Implementación de DNS VNF con red SRIOV en Openstack CVIM - Ejemplo de configuración para Prime Network Registrar (DNS)

Contenido

Introducción

Este documento describe la implementación paso a paso de CPNR en OpenStack Cisco Virtualized Infrastructure Manager (CVIM) mediante SR-IOV y la vinculación de Active-Backup.

Prerequisites

Requirements

Cisco recomienda que tenga conocimiento sobre estos temas:

- Familiaridad con OpenStack y los conceptos de virtualización de entrada/salida de raíz única (SR-IOV)
- Conocimientos prácticos sobre redes y comandos de Cisco Virtual Interface Manager (VIM) y Cisco Elastic Services Controller y Linux

Componentes Utilizados

Este documento no tiene restricciones específicas en cuanto a versiones de software y de hardware.

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

Antecedentes

En el panorama actual de las redes, las funciones de red virtual (VNF) desempeñan un papel fundamental a la hora de habilitar servicios de red ágiles, escalables y eficientes. Para las VNF que requieren conectividad de red de alto rendimiento, SR-IOV es una tecnología de uso común. SR-IOV permite que las VNF eludan el switch virtual del hipervisor y accedan directamente a los recursos físicos del controlador de interfaz de red (NIC), lo que reduce la latencia y aumenta el rendimiento.

Configuración

Antes de continuar con la implementación, asegúrese de que se cumplen estos requisitos previos.

1. Requisitos de hardware

- NIC compatibles con SR-IOV:
  - Al menos dos NIC físicas compatibles con SR-IOV con SR-IOV habilitado en la BIOS/interfaz de firmware extensible unificada (UEFI).
  - Ejemplo:sriov0asignado al nodo 0 de acceso a memoria no uniforme (NUMA) y sriov1asignado al nodo 1 de NUMA.

- Hosts con reconocimiento de NUMA:
  - Los nodos informáticos deben admitir la arquitectura NUMA.
  - La compatibilidad con NUMA debe estar habilitada en el BIOS/UEFI del host.

2. Identificación de tarjetas NIC Intel

Las tarjetas NIC Intel XL710 y E810CQDA2 se utilizan habitualmente para las redes SR-IOV de alto rendimiento. Para verificar el modelo de tarjeta NIC en el host, consulte estos pasos:

Paso 1. Uso del comando lspci

Ejecute este comando para enumerar los dispositivos de interconexión de componentes periféricos (PCI) relacionados con los controladores de red:

```
lspci | grep -i ethernet
```

Ejemplo de salida:

```
81:00.0 Ethernet controller: Intel Corporation Ethernet Controller XL710 for 40GbE QSFP+ (rev 02)
82:00.0 Ethernet controller: Intel Corporation Ethernet Controller E810-C for QSFP (rev 03)
```

Paso 2. Verificación del XL710

Si la NIC es Intel XL710, puede ver Controlador Ethernet XL710 en la salida.

Paso 3. Verificación de E810CQDA2

Si la NIC es Intel E810CQDA2, puede ver la salida Ethernet Controller E810-Cin.

Paso 4. Confirmación de la compatibilidad del controlador

Para verificar el controlador NIC en uso, ejecute:

```
ethtool -i
```

Ejemplo de salida para XL710:

```
driver: i40e
version: 2.13.10
```

Ejemplo de salida para E810CQDA2:

```
driver: ice
version: 1.7.12
```

Asegúrese de que la versión del controlador coincida con la matriz de compatibilidad de su distribución OpenStack y Linux.

3. Configuración de BIOS/UEFI

- Habilitar SR-IOV:

Asegúrese de que SR-IOV esté habilitado en el BIOS/UEFI de los servidores.

- Habilitación de la tecnología de virtualización para E/S dirigida (VT-d)/AMD-Vi:

Intel VT-d o AMD-Vi deben estar activados para el paso a través de PCI y la funcionalidad SR-IOV.

4. Configuración de OpenStack

- Servicios principales de OpenStack:

Asegúrese de que los servicios de OpenStack como Nova, Neutron, Glance y Keystone estén instalados y configurados.

