Capa de red: Plano de Control

La función de la capa de red es mover paquetes desde el src host hasta el dst host.

Función: Aquí se encuentra la **lógica de la red** que controla no sólo **cómo se reenvía** un datagrama, sino también cómo se **configuran** y **gestionan** los **servicios** y **componentes** de la capa de red.

5.1 Introducción

Existen distintos enfoques para determinar cómo se calculan, mantienen e instalan las tablas de forwarding y de flujo:

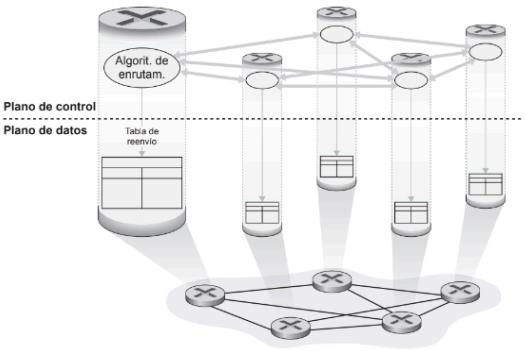
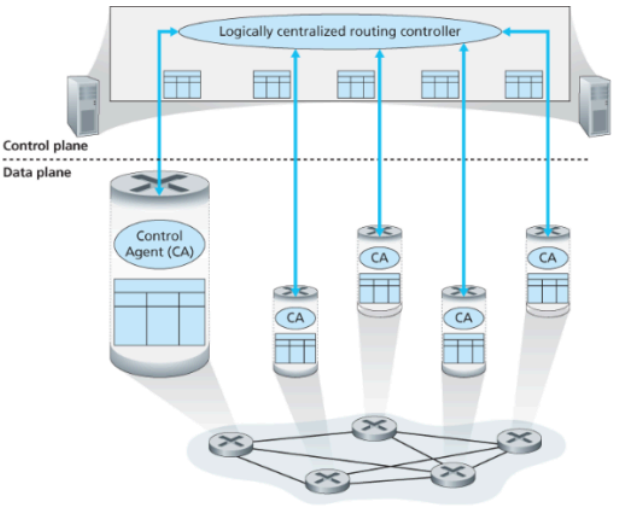


Figura 5.1 ♦ Control por router: los componentes individuales del algoritmo de enrutamiento interactúan en el plano de control.

- **Control por router**: un algoritmo se ejecuta en cada router. Cada router tiene un componente que se comunica con componentes de otros routers para ingresar los valores de la tabla de forwarding. Los protocolos OSPF y BGP están basadosen este enfoque

- **Control lógicamente centralizado:** calcula y distribuye las tablas de reenvío que tienen que usar todos y cada uno de los routers. Al servicio de control de enrutamiento se accede como si fuera un único punto central de servicio, aunque es probable que el servicio se implemente mediante múltiples servidores



El **controlador interactúa con un agente de control (AC)** que **reside en cada router** para configurar y gestionar la tabla de flujo de ese router. Su trabajo consiste en comunicarse con el controlador y en hacer lo que el controlador le ordene. Los AC no interactúan directamente entre sí, ni participan activamente en el cálculo de la tabla de reenvío.

5.2 Algoritmos de enrutamiento

Su objetivo es determinar buenas rutas (por lo general con coste mínimo) desde los emisores a los receptores.

Para plantear los problemas de routing, se puede pensar en grafos donde cada nodo es un router y cada arista es un enlace físico. Además cada arista tiene un peso que representa el costo. Típicamente, el costo refleja la distancia física, la velocidad o el costo monetario del enlace.

