

在Openstack CVIM上使用SRIOV網路部署DNS VNF - Prime Network Registrar(DNS)組態範例

目錄

[簡介](#)

[必要條件](#)

[需求](#)

[採用元件](#)

[背景資訊](#)

[組態](#)

[1.硬體要求](#)

[2.識別英特爾網絡卡](#)

[步驟1.使用lspci命令](#)

[步驟2.檢驗XL710](#)

[步驟3.檢驗E810CQDA2](#)

[步驟4.確認驅動程式支援](#)

[3. BIOS/UEFI配置](#)

[4. OpenStack安裝程式](#)

[5. Cisco Prime Network Registrar\(CPNR\)VNF映像](#)

[6.行政准入](#)

[架構概觀](#)

[VNF網路介面連線圖](#)

[流程圖](#)

[配置示例](#)

[要點](#)

[在OpenStack上使用SR-IOV埠和活動備份繫結介面部署CPNR VNF](#)

[部署的主要功能](#)

[為什麼需要跨NUMA模式](#)

[1. OpenStack中的NUMA感知網路](#)

[2.為什麼必須採用跨網路運營模式](#)

[OVS埠的Contrack大小限制](#)

[什麼是Contrack?](#)

[Contrack對OVS埠的影響](#)

[如何緩解連線限制](#)

[SR-IOV如何解決連線問題](#)

[1.消除連線跟蹤相關性](#)

[2.更高的可擴充性](#)

[3.減少延遲](#)

[為什麼為CPNR VM上的SR-IOV埠選擇活動備份模式](#)

[1.無複雜性的冗餘](#)

[2.無需鏈路聚合組\(LAG\)](#)

[3.無縫故障切換](#)

[4.硬體獨立](#)

[5.針對SR-IOV進行了最佳化](#)

[什麼是Linux Bond介面？](#)

[Active-Backup模式如何工作？](#)

[Active-Backup模式的主要功能](#)

[流量在活動備份模式中的傳輸方式](#)

[正常運作](#)

[故障轉移方案](#)

[故障恢復方案](#)

[使用案例:與SR-IOV埠的主用 — 備用繫結](#)

[步驟1. OpenStack網路](#)

[步驟1.1.建立Openvswitch網路](#)

[步驟1.2.為Openvswitch網路建立子網](#)

[步驟1.3.建立SR-IOV網路](#)

[步驟2. OpenStack風格](#)

[步驟2.1.建立跨NUMA口味](#)

[步驟2.2.配置NUMA屬性](#)

[步驟3.在活動備份模式下配置繫結](#)

[步驟3.1.繫結介面配置](#)

[步驟3.2.配置從介面](#)

[步驟3.3.應用配置](#)

[驗證](#)

[1.檢驗VNF狀態](#)

[2.檢驗網路連線](#)

[3.驗證NUMA位置](#)

[最佳實踐](#)

[疑難排解](#)

[1.檢驗SR-IOV配置](#)

[2.驗證NUMA位置](#)

[3.債券介面發行](#)

[4.網路連線問題](#)

[結論](#)

簡介

本檔案介紹使用SR-IOV和主動備份繫結在OpenStack Cisco Virtualized Infrastructure Manager(CVIM)上逐步部署CPNR。

必要條件

需求

思科建議您瞭解以下主題：

- 熟悉OpenStack和單根輸入/輸出虛擬化(SR-IOV)概念
- 有關Cisco Virtual Interface Manager(VIM)和Cisco Elastic Services Controller和Linux命令和網路的工作知識

採用元件

本文件所述內容不限於特定軟體和硬體版本。

本文中的資訊是根據特定實驗室環境內的裝置所建立。文中使用到的所有裝置皆從已清除（預設）的組態來啟動。如果您的網路運作中，請確保您瞭解任何指令可能造成的影響

背景資訊

在當前網路環境中，虛擬網路功能(VNF)在實現靈活、可擴展且高效的網路服務方面發揮著關鍵作用。對於需要高效能網路連線的VNF，SR-IOV是一項常用的技術。SR-IOV允許VNF繞過虛擬機器監控程式虛擬交換機，直接訪問物理網路介面控制器(NIC)資源，從而減少延遲並提高吞吐量。