- Configuración de neutrones:

Neutron debe admitir Openvswitch (OVS) para redes de orquestación/gestión y SR-IOV para redes de aplicaciones/servicios.

- Configuración de SR-IOV:

Los nodos informáticos deben configurarse para admitir SR-IOV, con funciones virtuales (VF) creadas en las NIC.

5. Imagen de VNF de Cisco Prime Network Registrar (CPNR)

- Compatibilidad de imagen VNF:

La imagen de VNF de CPNR debe admitir interfaces SR-IOV e incluir los controladores necesarios.

- Cargar en la guía rápida:

Asegúrese de que la imagen de CPNR VNF esté disponible en OpenStack Glance.

6. Acceso administrativo

- CLI de OpenStack:

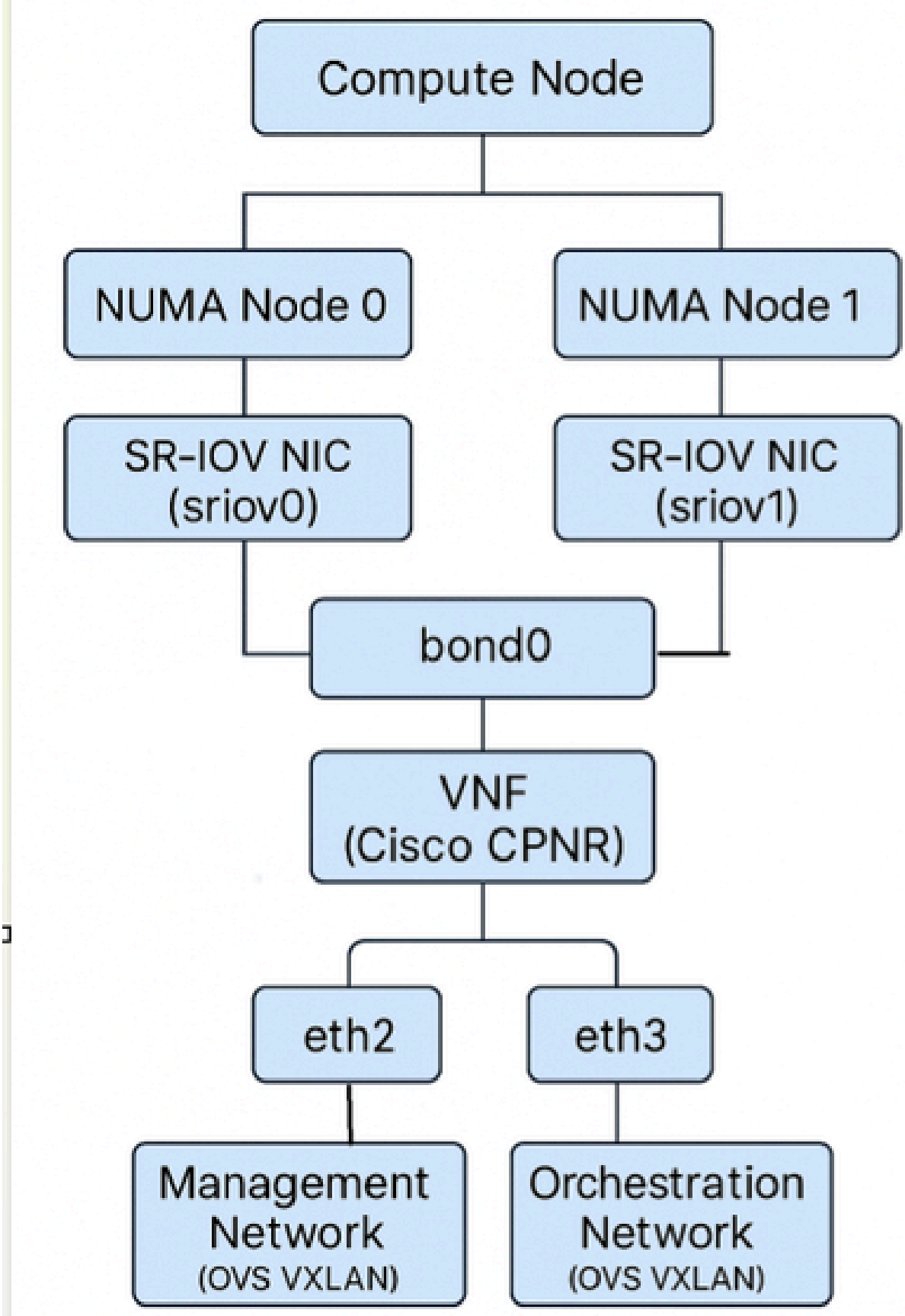
Garantizar el acceso a la CLI de OpenStack para crear redes, tipos e iniciar VNF.