Clasificación

centralizados o descentralizados

- **Algoritmo de routing centralizado**: Conocen la red completa antes de realizar el cálculo. El cálculo en sí puede hacerse en un solo sitio (ej: un controlador lógicamente centralizado), o replicarse en el componente de enrutamiento de todos y cada uno de los routers. Los

algoritmos con información de estado global a menudo se denominan algoritmos de estado de enlaces (LS, **Link-State**),

- **Algoritmo de routing descentralizado**: Cada nodo sólo conoce los costes de sus propios **enlaces directamente conectados**. A través de un proceso iterativo de cálculo e intercambio de información con sus nodos vecinos, cada nodo calcula gradualmente la ruta de coste mínimo hacia un destino o conjunto de destinos. Este se denomina **algoritmo de vector de distancias** (DV, Distance-Vector)

Estáticos o dinámicos

- **Estático**: las rutas cambian muy lentamente con el tiempo como resultado de la intervención humana.
- **Dinámico:** se modifican los caminos de enrutamiento a medida que la carga de tráfico o la topología de la red cambian. Son más susceptibles a problemas como los bucles de enrutamiento y la oscilación de rutas.

sensibles o no a la carga

- **sensible**: los costes de enlace varían de forma dinámica para reflejar el nivel actual de congestión en el enlace subyacente. el algoritmo de enrutamiento tenderá a elegir rutas que eviten tal enlace congestionado.
- **no sensible**: el coste de un enlace no refleja explícitamente su nivel de congestión. (OSPF y BGP son algoritmos no sensibles a la carga)

5.2.1 Algoritmo de enrutamiento de estado de enlaces (LS)

En un algoritmo de estado de enlaces, la topología de la red y el coste de todos los enlaces son onocidos. Esto se consigue haciendo que cada nodo **difunda paquetes del estado de los enlaces** a todos los demás nodos de la red.

En casos donde surjan patologías en la red como por ejemplo costos de enlaces igualados a la carga que acarrean, entonces los costos de los enlaces no serían simétricos. Esto puede llevar a oscilaciones debido a que la ruta en una dirección tiene un costo que difiere respecto de la ruta en sentido contrario.

¿Qué podemos hacer para evitar tales oscilaciones (que pueden producirse en cualquier algoritmo que utiliza una métrica de enlace basada en la congestión o el retardo)? Una solución consiste en garantizar que no todos los routers ejecutan el algoritmo LS al mismo tiempo. Esta parece una solución más razonable, ya que podríamos esperar que, aunque los routers ejecutan el algoritmo LS con la misma periodicidad, la instancia de ejecución del algoritmo no sería la misma en cada uno de los nodos.

5.2.2 Algoritmo de enrutamiento por vector de distancias (DV)

(DV) es iterativo, asíncrono y distribuido.

- **distribuido**: cada nodo recibe información de uno o más de sus vecinos directamente conectados, realiza un cálculo y luego distribuye los resultados de su cálculo de vuelta a sus vecinos.

- **iterativo:** porque este proceso continúa hasta que ya no se intercambia más información entre los vecinos.
- **asíncrono:** no requiere que todos los nodos operen sincronizados entre sí.

La única información que tendrá un nodo es el coste de los enlaces a los vecinos a los que está directamente conectado y la información que recibe de esos vecinos. Cada nodo espera a recibir una actualización de cualquier vecino, cuando la recibe calcula su nuevo vector de distancias y lo distribuye a sus vecinos. BGP utiliza DV
El algoritmo permanece en estado de reposo hasta que el coste de un enlace cambia.

Comparación de los algoritmos de enrutamiento LS y DV

Con el algoritmo de vector de distancias, cada nodo sólo se comunica con sus vecinos directamente conectados, pero les proporciona sus estimaciones de coste mínimo desde sí mismo a todos los demás nodos (conocidos) de la red. El algoritmo de estado de enlaces requiere información global.

Robustez.

Si un router falla con el algoritmo de LS, un router podría difundir un coste incorrecto para uno de sus enlaces conectados. Un nodo también podría corromper o eliminar cualquier paquete recibido como parte de un mensaje de difusión LS.