組態

繼續進行部署之前，請確保滿足這些前提條件。

1.硬體要求

- 支援SR-IOV的NIC：
 - 至少兩個SR-IOV功能的物理NIC，在BIOS/統一可擴展韌體介面(UEFI)中啟用SR-IOV。
 - 示例：sriov0對映到Non-Uniform Memory Access(NUMA)節點0,sriov1對映到NUMA節點1。
- NUMA感知主機：
 - 計算節點必須支援NUMA架構。
 - 必須在主機BIOS/UEFI中啟用NUMA支援。

2.識別英特爾網路卡

英特爾XL710和E810CQDA2 NIC卡通常用於高效能SR-IOV網路。若要驗證主機上的NIC卡型號，請參閱以下步驟：

步驟1.使用lspci命令

執行以下命令列出與網路控制器相關的外圍元件互連(PCI)裝置：

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

輸出示例：

```
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)
```

步驟2.檢驗XL710

如果網絡卡是Intel XL710，您可以在輸出中看到Ethernet Controller XL710。

步驟3.檢驗E810CQDA2

如果網絡卡是Intel E810CQDA2，您可以在輸出中看到乙太網控制器E810-Cin。

步驟4.確認驅動程式支援

要檢查正在使用的NIC驅動程式，請運行：

```
ethtool -i
```

XL710的輸出示例：

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

E810CQDA2的輸出示例：

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

確保驅動程式版本與OpenStack和Linux分發版的相容性清單相匹配。

3. BIOS/UEFI配置

- 啟用SR-IOV:

確保在伺服器BIOS/UEFI中啟用SR-IOV。

- 為定向I/O(VT-d)/AMD-Vi啟用虛擬化技術：

必須啟用Intel VT-d或AMD-Vi才能實現PCI直通和SR-IOV功能。

4. OpenStack安裝程式

- 核心OpenStack服務：

確保安裝和配置了OpenStack服務，例如Nova、Neutron、Glance和Keystone。

- 中子配置：

Neutron必須同時支援用於協調/管理網路的Openvswitch(OVS)和用於應用/服務網路的SR-IOV。

- SR-IOV配置：

必須將計算節點配置為支援SR-IOV，並在NIC上建立虛擬功能(VF)。

5. Cisco Prime Network Registrar(CPNR)VNF映像

- VNF映像相容性：

CPNR VNF映像必須支援SR-IOV介面並包含必需的驅動程式。

- 上傳至概覽：

確保OpenStack概覽中提供了CPNR VNF映像。

6.行政准入

- OpenStack CLI:

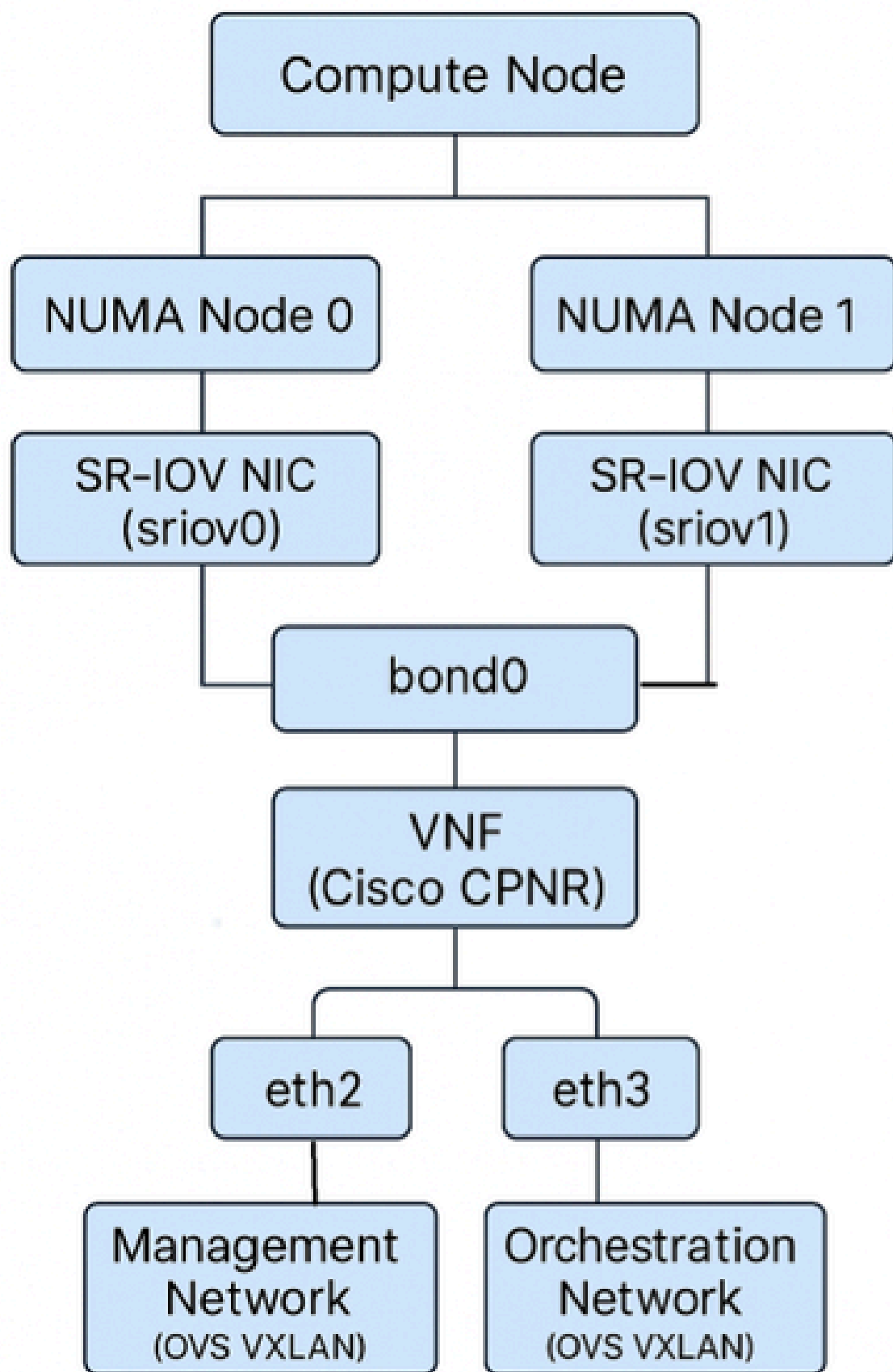
確保訪問OpenStack CLI以建立網路、風格和啟動VNF。

- Root或Admin許可權：

用於在Linux主機和VNF內配置網路的根或管理訪問許可權。

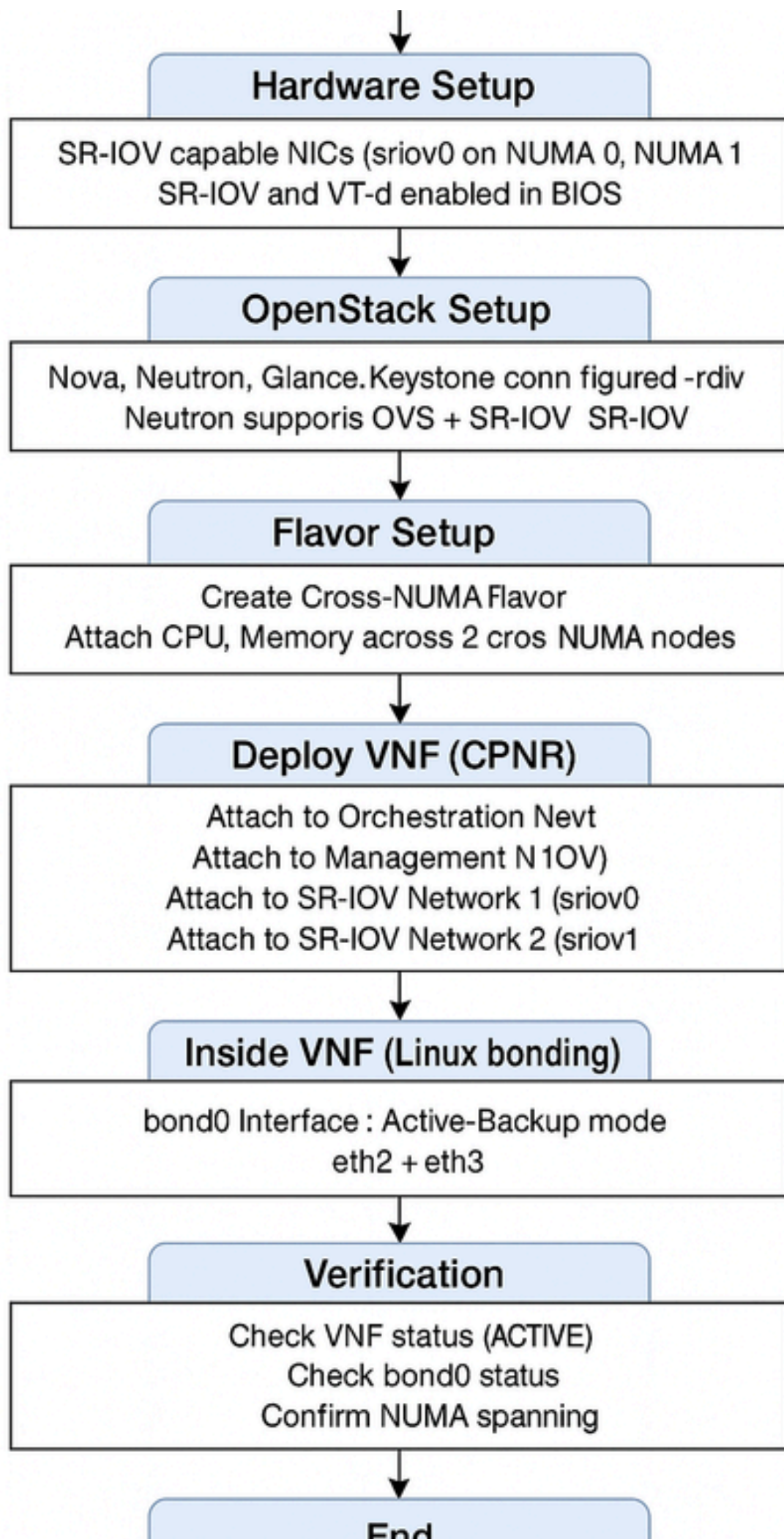
架構概觀

VNF網路介面連線圖



2

流程圖



test-tenant

true

false

sriov-vm-deployment

sriov-vm-1-group

vim1

default

sriov-image

custom-flavor

300

30

REBOOT_ONLY

0

mgmt-net

192.168.10.101

1

direct

srhov-net-1

0

sriov-subnet-1

10.10.10.10

2

direct

sriov-net-2

0

sriov-subnet-2

10.10.20.10

1

1

false

--user-data

file://tmp/init/sriov-vm-1.cfg

要點

- type>direct</type>under<interface>為該NIC啟用SR-IOV (PCI通過) 。
- 每個SR-IOV介面都有自己的網路和子網。
- 您可以根據需要在<addresses>中關聯IPv4/IPv6。

要通過Cisco ESC XML傳遞的Day0檔案示例：

Content-Type: multipart/mixed; boundary="====2678395050260980330=="
MIME-Version: 1.0`