- Privilegios de administrador o raíz:

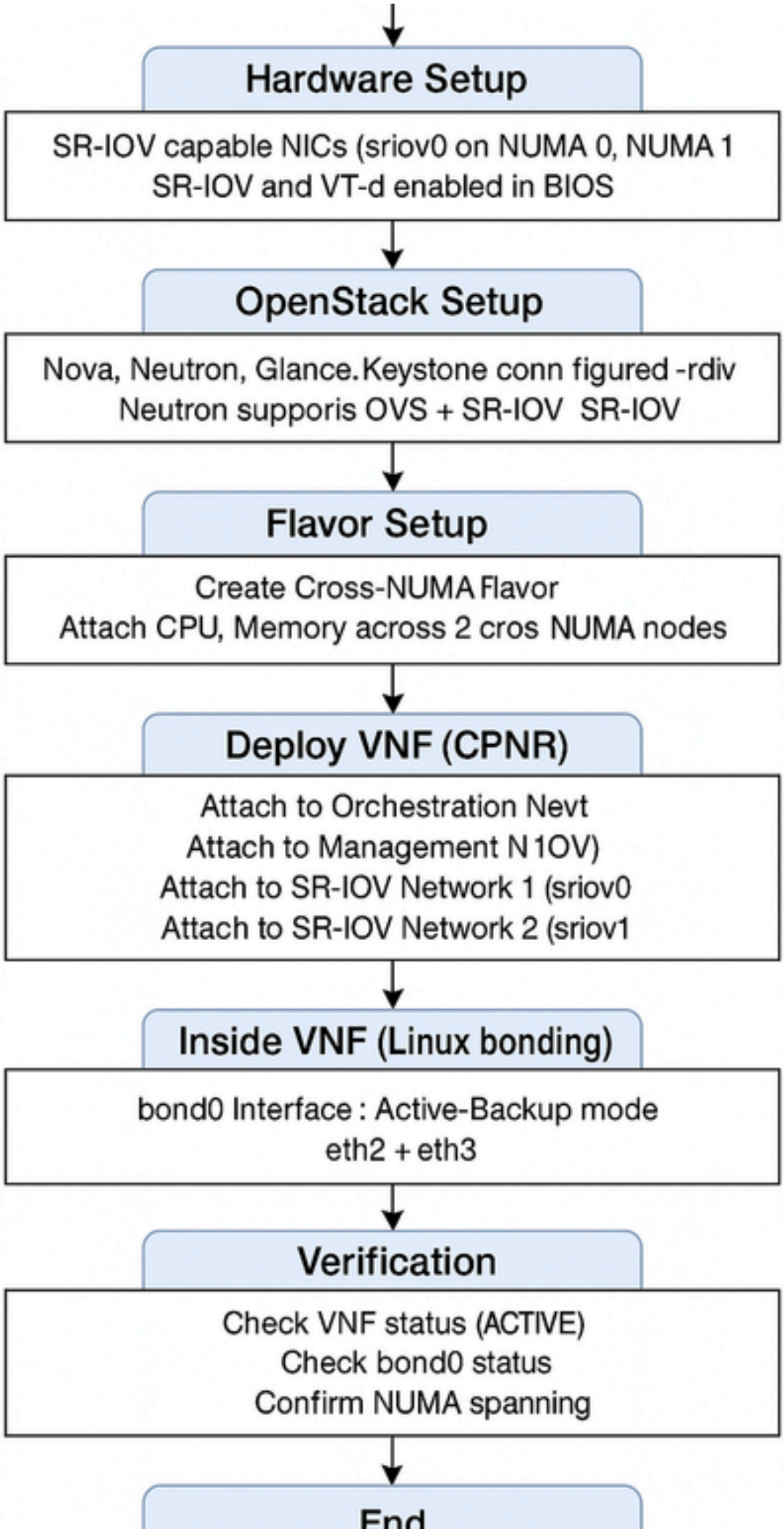
Acceso raíz o administrativo para configurar la red en el host Linux y dentro de VNF.