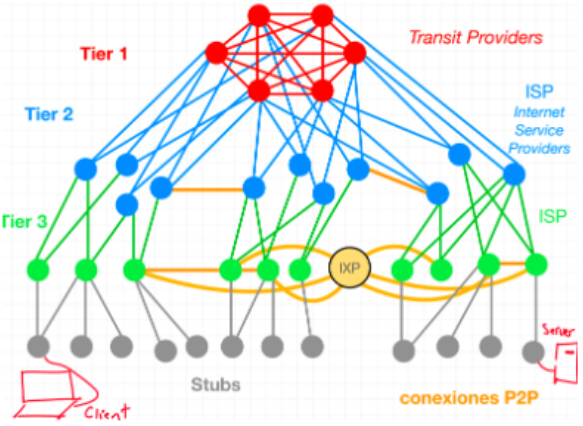
Con el algoritmo de vector de distancias, un nodo puede anunciar rutas de coste mínimo incorrectas a uno o a todos los destinos. En cada iteración, los cálculos de un nodo con el algoritmo de vector de distancias se pasan a sus vecinos. un cálculo de nodo incorrecto puede difundirse a través de toda la red.

5.3 Enrutamiento dentro de un sistema autónomo en Internet: **OSPF** (open shortest path first)

Hasta ahora se vio la red de routers como algo “homogeneo” donde todos los routers tienen que saber la información de todos los demás routers. Esto tiene dos problemas:

- Escalabilidad. Se requeriria muchisimo overhead para tener la info de todo internet!
- Autonomía. Internet es una red de ISPs, donde cada ISP tiene su propia red de routers. Cada ISP va a querer operar cómo le plazca.

Estos dos **problemas** pueden **resolverse** organizando los routers en sistemas autónomos (**AS**, Autonomous System), estando cada AS formado por un grupo de routers que se encuentran bajo el mismo control administrativo.



Los routers que están dentro de un mismo AS, corren el mismo algoritmo de routing y tienen información sobre los demás routers.
El algoritmo de routing que se ejecuta dentro de un sistema autónomo se llama **protocolo de enrutamiento interno del sistema autónomo**

OSPF - Open Shortest Path First

OSPF es un **protocolo de estado de enlaces** que utiliza la técnica de **inundación de información** de estado de los enlaces y un algoritmo de la ruta de coste mínimo de Dijkstra.
Con OSPF, un router **construye un mapa topológico completo** (es decir, un grafo) del sistema autónomo entero. Un router difunde la información de estado de los enlaces cuando se produce un cambio en el estado de un enlace y periódicamente el estado de un enlace.

Se utiliza ampliamente para el **enrutamiento interno** de los **sistemas autónomos** en Internet

El administrador puede decidir hacer igual a 1 el coste de todos los enlaces, proporcionando así un enrutamiento con un número mínimo de saltos, o puede definir los pesos de los enlaces para que sean inversamente proporcionales a la capacidad de los mismos, con el fin de disuadir al tráfico.

- ➢ Proporciona el mecanismo (el protocolo) para determinar el enrutamiento de coste mínimo para el conjunto de pesos de los enlaces.
- ➢ El protocolo OSPF también comprueba que los enlaces estén operativos y permite al router OSPF obtener de un vecino la base de datos de estado de los **enlaces de toda la red**.
- ➢ Esta actualización periódica de los anuncios del estado de los enlaces añade robustez al algoritmo LS

Funcionalidades avanzadas incluidas:

- **Seguridad**: Los intercambios entre routers OSPF pueden ser autenticados. Con la autenticación, solo pueden participar en el protocolo OSPF los routers de confianza del sistema autónomo. Hay 2 tipos de autenticación. Con la autenticación **simple**, se configura la misma contraseña en todos los routers y en texto plano. La autenticación **MD5** está basada en claves secretas compartidas que están configuradas en todos los routers.