--====2678395050260980330==

```
MIME-Version: 1.0
Content-Type: text/cloud-boothook; charset="us-ascii"
```

```
#cloud-boothook
#!/bin/bash
if [ ! -f /etc/cloud/cloud.cfg.orig ]; then
cp /etc/cloud/cloud.cfg /etc/cloud/cloud.cfg.orig
cp /etc/cloud/cloud.cfg.norootpasswd /etc/cloud/cloud.cfg
fi
```

```
-----2678395050260980330==
```

```
MIME-Version: 1.0
Content-Type: text/cloud-config; charset="us-ascii"
```

```
#cloud-config
hostname: cpnr
ssh_authorized_keys:
- ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQAC7pf8gv0WH/Zv8iA1Tv6LWEiPGA3B6t96G6LwTHF6iX0qQxyIUkg8IkqZ6wNwx
- ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQADAmkQGCZUrYqkZ0C0J9t7mF9La9zY0qfzzFkk1wWtPga+aAN0aFgjqbjj+VlBd
runcmd:
- /usr/sbin/useradd FMLVL1 -d /home/FMLVL1 -s /bin/bash -g users; (/bin/echo changeme; /bin/echo chang
- nmcli con add type ethernet con-name eth0 ifname eth0 ip4 10.xx.xx.xx/24
- nmcli con add type ethernet con-name eth1 ifname eth1 ip4 172.xx.xx.xx/23
- nmcli connection add type bond con-name bond0 ifname bond0 bond.options "mode=active-backup,miimon=1
- nmcli connection add type ethernet ifname ens6 master bond0
- nmcli connection add type ethernet ifname ens7 master bond0
- nmcli con up eth0
- nmcli con up eth1
- nmcli con up bond-slave-ens6
- nmcli con up bond-slave-ens7
- nmcli con down bond0
- nmcli con up bond0
- nmcli connection reload
- hostnamectl set-hostname CPNRDNS01CO
```

```
-----2678395050260980330==
```

在OpenStack上使用SR-IOV埠和活動備份繫結介面部署CPNR VNF

CPNR是一種重要的虛擬網路功能(VNF)，為企業和服務提供商網路提供IP地址管理(IPAM)、DHCP和域名伺服器(DNS)服務。在OpenStack中將CPNR部署為VNF需要仔細規劃，特別是在利用SR-IOV埠、跨NUMA配置和Active-Backup繫結介面以實現冗餘和效能時。

本文分步說明了在OpenStack上部署CPNR VNF的過程。它包括：

- 配置跨NUMA模式，這對於從多個NUMA節點訪問SR-IOV NIC至關重要。
- 設定Active-Backup繫結，確保高可用性，而無需交換機端配置。
- 配置OpenStack網路、口味和概覽。
- IP地址規劃、使用ifcfg檔案配置Linux網路，以及使用Cisco ESC部署VNF。

部署的主要功能

1. 跨NUMA意識：

- CPNR VNF跨越NUMA節點以訪問SR-IOV NIC(NUMA 0上的sriov0, NUMA 1上的sriov1)。
- 需要跨NUMA模式，因為在單NUMA模式下，OpenStack僅允許VNF連線到物理上位於啟動VNF的相同NUMA節點上的NIC。通過啟用跨NUMA模式，VNF可以利用來自兩個NUMA節點的NIC和資源。

2. Active-Backup繫結：

- bond0接口使用SR-IOV NIC建立(eth2來自sriov0, eth3來自sriov1)。
- 主用備份模式可確保冗餘和容錯，而無需交換機端配置。

3. OpenStack網路：

- 協調和管理網路：基於Openvswitch的控制和管理流量。
- 應用程式/服務網路：基於SR-IOV的高效能流量。

為什麼需要跨NUMA模式

1. OpenStack中的NUMA感知網路

NUMA是一種記憶體體系結構，其中每個CPU（及其本地記憶體和裝置）被分組到一個NUMA節點。在OpenStack中，支援NUMA的放置可確保將VNF最佳地分配給同一NUMA節點上的資源，以最小化延遲並最大化效能。

- SR-IOV NIC為NUMA-Local:
 - 每個物理NIC都連線到一個特定的NUMA節點。舉例來說：
 - sriov0連線到NUMA節點0。
 - sriov1連線到NUMA節點1。
- Single-NUMA模式限制：
 - 以單NUMA模式啟動VNF時，OpenStack僅允許VNF連線到啟動VNF的NUMA節點本地的NIC。這意味著：
 - 如果在NUMA 0上啟動VNF，則它只能連線到sriov0上的NIC。
 - 如果VNF在NUMA 1上啟動，則它只能連線到sriov1上的NIC。

