del dispositivo. Un puntero de retroceso muestra la ruta de acceso de estas instancias de servicio. No hay ningún puntero de retroceso para mostrar "device owned" (dispositivo propiedad). Los backpointers sólo se muestran en CDB para listas y contenedores. Las hojas individuales no muestran sus backpointers pero heredan de su padre.

Dispositivo propio

Además de que la configuración sea propiedad de un servicio, también puede ser propiedad del dispositivo. Esto se denomina a veces "propiedad del dispositivo" o "no propiedad del servicio". Aunque este documento utiliza "propiedad del dispositivo", tenga en cuenta que, aunque esto es más fácil de entender, la propiedad ajena al servicio no tiene que incluir los dispositivos. Las configuraciones de LSA o los servicios apilados pueden tener una configuración no perteneciente al servicio sin dispositivos.

La configuración es propiedad del dispositivo cuando la configuración se agrega a CDB sin el uso de una implementación de servicio, pero en su lugar se utiliza un método como sync-from, load merge o ncs_cli para establecer la configuración. Cuando una instancia de servicio toma posesión de una configuración que ya era propiedad del dispositivo, el refcount se establece en 2 para reflejar la propiedad compartida. Cuando se elimina la instancia de servicio, la configuración no se elimina a pesar de que solo una instancia de servicio posee la configuración antes de la eliminación. Además, la configuración propiedad del dispositivo agrega una etiqueta de "valor original". Si una instancia de servicio sobrescribe la configuración del dispositivo y el servicio se elimina posteriormente, la configuración se restaura al valor original.

La propiedad del dispositivo solo se asigna si la configuración no existía ya en CDB cuando se agrega a través de medios que no son de servicio. La configuración propiedad del servicio no se convierte en propiedad del dispositivo después de la sincronización desde. Sin embargo, la configuración propiedad del dispositivo pasa a ser propiedad del dispositivo y del servicio si implementa un servicio en la parte superior.

Propiedad en acción

Escaparate: solo propiedad del servicio

Cuando se implementa un servicio mientras la configuración de destino está vacía, el servicio crea la configuración y toma posesión de ella. El usuario puede verificar la propiedad usando un comando `show running-config` y anexar `| display service-meta-data`. Aunque no es obligatorio, se recomienda adjuntar también `| La visualización de xml` como la salida de estilo CLI predeterminada no siempre refleja correctamente cómo se modelan los datos en CDB.

```
admin@ncs(config)# do show running-config devices device mydevice0 config
% No entries found.
admin@ncs(config)# example-service s1 device mydevice0 ipaddress 192.0.2.1
admin@ncs(config-example-service-s1)# commit dry-run
cli {
  local-node {
    data devices {
      device mydevice0 {
        config {
          +       interface FE1 {
          +           ip-address 192.0.2.1;
          +       }
        }
      }
    }
  }
  +example-service s1 {
  +  device [ mydevice0 ];
  +  ipaddress 192.0.2.1;
  +}
}
}
admin@ncs(config-example-service-s1)# commit
Commit complete.
admin@ncs(config-example-service-s1)# do show running-config devices device mydevice0 config | display
```

mydevice0

FE1

192.0.2.1

Además, si agrega una segunda instancia de servicio cuyo destino es la misma configuración, la propiedad se comparte entre las dos instancias de servicio. El Refcount es 2 y hay 2 backpointers.

```
admin@ncs(config-example-service-s1)# example-service s2 device mydevice0 ipaddress 192.0.2.2
admin@ncs(config-example-service-s2)# commit dry-run
cli {
  local-node {
    data devices {
      device mydevice0 {
        config {
          interface FE1 {
            - ip-address 192.0.2.1;
            + ip-address 192.0.2.2;
```

```
        }
    }
}
+example-service s2 {
+  device [ mydevice0 ];
+  ipaddress 192.0.2.2;
+}
}
}
admin@ncs(config-example-service-s2)# commit
Commit complete.
admin@ncs(config-example-service-s2)# do show running-config devices device mydevice0 config | display
```