- **Varias rutas de igual coste**: Permite utilizar varias rutas

- **Soporte integrado para enrutamiento por unicast y por multicast.** O sea que manda el paquete a un grupo o a uno solo

- **Soporte para definir una jerarquía dentro de un mismo AS:** una y solo una de las áreas OSPF del AS se configura como área troncal. El papel principal del **área troncal es enrutar el tráfico entre las restantes áreas del AS**.

5.4 Enrutamiento entre los ISP: BGP

Para enrutar un paquete entre múltiples sistemas autónomos necesitamos un **protocolo de enrutamiento entre sistemas autónomos**. En Internet, todos los AS utilizan el mismo protocolo, BGP: Es el que une a los miles de ISP existentes en Internet.

BGP es un protocolo descentralizado y asíncrono.

5.4.1 El papel de BGP

En BGP, los paquetes se enrutan hacia prefijos CIDR (Máscaras de subred de longitud variable), representando cada prefijo a una subred o a una colección de subredes. (En el mundo de BGP, un destino puede tener la forma 138.16.68/22, que en este ejemplo incluiría 1.024 direcciones IP)

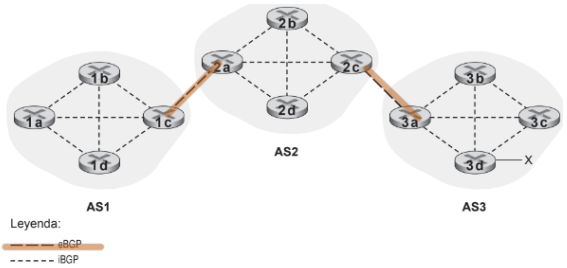
BGP proporciona a cada router **mecanismos** para:

1. Obtener de los sistemas autónomos vecinos información acerca de la alcanzabilidad de los prefijos: **BGP permite a cada subred anunciar su existencia al resto de Internet.**
2. Determinar las **"mejores" rutas** hacia los distintos prefijos.

5.4.2 Anuncio de la información de rutas BGP

En un sistema autónomo determinado, cada router puede ser un router de pasarela o un router interno

- **router de pasarela (gateway router):** está situado en la frontera de un sistema autónomo y se conecta directamente a uno o más routers de otros sistemas autónomos.
- **router interno:** sólo está conectado a hosts y routers pertenecientes a su propio sistema autónomo.



Quienes **intercambian mensajes son los routers**, no los SA
En BGP, las parejas de routers intercambian información de enrutamiento a través de conexiones **TCP semipermanentes**. Cada una de esas **conexiones TCP**, junto con todos los **mensajes BGP** enviados a través de la conexión, **se denomina conexión BGP**.

Una conexión BGP que **abarca dos sistemas autónomos** se llama **conexión BGP externa** (eBGP). Mientras que una sesión BGP entre **routers de un mismo sistema autónomo** se denomina **conexión BGP interna** (iBGP).

Para propagar la información de alcanzabilidad se utilizan sesiones tanto iBGP como eBGP.

5.4.3 Determinación de las mejores rutas

Cuando un router anuncia un prefijo a través de una conexión BGP, incluye con el prefijo varios atributos BGP. En la jerga de BGP, **un prefijo junto con sus atributos** se denomina **ruta**

Hay docenas de diferentes rutas posibles. Para esto hay algunos atributos que nos ayudan:

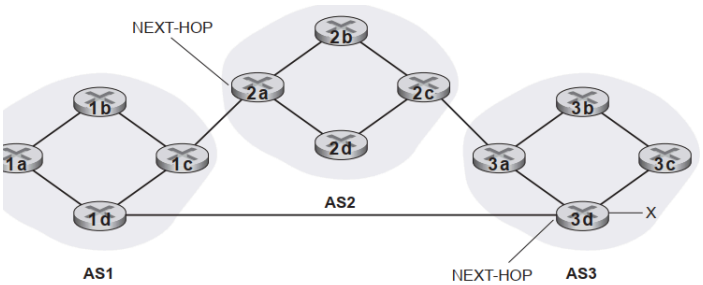
- **AS-PATH**
  - Contiene la lista de sistemas autónomos a través de los que ha pasado el anuncio. Para generar el valor AS-PATH, cuando se pasa una ruta a un sistema autónomo, el sistema añade su ASNumber a la lista contenida en AS-PATH.
  - **AS-PATH se utiliza para detectar e impedir los bucles de anuncio: si un router ve que su propio sistema autónomo está en la lista de la ruta**, rechazará el anuncio.

