NSP
Network Services Platform

NSP-DCN-NCI 容器解决方案

云杉网络

目录

一、摘要

云原生技术正在企业上云的过程中扮演着重要角色，企业采用容器、微服务、DevOps 后能在不修改业务代码的前提下完成企业应用上云，显著提升了业务在基础设施中的敏捷性、弹性和可移植性。Gartner 预测，到 2022 年将有超过 75%的全球组织机构会部署容器化的生产业务应用。中国信息通信研究院的调查报告显示，2019 年 43.9%的被访企业已经使用容器技术部署业务应用，计划使用容器技术部署业务应用的企业占比为 40.8%；28.9%的企业已经使用微服务架构进行应用系统开发，另外有 46.8%的企业计划使用微服务架构。但采用了云原生技术的企业在安全隔离、弹性扩展、跨资源池互联等方面又遇到了新的挑战。

NSP 是北京云杉世纪网络科技有限公司（以下简称：云杉网络）推出的一款数据中心混合云网络互联与服务平台软件，支持 OpenStack、VMware、Bare-metal、容器等多种类型的资源池和主流网络设备，为企业数据中心提供网络虚拟化、网络编排和边界网络服务，专注于解决企业从传统 IT 架构向云架构演进过程中遇到的业务连续性、敏捷性、安全性的难题。

二、企业采用容器之后网络的现状以及挑战

随着企业的不断发展，其 IT 基础设施也在不断迭代和演化。企业的 IT 环境因此往往存在多种资源池和多厂商、多型号的设备。此外，由于技术的更新和人员的流动，系统内会呈现出不同架构、不同语言的应用和服务，企业借助容器等云原生技术大幅提升了应用的交付效率，来达到降本增效的目标。据 IDC 预测，到 2022 年全球 60%的组织机构将增加对云原生应用及平台的资金投入。以容器、微服务、DevOps 为代表的云原生技术的引入，促使 IT 架构从稳态转向敏态。

目前常见的容器网络方案中 Flannel-VXLAN、Calico-IPIP、Weave Net、Contiv-VXLAN、NCP、OpenShift-SDN 都是基于 Overlay 隧道实现，而 Flannel-HostGW、Calico-BGP、Contiv-BGP 都是基于路由方式实现。此外，还有完全依赖 Underlay 实现的网络方案，如 SR-IOV、MACVLAN、IPVLAN 等。

在满足云原生的系统架构中，与任何事物的通信都要通过网络来进行。但容器的引入使得系统在隔离性方面加大了逃逸风险；在数据存储方面也更容易造成泄漏。现有的容器网络方案在安全性、可扩展性、可管理性、以及性能等方面对于大规模业务的支撑不够，难以满足业务持续演进的要求。此外，企业对于业务系统的跨资源池迁移和弹性部署以及对旧有设备的服务化需求也日趋强烈。

2.1 云原生业务的安全隔离

企业引入容器技术通常采用 Kubernetes 管理资源，然而 Kubernetes 在网络层面没有提供不同业务之间 Pod 隔离的逻辑，所有 Pod 默认都是能互通的，Pod 之间的隔离需要额外配置 Policy 来按需实现。而 Kubernetes 中子网网段通常是按照 Node 粒度来划分的，在不同的 Node 上 Pod 所分配的 IP 地址属于不同的子网网段。当这些不同的 Node 上的 Pod 按照不同的需求组合来提供不同业务时，业务隔离所需要创建的 Pod 之间的 Policy 将会非常复杂，并且势必在大规模业务场景下要求数量庞大的 Policy，这样就会导致 Policy 配置速度慢、运维复杂度高、处理性能低等诸多问题。

业界也有通过部署多个 Kubernetes 集群并且每个集群中只运行一个业务来实现业务隔离的方案，不同的集群之间通过二层广播域或三层路由域或防火墙 ACL 等方式进行隔离。虽然隔离边界清晰，但多集群增加了管理上的复杂度，并且在业务规模扩展以及资源利用上也有很大的限制。

2.2 容器网络的弹性扩展

企业 IT 基础设施的建设是一个循序渐进、新旧并存的过程，尤其是当前混合云成为主流趋势。为了保证业务的连续性并能平滑迁移至新资源池中，新上线的 Kubernetes 容器资源池需要能够和现有的裸机或者其他虚拟化资源池业务互通。此外，一些业务本身就需要基于混合资源池来部署提供，比如超算、共享存储等。

Kubernetes 集群自身的容器规模也受限于容器网络方案的可扩展性。对于 Calico-BGP 路由方案，每个 Node 上都会运行 BGP 实例，并且依赖于 Underlay 网络提供 BGP RR 和路由隔离能力，在这种大规模场景下对于 Underlay 网络设备的 RR 和路由规格要求非常高，并且管理的复杂度也会超线性上升。而对于 Overlay 隧道方案，其采用的主机 Overlay 本身就存在性能限制，如果所有 Node 上的 Pod 都需要互通，那么就需要建立 full-mesh 的隧道，N 个 Node 的集群规模将会产生 O(n2)的隧道数要求，不利于规模扩展。