mydevice0

FE1

192.0.2.2

Escaparate: solo el dispositivo es propiedad

Cuando agrega datos a CDB mediante la combinación de carga, ncs_cli o sync-from, sin utilizar un servicio, estos datos pasan a ser propiedad del dispositivo. El refcount y el backpointer están ocultos.

```
admin@ncs(config)# no example-service
admin@ncs(config)# commit
Commit complete.
admin@ncs(config)# do show running-config devices device mydevice0 config
% No entries found.
admin@ncs(config)# load merge merge-config.xml
Loading.
386 bytes parsed in 0.00 sec (137.22 KiB/sec)
admin@ncs(config)# commit dry-run
cli {
  local-node {
    data devices {
      device mydevice0 {
        config {
          +       interface FE1 {
          +         ip-address 192.0.2.1;
          +       }
        }
      }
    }
  }
}
admin@ncs(config)# commit
Commit complete.
admin@ncs(config)# do show running-config devices device mydevice0 config | display service-meta-data |
```

FE1

192.0.2.1

Escaparate: propiedad del dispositivo + propiedad del servicio

En este ejemplo se muestra cómo crear fácilmente la propiedad combinada de dispositivos y servicios mediante un servicio y la sincronización desde.

El servicio se implementa y, a continuación, se elimina solo en CDB mediante Commit no-networking. De esta manera, la configuración aún existe en el dispositivo final. Al realizar la sincronización desde, la configuración se vuelve a agregar en CDB, pero es propiedad del dispositivo en lugar del servicio. Recuerde que show running-config cuando se ejecuta en NSO muestra los datos de CDB, no los datos que se encuentran actualmente en los dispositivos.

```
admin@ncs(config)# no devices device mydevice0 config
admin@ncs(config)# commit
admin@ncs(config)# do show running-config devices device mydevice0 config
% No entries found.
```

```
admin@ncs(config)# example-service s1 device mydevice0 ipaddress 192.0.2.1
admin@ncs(config-example-service-s1)# commit
admin@ncs(config-example-service-s1)# top
admin@ncs(config)# no example-service s1
admin@ncs(config)# commit no-networking
Commit complete.
admin@ncs(config)# devices device mydevice0 sync-from
result true
admin@ncs(config)# do show running-config devices device mydevice0 config | display service-meta-data |
```

mydevice0

FE1

192.0.2.1

Después de la sincronización desde, la configuración solo es propiedad del dispositivo. Refcounter está oculto. Cuando se implementa el servicio de nuevo, el refcount se convierte en 2, pero backpointer sólo muestra una instancia de servicio. El segundo refcounter representa la propiedad del dispositivo. Se aplican las mismas reglas que con la propiedad de servicio compartido. Si se elimina el servicio, la configuración no se elimina porque el dispositivo también posee parcialmente la configuración. Además, si los datos del servicio no coinciden con el "valor original" almacenado en los metadatos del servicio, NSO devuelve el valor al "valor original" si se elimina el servicio.

```
admin@ncs(config)# example-service s1 device mydevice0 ipaddress 192.0.2.2
admin@ncs(config-example-service-s1)# commit dry-run
cli {
  local-node {
    data devices {
      device mydevice0 {
        config {
          interface FE1 {
            - ip-address 192.0.2.1;
            + ip-address 192.0.2.2;
          }
        }
      }
    }
  }
  +example-service s1 {
  + device [ mydevice0 ];
  + ipaddress 192.0.2.2;
  +}
}
}
admin@ncs(config-example-service-s1)# commit
Commit complete.
admin@ncs(config-example-service-s1)# do show running-config devices device mydevice0 config | display
```

mydevice0

FE1

192.0.2.2

Reconciliar

Sintaxis: <path-to-service instance> re-deploy reconcile

Indicadores opcionales: { keep-non-service-config } dry-run { outformat native }

Propósito

El objetivo principal de la funcionalidad de reconciliación es permitir que los usuarios se deshagan de la propiedad de dispositivos no deseados y transfieran la propiedad completamente a los servicios. Cuando los usuarios ya disponen de una red en funcionamiento y están intentando transferir la propiedad a NSO, normalmente introducen primero la configuración mediante operaciones de sincronización desde. Una vez que CDB tiene toda la configuración de red, el usuario implementa instancias de servicio sobre la configuración existente. En este momento, la configuración sigue siendo propiedad del dispositivo, lo que limita la capacidad del servicio para eliminar la configuración. Cuando los usuarios desean dar a sus servicios plena propiedad de la

configuración, pueden utilizar la funcionalidad de reconciliación que hace 3 cosas.