Descripción general de la arquitectura

Diagrama de conectividad de interfaz de red VNF



Organigrama


Hardware Setup
SR-IOV capable NICs (sriov0 on NUMA 0, NUMA 1
SR-IOV and VT-d enabled in BIOS
OpenStack Setup
Nova, Neutron, Glance.Keystone conn figured -rdiv
Neutron supporis OVS + SR-IOV SR-IOV
Flavor Setup
Create Cross-NUMA Flavor
Attach CPU, Memory across 2 cros NUMA nodes
Deploy VNF (CPNR)
Attach to Orchestration Nevt
Attach to Management N1OV)
Attach to SR-IOV Network 1 (sriov0
Attach to SR-IOV Network 2 (sriov1
Inside VNF (Linux bonding)
bond0 Interface : Active-Backup mode
eth2 + eth3
Verification
Check VNF status (ACTIVE)
Check bond0 status
Confirm NUMA spanning
End

2. Enlace de copia de seguridad activa:

    - Se crea una interfaz bond0 utilizando NIC SR-IOV (eth2 de sriov0 y eth3 de sriov1).
    - El modo de copia de seguridad activa garantiza la redundancia y la tolerancia a fallos sin necesidad de configuraciones en el switch.

3. Redes OpenStack:

    - Redes de orquestación y gestión: Basado en OpenVswitch para el tráfico administrativo y de control.
    - Redes de aplicaciones/servicios: Basado en SR-IOV para tráfico de alto rendimiento.

Por qué es necesario el modo Cross-NUMA

1. Redes con reconocimiento de NUMA en OpenStack

NUMA es una arquitectura de memoria en la que cada CPU (y su memoria y dispositivos locales) se agrupa en un nodo NUMA. En OpenStack, la ubicación con reconocimiento de NUMA garantiza que las VNF se asignen de forma óptima a los recursos del mismo nodo de NUMA para minimizar la latencia y maximizar el rendimiento.

- Las NIC SR-IOV son locales de NUMA:
    - Cada NIC física está vinculada a un nodo NUMA específico. Por ejemplo:
        - sriov0 está conectado al nodo NUMA 0.
        - sriov1 está conectado al nodo NUMA 1.
- Limitación del Modo Single-NUMA:
    - Cuando se inicia un VNF en modo de NUMA único, OpenStack sólo permite que el VNF se conecte a NIC locales del nodo NUMA donde se inicia el VNF. Esto significa:
        - Si el VNF se inicia en NUMA 0, sólo puede conectarse a NIC en sriov0.
        - Si la VNF se inicia en NUMA 1, solo puede conectarse a NIC en sriv1.

2. Por qué es necesario el modo Cross-NUMA

CPNR VNF requiere acceso a:

- Red de orquestación (Openvswitch, independiente de NUMA)
- Red de gestión(Openvswitch, independiente de NUMA)
- Red SR-IOV 1: Conectado a sriov0(nodo NUMA 0)
- Red SR-IOV 2: Conectado a sriov1(nodo NUMA 1).

En esta implementación, la VNF de CPNR requiere acceso a las NIC SR-IOV tanto de NUMA 0 (sriov0) como de NUMA 1 (sriov1) para proporcionar redundancia y alta disponibilidad. Para lograr esto:

- El VNF debe iniciarse en el modo NUMA cruzado, lo que permite a OpenStack asignar CPU, memoria y NIC desde múltiples nodos NUMA.
- Esto garantiza que VNF pueda conectarse a NIC en sriov0 y sriov1, lo que permite el uso de ambos puertos SR-IOV en una configuración de enlace de copia de seguridad activa.

Limitación de tamaño de pista para puertos OVS

¿Qué es Conntrack?