2. 為什麼必須採用跨網路運營模式

CPNR VNF需要訪問：

- 協調網路（Openvswitch、NUMA無關）
- 管理網路（Openvswitch、NUMA無關）
- SR-IOV網路1: Connected to sriov0（NUMA節點0）

- SR-IOV網路2:Connected to sriov1 (NUMA節點1)。

在此部署中，CPNR VNF要求從NUMA 0(sriov0)和NUMA 1(sriov1)訪問SR-IOV NIC，以提供冗餘和高可用性。為了實現這一目標：

- 必須以cross-NUMA模式啟動VNF，這允許OpenStack從多個NUMA節點分配CPU、記憶體和NIC。
- 這可確保VNF可以連線到sriov0和sriov1上的NIC，從而能夠在主用 — 備用繫結配置中使用兩個SR-IOV埠。

OVS埠的Contrack大小限制

什麼是Contrack？

Contrack是一種Linux核心功能，用於跟蹤網路連線，特別是網路地址轉換(NAT)和防火牆規則。對於OpenStack中基於OVS的埠，contrack用於管理連線狀態和實施安全組規則。

Contrack對OVS埠的影響

1. Contrack表：

- 每個活動連線都使用contrack表中的條目。
- contrack表的大小受`nf_contrack_max`parameter的限制。

2. 預設限制：

- 預設情況下，contrack表大小為65536個條目。對於高連線速率的工作負載（例如，具有許多同時流的VNF），此限制可能很快耗盡，導致丟包。

3. 對OVS埠的影響：

- 如果contrack表已滿，則會丟棄新連線，從而嚴重影響VNF效能。
- 這對於使用OVS埠的協調和管理網路尤其重要。

如何緩解連線限制

1. 增大跟蹤表大小：

- 檢視當前限制：

```
sysctl net.netfilter.nf_contrack_max
```

- 提高限制：


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

- 使更改具有永續性：

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

2. 監控連線使用情況：

檢查連線統計資訊：

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

3. 最佳化安全組規則：

減少應用於OVS埠的規則數量，以最小化控制開銷。

SR-IOV如何解決連線問題

1.消除連線跟蹤相關性

SR-IOV埠繞過OVS資料路徑和Linux核心功能（如conntrack）。這完全消除了連線跟蹤開銷。

2.更高的可擴充性

與OVS埠不同(受conntrack表大小(nf_conntrack_max)的限制),SR-IOV埠可以處理幾乎無限數量的連線。

3.減少延遲

通過將資料包處理解除安裝到NIC硬體，SR-IOV埠消除了基於軟體的連線處理帶來的延遲。

為什麼為CPNR VM上的SR-IOV埠選擇活動備份模式

Active-Backup繫結模式由於其簡單、容錯以及與SR-IOV介面的相容性，特別適合於此部署。原因如下：

1.無複雜性的冗餘

- Active-Backup模式：在任何給定時間，只有一個介面（活動介面）傳輸和接收流量。其他介

面仍保持待機模式。

- 如果活動介面發生故障（例如，由於鏈路故障或硬體問題），繫結會自動切換到備用介面。這樣可確保連續的網路連線，而無需人工干預。

2. 無需鏈路聚合組(LAG)

- 與其他繫結模式(例如，802.3ad或balance-alb)不同，主動備份模式不需要鏈路聚合控制協定(LACP)或交換機端配置。
- 這一點對於SR-IOV埠尤其重要，因為SR-IOV VF通常不支援LACP或LAG配置。

3. 無縫故障切換

- 故障轉移幾乎是瞬時的，對流量的中斷最小。
- 當活動介面發生故障時，連線立即將備用介面升級為活動狀態。

4. 硬體獨立

Active-Backup模式獨立於底層物理交換機或硬體運行。故障切換邏輯完全駐留在Linux核心中，使其具有高度可移植性和通用性。

5. 針對SR-IOV進行了最佳化

SR-IOV VF繫結到特定物理NIC和NUMA節點。通過使用主用備份模式，您可以將來自不同NUMA節點的VF合併到單個邏輯繫結介面(bond0)。這可確保高可用性，同時有效利用NUMA資源。