- 1) Transferir la propiedad a los servicios
- 2) 2) Eliminar indicadores de "valor original"
- 3) 3) Corrección de la propiedad del servicio

Transferencia de propiedad a servicios

Reconcile evalúa toda la configuración propiedad de un servicio y, si encuentra cualquier configuración que sea propiedad tanto de este servicio como del dispositivo o que no sea propiedad del servicio, elimina esta propiedad del dispositivo, convirtiendo el servicio en el propietario exclusivo. Reducir el descuento en 1.

Nota: Si 2 servicios poseen una configuración y esa configuración no pertenece a ningún servicio, el descuento es 3. La conciliación de cualquiera de los servicios elimina la propiedad ajena al servicio, lo que reduce el reembolso a 2 para reflejar ambos servicios.

Quitar indicadores de valor originales

Cuando se implementa un servicio y se sobrescriben los datos que no pertenecen al servicio, refcount se convierte en 2 y NSO agrega una etiqueta de "valor original". Si la instancia de servicio se elimina alguna vez, NSO intenta volver al valor original que existía antes del servicio.

Durante la reconciliación, no sólo se reduce el descuento, sino que se elimina el valor original. Al eliminar el servicio, el valor se vacía o se cambia a un valor predeterminado, tal como se define en el modelo Yang.

Corregir la propiedad del servicio

En algunas situaciones, la propiedad se puede asignar incorrectamente. El servicio es propietario incorrectamente de la configuración del dispositivo o el servicio y el dispositivo son propietarios incorrectos de la configuración, mientras que se espera que solo sea propiedad del servicio. Reconcile puede corregir estos desajustes. Esto es importante para evitar problemas en los que el servicio elimina la configuración que no pertenece al servicio.

Reconciliar como volver a implementar

La reconciliación es una función secundaria de la reimplementación de servicios. Si un servicio no está sincronizado con CDB, la operación de reconciliación también ejecuta una nueva implementación además de realizar la función de reconciliación.

Reconciliar el funcionamiento interno

Aunque los desarrolladores solo conocen los detalles exactos de cómo funciona reconcile, este documento proporciona una comprensión simplificada:

- 1) NSO corrige la propiedad del servicio
- 2) NSO elimina del CDB toda la configuración perteneciente a esta instancia de servicio, incluso si también pertenece a otros servicios o es propiedad del dispositivo
- 3) NSO vuelve a implementar la instancia de servicio
- 4) NSO restaura los reembolsos de servicios y los backpointers de otros servicios

Problemas relacionados con la propiedad del servicio

Error de reconciliación

En caso de que encuentre que la "re-implementación" funciona pero la "re-implementación reconcile" falla: Esto puede indicar que ha encontrado un conflicto entre su diseño de servicio y la forma en que funciona la función de reconciliación.

El problema surge del código de servicio que intenta leer la configuración de CDB que el servicio implementa más tarde. Sólo puede implementar este servicio porque la configuración ya existe parcialmente en CDB antes de la implementación. Sin embargo, durante la reconciliación, NSO elimina temporalmente toda la configuración propiedad de este servicio, incluida la configuración que el servicio intenta leer durante la reimplementación del servicio en el siguiente paso. Esto suele dar lugar a un error de Java o Python que informa de la falla de lectura de los datos.

El servicio está eliminando la configuración no perteneciente al servicio

En esta situación, se produce la eliminación por parte de NSO de una configuración que no es propiedad del servicio durante la eliminación o reimplementación de la instancia del servicio. Esto se debe a que la instancia de servicio creó y posee la configuración original, y posteriormente se agregó manualmente (a través de `ncs_cli`, `sync-from` u otro método) una parte de la configuración dentro de una lista o contenedor propiedad del servicio.

Se supone que esta nueva configuración no es propiedad del servicio, pero como el servicio posee la propiedad total del contenedor o lista, el servicio termina siendo propietario indirecto de la misma.

La manera de resolver esto es usando `re-deploy reconcile { keep-non-service-config }` para corregir la propiedad del servicio. Al realizar esta conciliación, el nuevo recuento del contenedor o lista aumenta para reflejar que este contenedor o lista tiene nodos secundarios tanto de servicios como de no servicios. Dentro del nodo principal, sólo los nodos propiedad del servicio tienen un `refcount` y un puntero de retorno.