2.3 容器网络的可管理性

云平台中的子网网段通常是根据业务构建需求来划分的，而 Kubernetes 是基于 Node 来划分子网网段的，在不同的 Node 上会根据该 Node 所配置的子网网段来给其上的 Pod 自动分配对应网段中的 IP 地址，无法做到按照业务来划分网段和分配 IP 地址。这种方式下 Pod 所对应的业务无法直接通过 IP 地址看出，并且 Pod 所属的业务层级也无法明确。另外，在某些金融场景中，按照合规性要求，业务是需要与 IP 地址绑定，对此基于 Node 粒度的网段划分将无法满足。

Kubernetes 服务分为用于集群外部访问的南北向服务和用于集群内部访问的东西向服务。南北向服务主要通过 Ingress 方式来实现，其本质是通过反向代理来提供对外的访问代理以及实现负载均衡。一般来说，企业都会通过专业厂商的硬件防火墙或者负载均衡设备来对外提供业务访问，一方面为了利旧，一方面为了性能，南北向服务需要能按需引入专业服务，而 Kubernetes 本身未提供此能力。东西向服务主要通过 ClusterIP 方式来实现，底层是基于 iptables 或者 ipvs 来进行地址转换以及实现负载均衡。一方面 ClusterIP 没有隔离，不相关业务的 Pod 都可以访问，另一方面所有流量都需要走复杂的内核协议栈和 netfilter/ipvs 进行转换和发送处理，效率低的同时出现问题也不方便定位。

三、云数据中心容器网络管控方案

NCI（NSP Container Infrastructure）是云杉网络 NSP-DCN 产品针对 Kubernetes 提供的统一网络编排和服务管理解决方案。该方案实现了对 Kubernetes 的 Pod 网络、东西向和南北向服务网络实现统一纳管，同时支持 Kubernetes 的资源弹性扩容和跨资源池互联，并满足高性能网络的需求。

3.1 设计原理

在 NCI 的设计实现中，一个 VPC 对应 Kubernetes 集群中的一个 Namespace，隶属于同一个 Namespace 的 Pod 之间是可以互通的，而不同 Namespace 所属的 Pod 之间默认是隔离的。相应地，一个 Namespace 中的 Pod 通过组网编排，可以对外提供完整的业务；多个业务则对应由多个 Namespace 提供。

3.2 方案组件

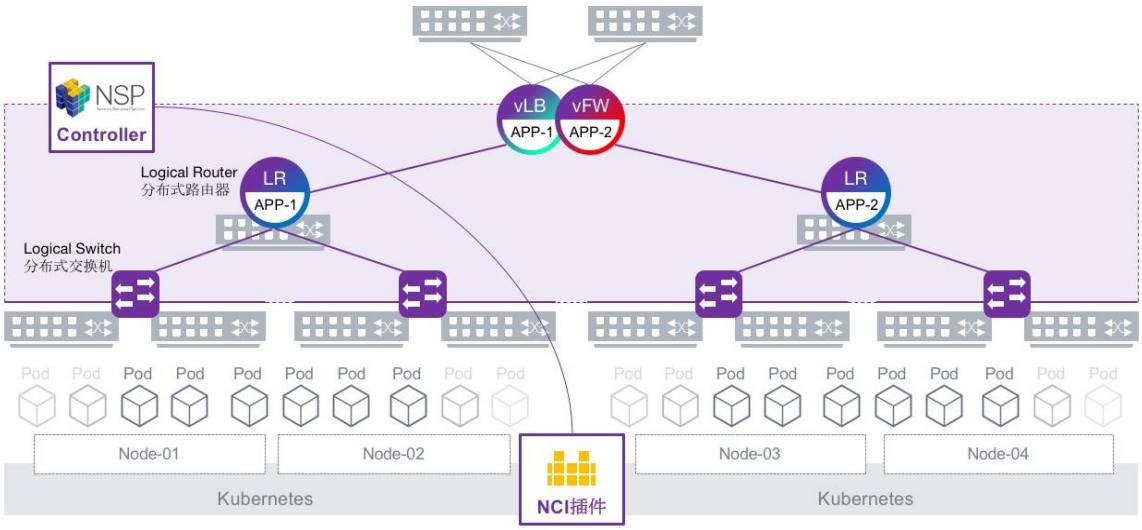
NSP 的主要组件通常部署在标准 x86 集群中，共有包括控制器、逻辑路由器、逻辑交换机、虚拟防火墙等多个网络功能模块组成。本方案涉及的插件主要是 NSP 容器插件和相关控制器模块。

3.2.1 基于 Kubernetes 的插件

NCI 中包含的 Kubernetes 插件是 NSP-DCN 产品中适配容器资源池的功能部分。通过 Kubernetes 插件，NSP 控制器完成了对现网中容器网络的对接、实现了对容器网络的基本管理。