Active-Backup模式是Linux繫結中最簡單、使用最廣泛的模式之一。它旨在提供高可用性，確保即使某個繫結介面發生故障，流量也能繼續無縫流動。這是對活動備份模式的工作方式、主要特性和優勢的深入解釋。

什麼是Linux Bond介面？

Linux中的bond介面將兩個或多個網路介面合併到一個邏輯介面中。此邏輯介面(稱為繫結(例如bond0))用於提供：

- 備援：確保網路連線的高可用性。
- 效能改進：在其他模式(例如balance-rror802.3ad)中，它也可以聚合頻寬。

Active-Backup模式如何工作？

在Active-Backup模式下，在任何給定時間都僅使用一個介面(稱為active interface)來傳輸和接收流量。其他介面仍保留備用模式。如果主用介面發生故障，其中一個備用介面將提升為主用狀態，流量將自動重新路由到新的主用介面。

Active-Backup模式的主要功能

1. 單個活動介面：

- 在任何給定時間，繫結中只有一個物理介面處於活動狀態，用於傳送和接收流量。
- 備用介面完全被動，除非發生故障轉移。

2. 自動故障轉移：

- 如果主用介面發生故障（例如，由於硬體問題、電纜斷開或鏈路故障），連線會自動切換到備用介面。
- 故障切換是無縫的，無需手動干預。

3. 故障恢復支援：

故障介面恢復後，根據繫結配置，它可以自動再次變為活動狀態（如果已進行配置）或保持在備用模式。

4. 無交換機端要求：

- 與其他繫結模式（例如，802.3ad or balance-rr）不同，活動備份模式不需要在物理交換機上進行任何特殊配置（例如LAG或LACP）。
- 這使它非常適合於無法進行交換機端配置或繫結SR-IOV虛擬功能（不支援LAG）的情形。

5. 監控：

- 連線使用themimon（媒體獨立介面監視器）引數連續監控所有成員介面的運行狀況。
- 如果檢測到鏈路故障，連線立即切換到正常備用介面。

流量在活動備份模式中的傳輸方式

正常運作

1. Active Interface:

- 流量專門通過活動介面（例如連線ofeth2和連線3中）。
- 備用介面(eth3)保持空閒，不會傳輸或接收流量。

2. 監控：

- 連線定期監視所有成員介面的狀態。這通過以下方式完成：
 - miimon: 以可配置的時間間隔（例如每100毫秒）檢查每個介面的鏈路狀態。
 - 位址解析通訊協定(ARP)監控（可選）：傳送ARP請求以確保活動介面可訪問。

故障轉移方案

1. 活動介面上的鏈路故障：

如果活動介面(eth2)發生故障（例如，電纜拔出、NIC硬體故障或鏈路斷開），鏈路連線會立即使用MIMOON或ARP監控來檢測故障。

2. 自動故障轉移：

- 繫結會切換到備用介面(eth3)，該介面會成為新的活動介面。
- 流量通過新的活動介面重新路由，無需手動干預。

3. 故障轉移及時性：

故障轉移過程幾乎是瞬時的(通常在幾毫秒內，具體取決於最小間隔)。

故障恢復方案

1. 恢復故障介面：

- 恢復以前發生故障的介面(eth2)時，可以：
 - 自動回收活動角色（如果配置為這樣做）。
 - 保持待機模式（預設行為）。

2. 流量連續性：

回切是無縫的，確保不會中斷持續流量。

使用案例:與SR-IOV埠的主用 — 備用繫結

Active-Backup模式尤其適用於SR-IOV介面，因為：

- SR-IOV VF通常不支援鏈路聚合協定，例如LACP。
- 主用備份模式下的繫結可以提供冗餘，而無需任何交換機端配置。

舉例來說：

- eth2對映到SR-IOV VF onsriov0（NUMA節點0）。
- eth3對映到SR-IOV VF onsriov1(NUMA節點1)。
- 繫結(bond0)組合了這些介面，在SR-IOV VF之間提供了無縫故障切換。

步驟1. OpenStack網路

CPNR VNF需要以下四個網路：

1. 協調網路：用於控制和協調流量（基於Openvswitch）。
2. 管理網路：用於管理訪問（基於Openvswitch）。
3. SR-IOV網路1:sriov0上的應用/服務流量。