Escaparate: el servicio está eliminando la configuración que no pertenece al servicio

A partir de una única instancia de servicio implementada con plena propiedad, agregue manualmente una descripción a la interfaz mediante `ncs_cli`.

```
admin@ncs(config-example-service-s1)# do show running-config devices device mydevice0 config | display
```

```
mydevice0
```

```
FE1
```

```
192.0.2.1
```

```
admin@ncs(config-example-service-s1)# top
admin@ncs(config)# devices device mydevice0 config interface FE1 description "This is an example descri
admin@ncs(config-interface-FE1)# commit
Commit complete.
```

```
admin@ncs(config-interface-FE1)# do show running-config devices device mydevice0 config | display servi
```

mydevice0

FE1

192.0.2.1

This is an example description

Observe cómo el refcount en <interface> sigue siendo 1 aunque se agregó la configuración propiedad del dispositivo. Al intentar eliminar la instancia de servicio, la descripción también se elimina aunque no se supone que forme parte de la instancia de servicio. Para evitar esto, puede ejecutar el comando reconcile.

```
admin@ncs(config-interface-FE1)# top
admin@ncs(config)# no example-service s1
admin@ncs(config)# commit dry-run
cli {
  local-node {
    data devices {
      device mydevice0 {
        config {
          -       interface FE1 {
          -         ip-address 192.0.2.1;
          -         description "This is an example description";
          -       }
        }
      }
    }
  }
  -example-service s1 {
  -  device [ mydevice0 ];
  -  ipaddress 192.0.2.1;
  -}
}
}
admin@ncs(config)# abort
admin@ncs# config
Entering configuration mode terminal
admin@ncs(config)# example-service s1 re-deploy reconcile { keep-non-service-config }
admin@ncs(config)# do show running-config devices device mydevice0 config | display service-meta-data |
```

mydevice0

FE1

192.0.2.1

This is an example description

Después de la reconciliación, el refcount para la interfaz de lista ha aumentado a 2. Mientras tanto, refcount on leaf ip-address, ha permanecido 1. La entrada de la lista "interfaz FE1" contiene tanto datos propiedad del servicio como datos no propiedad del servicio. Al utilizar la función de reconciliación, NSO vuelve a evaluar la propiedad y asigna los reembolsos en consecuencia. La eliminación ahora solo se dirige a las áreas que son totalmente propiedad de la instancia de servicio. Ni la descripción ni la entrada de la lista se eliminan.

```
admin@ncs(config)# no example-service s1
admin@ncs(config)# commit dry-run
cli {
  local-node {
```

```

data devices {
    device mydevice0 {
        config {
            interface FE1 {
-                ip-address 192.0.2.1;
            }
        }
    }
}
-example-service s1 {
- device [ mydevice0 ];
- ipaddress 192.0.2.1;
-}
}
}

```

Keep-non-service-config vs Discard-non-service-config

A veces, los usuarios no entienden bien el uso de discard-non-service-config.

En el ejemplo de reconciliación, se utilizó "keep-non-service-config". Si se utiliza el método discard, el aspecto es el siguiente:

```

admin@ncs(config)# example-service s1 re-deploy reconcile { discard-non-service-config } dry-run
cli {
    local-node {
        data devices {
            device mydevice0 {
                config {
                    interface FE1 {
-                        description "This is an example description";
                    }
                }
            }
        }
    }
}
}
}

```

El valor predeterminado es "keep-non-service-config". Si ninguna de las opciones está definida, NSO la mantiene de forma predeterminada. Rara vez se utiliza el método de descarte, ya que la mayoría de los usuarios prefieren conservar lo que hay en su red, aunque NSO no lo administre. Reconcile { discard-non-service-config } dry-run se puede utilizar para averiguar qué puntos de datos existen en CDB que no forman parte de la configuración del servicio, pero se eliminarían si el servicio se eliminara o se volviera a implementar.

No hay creación

Una alternativa al uso de la función de reconciliación de reimplementación para corregir la propiedad del servicio cuando se mezcla con datos que no son propiedad del servicio es evitar el

conflicto mediante la etiqueta nocreate.