Contrack es una función del núcleo de Linux que se utiliza para realizar un seguimiento de las conexiones de red, especialmente para la traducción de direcciones de red (NAT) y las reglas de firewall. En el caso de los puertos basados en OVS en OpenStack, conntrack se utiliza para administrar el estado de la conexión y aplicar reglas de grupos de seguridad.

Cómo Afecta Conntrack a los Puertos OVS

1. Tabla de seguimiento:

    - Cada conexión activa consume una entrada en la tabla de conexiones.
    - El tamaño de la tabla conntrack está limitado por el parámetro f_conntrack_max.

2. Límite predeterminado:

    - De forma predeterminada, el tamaño de la tabla conntrack es de 65536 entradas. En el caso de cargas de trabajo con altas velocidades de conexión (por ejemplo, VNF con muchos flujos simultáneos), este límite se puede agotar rápidamente, lo que provoca la pérdida de paquetes.

3. Impacto en los puertos de OVS:

    - Si la tabla conntrack está llena, se descartan nuevas conexiones, lo que puede afectar seriamente el rendimiento de VNF.
    - Esto es especialmente relevante para las redes de orquestación ygestión, que utilizan puertos OVS.

Cómo reducir las limitaciones de Conntrack

1. Aumentar tamaño de tabla de seguimiento:

    - Ver el límite actual:

```
sysctl net.netfilter.nf_conntrack_max
```

- Aumentar el límite:

```
sysctl -w net.netfilter.nf_conntrack_max=262144
```

- Haga que el cambio sea persistente:

```
echo "net.netfilter.nf_conntrack_max=262144" >> /etc/sysctl.conf
```

2. Supervisar el uso de Conntrack:

   Comprobar estadísticas de seguimiento:

```
cat /proc/sys/net/netfilter/nf_conntrack_count
```

3. Optimice las reglas de grupos de seguridad:

   Reduzca el número de reglas aplicadas a los puertos OVS para minimizar la sobrecarga de pista de conexión.

Cómo resuelve SR-IOV los problemas de seguimiento

1. Elimina la dependencia de Conntrack

Los puertos SR-IOV omiten la ruta de datos OVS y las funciones del núcleo Linux como conntrack. Esto elimina por completo la sobrecarga de seguimiento de conexión.

2. Mayor escalabilidad

A diferencia de los puertos OVS, que están limitados por el tamaño de la tabla de conexiones (nf_conntrack_max), los puertos SR-IOV pueden gestionar un número prácticamente ilimitado de conexiones.

3. Latencia reducida

Al descargar el procesamiento de paquetes al hardware NIC, los puertos SR-IOV eliminan la latencia que introduce el procesamiento de seguimiento basado en software.

Por qué se elige el modo de copia de seguridad activa para los

puertos SR-IOV en la máquina virtual CPNR

El modo de vinculación de copia de seguridad activa es especialmente adecuado para esta implementación debido a su simplicidad, tolerancia a errores y compatibilidad con las interfaces SR-IOV. He aquí por qué:

1. Redundancia sin complejidad

- Modo de copia de seguridad activa: Sólo una interfaz (la interfaz activa) transmite y recibe tráfico en un momento dado. Las otras interfaces permanecen en modo de espera.
- Si la interfaz activa falla (por ejemplo, debido a una falla de link o a un problema de hardware), el enlace cambia automáticamente a una interfaz en espera. Esto garantiza una conectividad de red continua sin necesidad de intervención manual.

2. No se requiere ningún grupo de agregación de enlaces (LAG)

- A diferencia de otros modos de vinculación (por ejemplo, 802.3adorbalance-alb), el modo de copia de seguridad activa no requiere el protocolo de control de agregación de enlaces (LACP) ni configuraciones de switch.
- Esto es especialmente importante para los puertos SR-IOV, ya que los VF SR-IOV normalmente no admiten configuraciones LACP o LAG.

3. Conmutación por fallo perfecta

- La conmutación por fallo es casi instantánea, con una interrupción mínima del tráfico.
- Cuando la interfaz activa falla, el enlace promueve inmediatamente una interfaz en espera al estado activo.