- **NEXT-HOP**

- Es la dirección IP del router más cercano del **AS adyacente**.

Enrutamiento de la patata caliente

El algoritmo más simple de ruteo es el **hot potato routing**. En este algoritmo, el camino elegido es aquel con el menor costo al router NEXT-HOP. La idea es conseguir que los paquetes **salgan de su propio sistema autónomo** lo más **rápidamente** posible sin preocuparse del coste que tengan las partes restantes de la ruta hasta el destino, situadas fuera de su propio sistema autónomo. Se trata de reducir el coste dentro de su propio sistema autónomo.



Algoritmo de selección de ruta

En la práctica, el algoritmo que se usa es el de route-selection. El input de este algoritmo es un set con todas los caminos hacia el prefijo, que ya fueron aprendidos y aceptados por el router. Si hay más de un camino al mismo prefijo, se aplican reglas de eliminación:

1. Un camino tiene asignado un atributo llamado 'local preference' tal que se eligen los caminos con el mayor valor.
2. De los restantes, se seleccionan los caminos con el AS-PATH más corto.
3. Luego se utiliza el algoritmo de hot potato.
4. Si siguen quedando más de un camino, se utilizan identificados BGP

5.4.4 IP-Anycast

BGP se usa para implementar el servicio IP-anycast

1. Durante la etapa de configuración de IP-anycast la empresa CDN asigna la misma dirección IP a cada uno de sus servidores, y utiliza BGP estándar para anunciar esa dirección IP de cada uno de los servidores.
2. Cuando un router BGP recibe múltiples anuncios de ruta para esta dirección IP, trata esos anuncios como si proporcionaran diferentes rutas hacia una misma ubicación física (cuando, de hecho, los anuncios son para diferentes rutas hacia distintas ubicaciones físicas).
3. Al configurar su tabla de enrutamiento, cada router utilizará localmente el algoritmo de selección de rutas de BGP para elegir la "mejor" ruta hacia esa dirección IP

IP-anycast sí que es ampliamente utilizado por el sistema DNS para dirigir las consultas DNS al servidor DNS raíz más cercano. Cuando un host envía un datagrama a una dirección anycast, la infraestructura de red buscará el camino más corto hasta uno, y preferiblemente sólo uno, de los equipos que aceptan datagramas dirigidos a la dirección anycast utilizada.

5.4.5 Política de enrutamiento

Cuando un router selecciona una ruta hacia un destino, la política de enrutamiento del sistema autónomo puede prevalecer sobre todas las restantes consideraciones.

- Todo el tráfico que entra en la red de un ISP de acceso tiene que estar destinado a esa red
- Todo el tráfico que sale de la red de un ISP de acceso tiene que haber sido originado en esa red.
- Regla heurística seguida por los ISP comerciales: cualquier tráfico que fluya a través de la red troncal de un ISP tiene que tener su origen o su destino (o ambos) en una red que sea un **cliente** de dicho ISP; si fuera de otro modo, ese tráfico estaría obteniendo un viaje gratis a través de la red del ISP.