Se trata de una etiqueta que se puede utilizar en la plantilla de servicio XML. La documentación indica "ncreate: la combinación sólo afecta a los elementos de configuración que ya existen en la plantilla. Nunca crea la configuración con esta etiqueta. Solo modifica las estructuras de configuración existentes".

El uso de esta etiqueta tiene un efecto secundario interesante: debido a que el servicio no crea el nodo, no toma posesión de él.

Esto se suele utilizar para evitar situaciones en las que NSO intenta eliminar una configuración que el dispositivo no permite eliminar.

Tenga en cuenta que esta etiqueta es heredada por los nodos secundarios, esto significa que si agrega la etiqueta nocreate a la interfaz, también se aplica a cualquier nodo dentro de esa interfaz a menos que los marque con una etiqueta diferente como merge"

Agregue una etiqueta nocreate a la plantilla de servicio. Si la interfaz FE1 no existe, no se configura nada.

```
{/device}
```

FE1

```
{/ipaddress}
```

Vuelva a compilar y a cargar el paquete y, a continuación, pruébelo.

```
admin@ncs(config)# do show running-config devices device mydevice0 config | display service-meta-data |
No entries found.
admin@ncs(config)# example-service s1 device mydevice0 ipaddress 192.0.2.1
admin@ncs(config-example-service-s1)# commit dry-run
cli {
  local-node {
    data +example-service s1 {
      +   device [ mydevice0 ];
      +   ipaddress 192.0.2.1;
      +}
    }
  }
}
```

Aunque se definieron los mismos parámetros que antes, la interfaz o cualquier configuración subyacente no se crean en la configuración del dispositivo.

Fusionar

Agregue una etiqueta merge en la configuración dentro de la interfaz. No agregue una etiqueta a "interface-name" ya que esta es la clave de la interfaz de lista. La clave siempre debe poder heredar el comportamiento de la lista. Vuelva a compilar y a cargar el paquete.

```
{/device}
```

```
FE1
```

```
{/ipaddress}
```

Configure manualmente la interfaz FE1 antes de implementar el servicio.

```
admin@ncs(config)# no example-service
admin@ncs(config)# commit
admin@ncs(config)# do show running-config devices device mydevice0 config | display service-meta-data |
No entries found.
```

```

admin@ncs(config)# devices device mydevice0 config interface FE1
admin@ncs(config-interface-FE1)# commit dry-run
cli {
  local-node {
    data devices {
      device mydevice0 {
        config {
          +       interface FE1 {
          +       }
        }
      }
    }
  }
}
admin@ncs(config-interface-FE1)# commit
Commit complete.
admin@ncs(config-interface-FE1)# top
admin@ncs(config)# example-service s1 device mydevice0 ipaddress 192.0.2.1
admin@ncs(config-example-service-s1)# commit dry-run
cli {
  local-node {
    data devices {
      device mydevice0 {
        config {
          +       interface FE1 {
          +         ip-address 192.0.2.1;
          +       }
        }
      }
    }
    +example-service s1 {
    +  device [ mydevice0 ];
    +  ipaddress 192.0.2.1;
    +}
  }
}
admin@ncs(config-example-service-s1)# commit
Commit complete.
admin@ncs(config-example-service-s1)# do show running-config devices device mydevice0 config | display

```

mydevice0

FE1

192.0.2.1

La interfaz tiene un refcount 1 oculto: la interfaz fue implementada usando ncs_cli, pero tiene una etiqueta nocreate en el paquete de servicio; el servicio no tomó posesión. Es propiedad del dispositivo.

El principal tiene un descuento 1: solo es propiedad del servicio

Si se elimina la instancia de servicio, solo se eliminará ipaddress, ya que es la única parte que pertenece totalmente al servicio.

Acerca de esta traducción

Cisco ha traducido este documento combinando la traducción automática y los recursos humanos a fin de ofrecer a nuestros usuarios en todo el mundo contenido en su propio idioma.

Tenga en cuenta que incluso la mejor traducción automática podría no ser tan precisa como la proporcionada por un traductor profesional.

Cisco Systems, Inc. no asume ninguna responsabilidad por la precisión de estas traducciones y recomienda remitirse siempre al documento original escrito en inglés (insertar vínculo URL).