4. SR-IOV網路2:sriov1上的應用/服務流量。

逐步部署：

步驟1.1.建立Openvswitch網路

- 協調網路：

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

- 管理網路：

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

步驟1.2.為Openvswitch網路建立子網

- 業務流程子網：

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

- 管理子網：

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

步驟1.3.建立SR-IOV網路

- SR-IOV網路1:

```
openstack network create --provider-network-type vlan \  

```

```
--provider-physical-network sriov0 --provider-segment 101 sriov-network-1
```

- SR-IOV網路2:

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

步驟2. OpenStack風格

步驟2.1.建立跨NUMA口味

為了確保VNF可以訪問來自兩個NUMA節點的SR-IOV NIC，請建立具有跨NUMA支援的風格：

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

步驟2.2.配置NUMA屬性

設定NUMA特定的屬性：

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

步驟3.在活動備份模式下配置繫結

啟動VNF後，為VNF上的SR-IOV埠(eth2和eth3)配置繫結介面。

步驟3.1.繫結介面配置

在活動備份模式下建立繫結介面(bond0):

```
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
```

步驟3.2.配置從介面

- 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
```

步驟3.3.應用配置

重新啟動網路服務以應用組態：

```
systemctl restart network
```

驗證

部署VNF後，使用以下步驟驗證其功能：

1.檢驗VNF狀態

檢查VNF例項是否處於活動狀態：

```
openstack server show cpmr-instance
```

確保狀態為ACTIVE。

2.檢驗網路連線

- Ping測試：驗證VNF是否可以通過所有網路通訊：

```
ping
```

```
ping
```

- Bond Interface:
 - 確認bond0處於活動狀態：

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

尋找：

- 當前活動的從屬：指示活動介面。
- 從介面：確認2和3都是鍵的一部分。

3.驗證NUMA位置

確保VNF正在使用來自兩個NUMA節點的資源：

```
nova show
```

```
--human | grep numa
```

最佳實踐

- 監控和故障排除：使用工具（例如keetcpdumpandethtoolkit）監控SR-IOV介面。
- 資安：小心管理對物理網路的訪問，並在租戶之間實施嚴格隔離。
- 擴展：在擴展SR-IOV部署時規劃物理NIC容量，因為可用VF的數量受NIC硬體的限制。

疑難排解

如果部署未按預期工作，請參閱以下故障排除步驟：

1.檢驗SR-IOV配置

- 檢查BIOS中是否啟用了SR-IOV:

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

- 確認已在NIC上建立VF:

```
lspci | grep Ethernet
```

2.驗證NUMA位置

如果VNF無法訪問兩個NIC，請確保啟用跨NUMA模式：

- 檢查風味的NUMA屬性：

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

3.債券介面發行

- 檢查繫結狀態：

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

- 如果連線不起作用：
 - 確保從介面(eth2和eth3)已正確配置為繫結的一部分。
 - 重新啟動網路服務：

```
systemctl restart network
```

4.網路連線問題

- 驗證OpenStack埠繫結：

```
openstack port list --server cplr-instance
```

- 檢查VNF中正確的IP配置：

```
ip addr show
```

結論

在具有SR-IOV埠的OpenStack上部署CPNR VNF需要跨NUMA模式，以使VNF從兩個NUMA節點連線到NIC。這是至關重要的，因為OpenStack將VNF限制在單NUMA模式下，只訪問啟動VNF的NUMA節點內的資源（NIC、CPU、記憶體）。將跨NUMA模式與Active-Backup繫結結合使用可確保高可用性、容錯性和高效的資源利用率，從而使此部署具有極高的彈性和效能。

關於此翻譯

思科已使用電腦和人工技術翻譯本文件，讓全世界的使用者能夠以自己的語言理解支援內容。請注意，即使是最佳機器翻譯，也不如專業譯者翻譯的內容準確。Cisco Systems, Inc. 對這些翻譯的準確度概不負責，並建議一律查看原始英文文件（提供連結）。