4. Independencia del hardware

El modo de copia de seguridad activa funciona independientemente del hardware o los switches físicos subyacentes. La lógica de conmutación por fallas reside completamente en el kernel de Linux, lo que lo hace altamente portátil y versátil.

5. Optimizado para SR-IOV

Las VF SR-IOV están vinculadas a NIC físicas específicas y nodos NUMA. Mediante el modo Active-Backup, puede combinar VF de diferentes nodos NUMA en una única interfaz de enlace lógico (bond0). Esto garantiza una alta disponibilidad al tiempo que se hace un uso eficiente de los recursos de NUMA.

El modo de copia de seguridad activa es uno de los modos más simples y ampliamente utilizados en la vinculación de Linux. Está diseñado para proporcionar alta disponibilidad asegurando que el tráfico continúe fluyendo sin problemas incluso si falla una de las interfaces vinculadas. Esta es una explicación detallada de cómo funciona el modo de copia de seguridad activa, sus características clave y ventajas.

¿Qué es una interfaz de Linux Bond?

Una interfaz de enlace en Linux combina dos o más interfaces de red en una sola interfaz lógica. Esta interfaz lógica, denominada enlace (por ejemplo, bond0), se utiliza para proporcionar:

- Redundancia: Garantizar una alta disponibilidad de la conectividad de red.
- Mejora del rendimiento: En otros modos (por ejemplo, balance-error802.3ad), también puede agregar ancho de banda.

¿Cómo Funciona El Modo De Copia De Seguridad Activa?

En el modo de copia de seguridad activa, sólo se utiliza una interfaz (denominada interfaz activa) en un momento dado para transmitir y recibir tráfico. Las otras interfaces permanecen en modo de espera. Si la interfaz activa falla, una de las interfaces en espera pasa al estado activo y el tráfico se redirige automáticamente a la nueva interfaz activa.

Características clave del modo de copia de seguridad activa

1. Interfaz activa única:

    - En un momento dado, solo una interfaz física en el enlace está activa para transmitir y recibir tráfico.
    - Las interfaces en espera son completamente pasivas a menos que ocurra una falla.

2. Conmutación por fallo automática:

    - Si la interfaz activa falla (por ejemplo, debido a un problema de hardware, desconexión de cable o falla de link), el link cambia automáticamente a una interfaz en espera.
    - La conmutación por fallo es perfecta y no requiere intervención manual.

3. Soporte de Failover:

    Una vez restaurada la interfaz fallida, puede volver a activarse automáticamente (si está configurada para ello) o permanecer en modo de espera, según la configuración de vinculación.

4. Sin requisitos en el switch:

    - A diferencia de otros modos de vinculación (por ejemplo, 802.3adorbalance-rr), el modo Active-Backup no requiere ninguna configuración especial en los switches físicos (por ejemplo, LAG o LACP).
    - Esto lo hace ideal para escenarios donde la configuración del lado del switch no es posible o cuando se vinculan funciones virtuales SR-IOV, que no admiten LAG.

5. Control:

    - El enlace supervisa continuamente el estado de todas las interfaces miembro mediante el parámetro `emimon` (Media Independent Interface Monitor, Monitor de interfaz independiente de medios).
    - Si se detecta una falla de link, el link pasa inmediatamente a una interfaz en espera saludable.

Cómo fluye el tráfico en el modo de copia de seguridad activa

Funcionamiento normal

1. Interfaz activa:

    - El tráfico fluye exclusivamente a través de la interfaz activa (por ejemplo, eth2en un enlace ofeth2andeth3).
    - La interfaz en espera (eth3) permanece inactiva y no transmite ni recibe tráfico.

2. Control:

    - El enlace monitorea periódicamente el estado de todas las interfaces miembro. Esto se realiza mediante:
        - miimon: Comprueba el estado del enlace de cada interfaz en un intervalo configurable (por ejemplo, cada 100 ms).
        - Supervisión del protocolo de resolución de direcciones (ARP) (opcional): Envía solicitudes ARP para asegurarse de que la interfaz activa sea accesible.