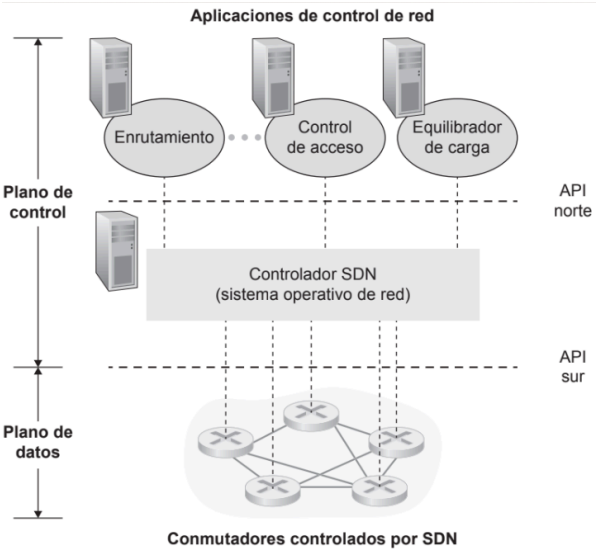
Obtención de presencia en Internet
Firmando un contrato con un ISP local y conectándose a él conseguira conectividad Internet. Su empresa tendrá un router de pasarela conectado a un router de su ISP local. También tendrá que obtener un nombre de dominio para su empresa. Para que la gente pueda descubrir las direcciones IP de su servidor web, tendrá que incluir entradas en su servidor DNS que establezcan la correspondencia entre el nombre de host de su servidor web cuando su empresa firma un contrato con un ISP local y se le asigna un prefijo (es decir, un rango de direcciones), su ISP local usará BGP para anunciar su prefijo a los ISP con los que esté conectado. Esos ISP usarán BGP, a su vez, para propagar el anuncio. Al final, todos los routers de Internet conocerán su prefijo y serán así capaces de reenviar los datagramas destinados a su servidor web

5.5 El plano de control SDN

Es la lógica de la red que controla el reenvío de paquetes entre los dispositivos compatibles con SDN de una red, así como la configuración y gestión de estos dispositivos y sus servicios.

Cuatro características fundamentales de una arquitectura SDN

- **Reenvío basado en el flujo**: es responsabilidad del plano de control SDN calcular, gestionar e instalar las entradas de las tablas de flujo en todos los conmutadores de la red.
- **Separación del plano de datos del plano de control**: El plano de datos consiste en los switches. El plano de control consiste en servidores y software que determinan las tablas de flujo para los switches.
- **Funciones de control de red**: El plano de control consta de 2 componentes: un controlador SDN y un conjunto de aplicaciones de control de red.



- **Red programable**: La red es programable a través de las aplicaciones de control de red que corren en el control-plane. Son el 'cerebro' de la SDN

SDN representa un significativo "**desempaquetamiento**" de la funcionalidad de red: los conmutadores del plano de datos, los controladores SDN y las aplicaciones de control de red son **entidades separadas**, que pueden ser suministradas por **diferentes proveedores y organizaciones**.

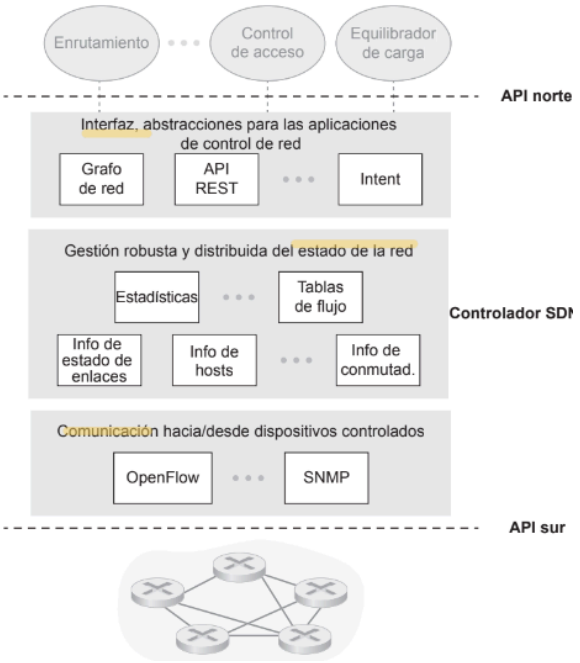
El modelo pre-SDN, en el que un router/switch (+ software integrado del plano de control y sus implementaciones de protocolos) era **monolítico**, estaba integrado verticalmente y era suministrado por un único fabricante.