3.2.2 NSP 控制器中的 NCI 模块

NCI 中的 NSP 控制器首先自动同步容器资源池的上联设备，为数据中心网络（包括容器）管理提供了统一的北向接口，对包括容器资源池在内的数据中心网络提供配置管理、策略管理、资源管理、服务管理、组网编排等功能。



整体而言，NCI 的组件包含了运行在 Kubernetes Node 上的 NCI-Controller 和 NCI-OpenvSwitch 模块，运行在 Kubernetes Master 上的 NCI-Supervisor 模块，以及运行在 NSP-DCN 控制器上的 NCI-Provider 模块。各模块主要功能如下：

- NCI-Controller 实现了 CNI，负责响应 Pod 以及服务的增删改，并执行相应的组网和服务配置；
- NCI-OpenvSwitch 负责运行 OvS（OpenvSwitch）网桥，组网和服务配置通过下发给 NCI-OpenvSwitch 的接口和流表配置来实现；

- NCI-Supervisor 负责资源（namespaces、subnets、ips、node-subnets 等）的管理，一方面对 NCI-Controller 申请的资源执行分配，一方面向 NCI-Provider 同步资源信息；
- NCI-Provider 则负责根据 NCI-Supervisor 同步过来的信息去配置 Kubernetes Node 所上联的 Leaf 交换机，完成 Kubernetes 资源池内部的整体组网配置。

3.3. 方案优势

- 与数据中心网络联动

NCI 方案中 Pod 的组网逻辑主要基于硬件 Leaf 交换机上的 LS（Logical Switch）和 LR（Logical Router）来实现，而接入的 OvS 网桥仅负责 Node 内部的 Pod 二层互通。当默认隔离的 Namespace 之间需要互相访问时，通过 NSP-DCN 的组网编排，LS 或 LR 在两个 Namespace 之间建立对等连接，从而通过东西向防火墙实现互通。

面向多资源池场景下的容器网络跨资源池（异构）互联时，NCI 方案采用 Multi-Fabric 架构组网，通过部署在 Region 中的 Service Leaf（虚拟路由）为容器资源池与 Region 内部相同资源池、Region 内部异构资源池、Region 之间异构资源池以及多中心提供统一互联的二层、三层虚拟网络，从而实现容器资源池与整个数据中心网络的联动。

- 简化容器网络的使用

NCI（控制器模块）通过对接和纳管上联设备，将多层级的容器网络抽象为双层设计，解耦并重构了容器资源和网络的复杂访问逻辑。对于外部访问，NSP-DCN 在逻辑网络中创建 VFW（Virtual Firewall）和 VLB（Virtual Load Balancer），并将 VFW 和 VLB 与 Kubernetes 的 Pod 进行组网编排，NCI 将对应的南北向服务配置同 VLB 进行关联，实现对应的南北向服务配置；当 Service、Deployment 发生变化时，对应的后端 Pod IP 地址/端口变化会实时更新到 VLB 中的 Real Server Pool，以保证业务联动的扩展和调整。对于 Ingress 的模型，VLB 直接将流量转成 Ingress 服务所对应的 IP 地址/端口并转发至 Ingress Controller，通过 Ingress Controller 进一步访问到后端的 ClusterIP 以及相应的 Pod 集群。

NCI 基于 Subnet 来为 Pod 分配地址，同一服务可以映射同一网段的 Pod，不同服务则可以映射不同网段的 Pod，以满足业务组网编排时层次化的地址划分需求。基于 Subnet 来分配 IP 地址也大幅简化了策略的配置管理。

- 基于 VPC 的业务隔离

NCI 在 Kubernetes 资源池中引入 VPC 的实现，VPC 提供了一致的资源纳管粒度，使得容器网络能够被 SDN 控制器统一管理和编排。在 NCI 的设计实现中，一个 VPC 对应 Kubernetes 集群中的一个 Namespace；同一 Namespace 中的 Pod 通过组网编排，最终可以对外提供完整的业务。多个业务则对应由多个 Namespace 提供。

NCI 允许在 Namespace 中创建多个子网，通过对应 VPC 中的 VNF 设备（VGW、VFW、VLB...），Namespace（中的 Pod）各自以 VPC 粒度对外提供业务，对应独立的 Pod 资源以及 LS、LR、VNF 等实例。从而延续了企业上云后的使用习惯、降低了企业的使用成本。

四、 方案总结

云杉网络 NCI 容器方案采用双层网络的设计、通过控制器和插件的方式简化容器网络的配置、编排和管理，满足企业容器资源池的按需弹性扩容和一致性的服务访问，兼顾了混合云网络的性能与灵活性。在编排层面向多中心、多资源池统一管控，为企业上云提供了自服务的混合云网络，满足了云数据中心网络安全高效的要求，增强了企业云数据中心的延展性和应用的灵活部署与调度能力。

了解更多信息

专业的售前技术支持及商务合作，协助您选择最合适的解决方案

详询：400-9696-121

网址：www.yunshan.net

北京云杉世纪网络科技有限公司

北京市海淀区成府路 28 号优盛大厦 A 座 1209