Escenario de Failover

1. Falla de link en la interfaz activa:

    Si la interfaz activa (eth2) falla (por ejemplo, cable desconectado, falla de hardware NIC o link caído), el link detecta inmediatamente la falla usando monitoreo ARP mínimo.

2. Conmutación por fallo automática:

    - El enlace cambia a la interfaz standby (eth3), que se convierte en la nueva interfaz activa.
    - El tráfico se redirige a través de la nueva interfaz activa sin que sea necesaria la intervención manual.

3. Tiempo de conmutación por fallas:

    El proceso de conmutación por fallo es casi instantáneo (normalmente en unos pocos milisegundos, dependiendo delintervalo de minuto).

Escenario de recuperación

1. Restauración de la interfaz fallida:
    - Cuando se restaura la interfaz que falló anteriormente (eth2), puede:
        - Recuperar automáticamente el rol activo (si se ha configurado para ello).
        - Permanecer en modo de espera (comportamiento predeterminado).
2. Continuidad del tráfico:

    La conmutación por recuperación es perfecta, lo que garantiza que no se interrumpirán los flujos de tráfico en curso.

caso de uso: Enlace de copia de seguridad activa con puertos SR-IOV

El modo de copia de seguridad activa es especialmente adecuado para las interfaces SR-IOV porque:

- Las VF de SR-IOV no suelen admitir protocolos de agregación de enlaces como LACP.
- El enlace en el modo de copia de seguridad activa puede proporcionar redundancia sin ninguna configuración del lado del switch.

Por ejemplo:

- eth2se mapea a un SR-IOV VF onsriov0(nodo NUMA 0).
- eth3se asigna a un VF Onsriov 1 SR-IOV (nodo NUMA 1).
- El enlace (bond0) combina estas interfaces, proporcionando una conmutación por fallo perfecta entre VF SR-IOV.

Paso 1. Redes OpenStack

La VNF CPNR requiere estas cuatro redes:

1. Red de orquestación: Para el tráfico de control y orquestación (basado en Openvswitch).
2. Red de gestión: Para acceso administrativo (basado en Openvswitch).
3. Red SR-IOV 1: Tráfico de aplicaciones/servicios en sriov0.
4. Red SR-IOV 2: Tráfico de aplicaciones/servicios en sriv1.

Implementación paso a paso:

Paso 1.1. Creación De Redes Openvswitch

- Red de orquestación:

```
openstack network create --provider-network-type vxlan orchestration-network
```

- Red de gestión:

```
openstack network create --provider-network-type vxlan management-network
```

Paso 1.2. Creación de subredes para redes Openvswitch

- Subred de orquestación:

```
openstack subnet create --network orchestration-network \
--subnet-range 192.168.100.0/24 orchestration-subnet
```

- Subred de administración:

```
openstack subnet create --network management-network \
--subnet-range 10.10.10.0/24 management-subnet
```

Paso 1.3. Creación de redes SR-IOV

- Red SR-IOV 1:

```
openstack network create --provider-network-type vlan \
--provider-physical-network sriov0 --provider-segment 101 sriov-network-1
```

- Red SR-IOV 2:

```
openstack network create --provider-network-type vlan \
--provider-physical-network sriov1 --provider-segment 102 sriov-network-2
```

Paso 2. Sabores de OpenStack

Paso 2.1. Crear un sabor NUMA cruzado

Para asegurarse de que VNF pueda acceder a las NIC SR-IOV desde ambos nodos NUMA, cree un sabor con compatibilidad con NUMA cruzada:

```
openstack flavor create --ram 8192 --vcpus 4 --disk 40 cross-numa-flavor
```

Paso 2.2. Configuración de las propiedades de NUMA

Establecer propiedades específicas de NUMA:

```
openstack flavor set cross-numa-flavor \
--property hw:numa_nodes=2 \
--property hw:cpu_policy=dedicated \
--property hw:mem_page_size=large
```

Paso 3. Configure el enlace en el modo de copia de seguridad activa

Después de iniciar VNF, configure la interfaz de enlace para los puertos SR-IOV (eth2 y eth3) en VNF.