5.5.1 El plano de control SDN: controlador SDN y aplicaciones SDN de control de red

El plano de control SDN está dividido, a grandes rasgos, en dos componentes: el controlador SDN y las aplicaciones SDN de control de red.

La funcionalidad de un controlador puede organizarse en tres capas:

- **Capa de comunicaciónes**: Comunicación entre el controlador SDN y los dispositivos de red controlados. Se necesita un protocolo para transferir información del controlador a un dispositivo. Interfaz "sur" del controlador
- **Capa de gestión de estado en toda la red**: Todas las decisiones que toma la SDN requieren que el controlador tenga toda la información sobre el estado de la red actualizada (información sobre hosts, enlaces, switches y sobre otros dispositivos controlados). Además, debe tener una copia de las tablas de flujo de los switches.
- **Interfaz para la capa de aplicación de control de red**: El controlador interactúa con las aplicaciones de control de red a través de la interfaz norte. Esta API permite a las aplicaciones de control de red leer/escribir el estado de la red y las tablas de flujo dentro de la capa de gestión de estado



5.5.2 Protocolo OpenFlow

Mensajes importantes que fluyen **desde el controlador hacia el conmutador controlado**:

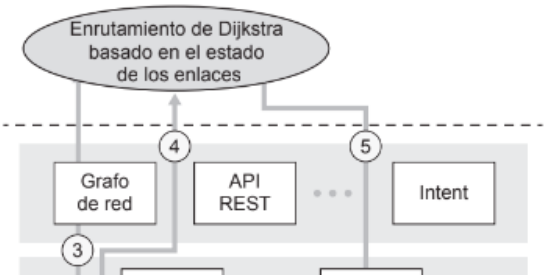
- Configuración
- Modificar estado
- Leer estado
- Enviar paquete

Mensajes que fluyen **desde el conmutador controlado por SDN hacia el controlador**:

- Flujo eliminado
- Estado de puerto
- Paquete entrante

5.5.3 Interacción entre los planos de datos y de control: ejemplo

1. El conmutador s1, al experimentar un fallo en el enlace entre él mismo y s2, notifica el cambio en el estado del enlace al controlador SDN, utilizando el mensaje port-status de OpenFlow.



2. El controlador SDN recibe el mensaje OpenFlow que indica el cambio en el estado del enlace y envía una notificación al gestor del estado de los enlaces, que actualiza una base de datos de estado de los enlaces.
3. La aplicación de control de red que implementa el enrutamiento de Dijkstra basado en el estado de los enlaces, se había registrado previamente para ser notificada cuando hubiera cambios en el estado de los enlaces. Esa aplicación recibe la notificación del cambio en el estado del enlace.
4. La aplicación de enrutamiento basada en el estado de los enlaces interactúa con el gestor de estado de los enlaces para obtener información actualizada del estado de los enlaces. Después calcula las nuevas rutas de coste mínimo.
5. La aplicación de enrutamiento basada en el estado de los enlaces interactúa entonces con el gestor de tablas de flujo, que determina las tablas de flujo que hay que actualizar.
6. El gestor de tablas de flujo utiliza entonces el protocolo OpenFlow para actualizar las entradas de las tablas de flujo de los conmutadores afectados: s1 (que ahora enrutará a través de s4 los paquetes destinados a s2), s2 (que ahora comenzará a recibir paquetes de s1 a través del conmutador intermedio s4) y s4 (que ahora debe reenviar los paquetes de s1 destinados a s2).

5.6 Protocolo de mensajes de control de Internet (ICMP)

Los hosts y los routers utilizan ICMP para intercambiar información acerca de la capa de red. El uso más típico de ICMP es la generación de informes de error.

