

// Jivops

OpenStack Neutron

Guía completa — 20 lecciones

Fundamentos · Servicios de red · Backends y avanzado · Operación

Contenido

Módulo 1: Fundamentos y arquitectura

- Lección 01 — Qué es Neutron
- Lección 02 — Arquitectura de Neutron
- Lección 03 — El plugin ML2
- Lección 04 — Redes, subredes y puertos
- Lección 05 — Tipos de red y encapsulación

Módulo 2: Routing, seguridad y servicios

- Lección 06 — Routers y NAT
- Lección 07 — IP flotantes
- Lección 08 — DHCP y metadatos
- Lección 09 — Grupos de seguridad y port security
- Lección 10 — Calidad de servicio (QoS)

Módulo 3: Backends y funciones avanzadas

- Lección 11 — Open vSwitch y Linux Bridge
- Lección 12 — OVN
- Lección 13 — Enrutamiento distribuido y alta disponibilidad L3
- Lección 14 — Servicios avanzados: LBaaS, VPNaaS y FWaaS
- Lección 15 — Redes provider y SR-IOV

Módulo 4: Operación

- Lección 16 — Despliegue y configuración
- Lección 17 — Resolución de problemas
- Lección 18 — Rendimiento y escalabilidad
- Lección 19 — Seguridad y multitenencia
- Lección 20 — Proyecto final: red de una nube privada con Neutron

Módulo 1: Fundamentos y arquitectura

Lección 01 — Qué es Neutron

El servicio de redes como servicio de OpenStack: ofrece una API para crear y gestionar redes, subredes, puertos, routers, IP flotantes y grupos de seguridad que conectan las instancias de Nova, delegando la implementación en plugins y agentes.

Permite autoservicio y automatización, con redes aisladas por proyecto.

Lección 02 — Arquitectura de Neutron

El servidor neutron-server expone la API REST y guarda el estado en la base de datos; los plugins implementan la lógica, y los agentes en los nodos (L2, L3, DHCP, metadata) configuran la red real, comunicándose con el servidor mediante una cola de mensajes.

Para comunicar el servidor con los agentes de forma asíncrona (por ejemplo con RabbitMQ).

Lección 03 — El plugin ML2

El plugin modular de capa 2 de Neutron: separa los tipos de red (type drivers como flat, VLAN, VXLAN, GRE o Geneve) de los mecanismos que los implementan (mechanism drivers como Open vSwitch, Linux Bridge u OVN), permitiendo combinarlos y usar varios a la vez.

Da flexibilidad: se combinan tipos y mecanismos distintos y es fácil añadir nuevos drivers.

Lección 04 — Redes, subredes y puertos

La red es el dominio de capa 2 aislado; la subred es un rango de direcciones IP (CIDR) con puerta de enlace y DHCP asociado a esa red; y el puerto es el punto de conexión de una instancia u otro dispositivo, con su MAC y sus direcciones IP.

Las instancias no reciben dirección IP ni DHCP en esa red.

Lección 05 — Tipos de red y encapsulación

Flat (sin etiquetas), VLAN (segmentos 802.1Q de la red física) y las redes superpuestas VXLAN, GRE y Geneve, que encapsulan el tráfico sobre la red IP; las superpuestas escalan a muchos más segmentos y no dependen de la configuración VLAN del switch.

Porque la encapsulación añade cabecera (unos 50 bytes en VXLAN) y evita la fragmentación de paquetes.

Módulo 2: Routing, seguridad y servicios

Lección 06 — Routers y NAT

Conecta redes de proyecto entre sí y con una red externa: enruta entre subredes, hace SNAT del tráfico saliente de las instancias y DNAT para las IP flotantes; se implementa con namespaces (agente L3) o con lógica en OVN.

Se crea un router, se le añade la subred como interfaz y se configura el gateway externo.

Lección 07 — IP flotantes

Para dar acceso externo a una instancia asociando una IP pública de la red externa a su puerto interno mediante NAT uno a uno, pudiendo reasignarla a otra instancia sin cambiar la IP interna.

Permite cambiar de instancia (por ejemplo en mantenimiento o fallo) sin cambiar la IP pública ni el DNS.

Lección 08 — DHCP y metadatos

El agente DHCP ejecuta un dnsmasq por red en un namespace (qdhcp) y entrega IP, DNS y rutas; el agente de metadatos permite a la instancia consultar 169.254.169.254 para obtener datos como claves SSH y scripts de cloud-init.

cloud-init no obtiene la configuración inicial, por ejemplo la clave SSH, y la instancia puede quedar inaccesible.