Paso 3.1. Configuración de la interfaz de enlace

Cree una interfaz de enlace (bond0) en el modo Active-Backup:

```
vi /etc/sysconfig/network-scripts/ifcfg-bond0
```

```
DEVICE=bond0
BOOTPROTO=static
ONBOOT=yes
BONDING_OPTS="mode=active-backup miimon=100"
IPADDR=172.16.1.10
```

```
NETMASK=255.255.255.0
GATEWAY=172.16.1.1
```

Paso 3.2. Configuración de interfaces esclavas

- eth2:

```
vi /etc/sysconfig/network-scripts/ifcfg-eth2
```

```
DEVICE=eth2
ONBOOT=yes
MASTER=bond0
SLAVE=yes
```

- eth3:

```
vi /etc/sysconfig/network-scripts/ifcfg-eth3
```

```
DEVICE=eth3
ONBOOT=yes
MASTER=bond0
SLAVE=yes
```

Paso 3.3. Aplicar configuración

Reinicie el servicio de red para aplicar la configuración:

```
systemctl restart network
```

Verificación

Después de implementar VNF, verifique su funcionalidad mediante los siguientes pasos:

1. Verificar el estado de VNF

Compruebe que la instancia de VNF está activa:

```
openstack server show cpnr-instance
```

Asegúrese de que el estado es ACTIVE.

2. Verifique la conectividad de red

- Prueba de ping: Verifique que VNF pueda comunicarse en todas las redes:

  ```
  ping
  ```

  ```
  ping
  ```

- Interfaz de enlace:
  - Confirme que bond0 está activo:

    ```
    cat /proc/net/bonding/bond0
    ```

    Busque:

    - Esclavo activo actualmente: Indica la interfaz activa.
    - Interfaz esclava: Confirma que tantoth2como th3 son parte del bono.

3. Verificar la ubicación de NUMA

Asegúrese de que VNF esté utilizando recursos de ambos nodos NUMA:

```
nova show

        --human | grep numa
```

Mejores medidas

- Supervisión y solución de problemas: Utilice herramientas como etcpdumpandethtool para supervisar las interfaces SR-IOV.
- Seguridad: Gestione cuidadosamente el acceso a la red física y aplique un aislamiento estricto entre los arrendatarios.
- Escalabilidad: Planifique la capacidad de NIC física al ampliar las implementaciones de SR-IOV, ya que el número de VF disponibles está limitado por el hardware de NIC.

Troubleshoot

Si la implementación no funciona como se esperaba, consulte estos pasos de solución de problemas:

1. Verificar la configuración de SR-IOV

- Compruebe si SR-IOV está habilitado en el BIOS:

```
dmesg | grep -i "SR-IOV"
```

- Confirmar que las VF se crean en las NIC:

```
lspci | grep Ethernet
```

2. Verificar la ubicación de NUMA

Si VNF no puede acceder a ambas NIC, asegúrese de que el modo NUMA cruzado esté habilitado:

- Compruebe las propiedades NUMA del sabor:

```
openstack flavor show cross-numa-flavor
```

3. Problemas de interfaz de bonos

- Compruebe el estado de la unión:

```
cat /proc/net/bonding/bond0
```

- Si el enlace no funciona:
  - Asegúrese de que las interfaces esclavas (eth2y eth3) estén correctamente configuradas como parte del enlace.
  - Reinicie el servicio de red:

```
systemctl restart network
```

4. Problemas de conectividad de red

- Verifique las vinculaciones de puerto de OpenStack:

```
openstack port list --server cpnr-instance
```

- Compruebe si la configuración IP en el VNF es correcta:

```
ip addr show
```

Conclusión

La implementación de CPNR VNF en OpenStack con puertos SR-IOV requiere el modo NUMA cruzado para permitir que VNF se conecte a NIC desde ambos nodos NUMA. Esto es esencial porque OpenStack restringe las VNF en modo de NUMA único para que solo accedan a los recursos (NIC, CPU, memoria) dentro del nodo NUMA donde se inicia la VNF. La combinación del

modo de NUMA cruzado con la vinculación de copia de seguridad activa garantiza una alta disponibilidad, tolerancia a fallos y una utilización eficiente de los recursos, lo que hace que esta implementación sea altamente resistente y eficaz.

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).