Lección 09 — Grupos de seguridad y port security

Con grupos de seguridad, conjuntos de reglas de entrada y salida con estado aplicados al puerto (por defecto se deniega la entrada y se permite la salida), y port security, que evita suplantación de MAC e IP en el puerto.

Una regla de entrada TCP al puerto 22 desde el origen permitido.

Lección 10 — Calidad de servicio (QoS)

Políticas aplicadas a puertos o redes: límite de ancho de banda, marcado DSCP y ancho de banda mínimo garantizado (según el backend), para evitar que una instancia monopolice la red y priorizar el tráfico.

Para evitar el 'vecino ruidoso' y repartir el ancho de banda de forma equitativa.

Módulo 3: Backends y funciones avanzadas

Lección 11 — Open vSwitch y Linux Bridge

Open vSwitch es un switch virtual programable con túneles, OpenFlow y más funciones (se usa en la mayoría de despliegues), mientras que Linux Bridge es más simple, con menos funciones y a veces más fácil de depurar; ambos conectan los puertos de las instancias.

Para entornos simples donde importa la sencillez y la facilidad de depuración.

Lección 12 — OVN

Un controlador SDN que implementa redes virtuales sobre Open vSwitch con bases de datos propias (Northbound y Southbound), encapsulación Geneve y routers, DHCP, NAT y grupos de seguridad distribuidos sin los agentes L3 y DHCP tradicionales.

Es más distribuido, con menos agentes y mejor escala y rendimiento.

Lección 13 — Enrutamiento distribuido y alta disponibilidad L3

Con routers de alta disponibilidad (VRRP/keepalived entre dos nodos de red) para evitar el punto único de fallo, y con DVR (enrutamiento virtual distribuido), que enruta el tráfico este-oeste y las IP flotantes en los nodos de cómputo; en OVN es nativo.

Evita que todo el tráfico pase por los nodos de red, distribuyendo el enrutamiento este-oeste en cómputo.

Lección 14 — Servicios avanzados: LBaaS, VPNaaS y FWaaS

Balanceo de carga mediante Octavia (con instancias amphora u OVN), VPN como servicio para conexiones IPsec y firewall como servicio a nivel de router o puerto, accesibles por API y ofrecidos a los proyectos.

Los proyectos crean balanceadores por API sin hardware dedicado y con alta disponibilidad.

Lección 15 — Redes provider y SR-IOV

Las redes provider mapean a redes físicas existentes (VLAN o flat) que crea el administrador; SR-IOV asigna a la instancia una función virtual de la tarjeta de red para obtener máximo rendimiento y baja latencia, a costa de menos funciones de red virtual.

Para cargas que exigen alto rendimiento y baja latencia, como NFV.

Módulo 4: Operación

Lección 16 — Despliegue y configuración

Con herramientas de automatización (Kolla-Ansible, OpenStack-Ansible, charms u operadores), varios neutron-server detrás de un balanceador, base de datos y cola replicadas, nodos de red dedicados o distribución con OVN, y configuración definida como código.

El backend (OVS u OVN), los tipos de red, la alta disponibilidad y el MTU.

Lección 17 — Resolución de problemas

Que el puerto esté activo, que haya recibido IP por DHCP, los grupos de seguridad, el estado de los agentes o de OVN, los namespaces (qrouter y qdhcp) y las rutas, y con tcpdump seguir el paquete por cada tramo, comprobando también el MTU.

ip netns exec seguido del namespace (qrouter o qdhcp) y el comando.

Lección 18 — Rendimiento y escalabilidad

Replicando neutron-server y workers de API, usando OVN o DVR para distribuir el tráfico, ajustando bases de datos y colas, reduciendo la sobrecarga de agentes, y con ajuste del MTU, offloads de tarjeta y huge pages para el rendimiento del plano de datos.

La base de datos, la cola de mensajes y la carga de los agentes.

Lección 19 — Seguridad y multitenencia

Cada proyecto tiene sus redes aisladas; el administrador puede compartir redes con políticas RBAC concretas hacia proyectos elegidos, aplicar cuotas de recursos y usar grupos de seguridad y port security para proteger el tráfico.

Para evitar que un proyecto agote IP, routers o puertos por error o abuso.

Lección 20 — Proyecto final: red de una nube privada con Neutron

Backend OVN (o OVS con DVR y HA), red subyacente leaf-spine, redes de proyecto superpuestas con Geneve y MTU ajustado, redes provider para externo, routers con alta disponibilidad, IP flotantes, grupos de seguridad por defecto, QoS, cuotas y RBAC, Octavia para balanceo, monitorización y despliegue como código.

Crear el proyecto con cuotas, su red y subred, un router al exterior y los grupos de seguridad.