Los mensajes ICMP **viajan dentro de los datagramas IP**, como payload de IP, tienen un campo type y otro code y contienen el header y los primeros 8 bytes del datagrama IP que causó que se genere el mensaje. ICMP está por encima de IP

El programa ping envía un mensaje ICMP de tipo 8 y código 0 al host especificado.

Su propósito original era llevar a cabo el control de congestión (permitir a un router congestionado enviar un mensaje ICMP de este tipo a un host para forzarle a reducir su velocidad de transmisión).

Traceroute, el cual nos permite trazar una ruta desde un host a cualquier otro host del mundo, se implementa con mensajes ICMP.

Traceroute del origen envía una serie de datagramas IP ordinarios al destino. Cada uno de estos datagramas transporta un segmento UDP con un número de puerto UDP poco probable. El primero de estos datagramas tiene un TTL de 1, el segundo de 2, el tercero de 3, y así sucesivamente. Cuando el datagrama n-ésimo llega al router n-ésimo, este observa que el TTL del datagrama acabade caducar. El router descarta el datagrama y envía al origen un mensaje de advertencia ICMP. Cuando este mensaje ICMP llega de vuelta al origen, este obtiene el tiempo de ida y vuelta del temporizador, y obtiene también del propio mensaje ICMP el nombre y la dirección IP del router n-ésimo.
Por tanto, uno de los datagramas terminará recorriendo el camino completo hasta el host de destino.
Cuando el host de origen recibe este mensaje ICMP, sabe que no tiene que enviar más paquetes de sondeo. De esta forma, el host de origen obtiene el número y la identidad de los routers que existen entre él y el host de destino, así como el tiempo de ida y vuelta entre los dos hosts.

5.7 Gestión de red

La gestión de red incluye la implantación, integración y coordinación del hardware, el software y los elementos humanos para monitorizar, probar, sondear, configurar, analizar, evaluar y controlar los elementos y recursos de la red, con el fin de satisfacer los requisitos de tiempo real, de rendimiento operativo y de Calidad de Servicio, a un coste razonable.

Componentes clave:

- **managing server**: es la aplicación que controla la recopilación, procesamiento, análisis y/o display de la información.El servidor de gestión se ejecuta en una estación central de gestión de red situada en el centro de operaciones de red (NOC) que es el punto focal de la actividad de administración de la red
- **managed device**: es una pieza del equipamiento de red que reside en la managed network. Puede ser un host, router, middlebox, modem, etc. Toda la información recolectada se almacena en Management Information Base
- **Management Information Base** (MIB): Cada objeto gestionado de un dispositivo gestionado tiene asociados distintos elementos de información, que se recopilan en una base de información de gestión MIB
- Un **network management agent** es un proceso corriendo en el managed device que se comunica con el managing server. Lleva a cabo acciones locales en el dispositivo gestionado bajo control y por orden del servidor de gestión. El agente de gestión de red es similar al agente de enrutamiento
- **Protocolo de gestión de red**. El protocolo se ejecuta entre el servidor de gestión y los dispositivos gestionados, permitiendo al servidor consultar el estado de los dispositivos gestionados y llevar a cabo acciones de manera indirecta en estos dispositivos a través de sus agentes.

5.7.2 El protocolo SNMP

El protocolo simple de gestión de red **es un protocolo de la capa de aplicación** que se utiliza para transportar mensajes de información y control para la gestión de red entre un servidor de gestión y un agente que actúa por cuenta del servidor de gestión.
Normalmente, una solicitud se utilizara para consultar (recuperar) o modificar (establecer) los valores de objetos MIB asociados con un dispositivo gestionado. Un segundo uso común de SNMP es cuando un agente envía un mensaje no solicitado, conocido como mensaje TRAP, a un servidor de gestión. Los mensajes TRAP se utilizan para notificar a un servidor de gestión una situación excepcional (por ejemplo, la activación o desactivación de un enlace) que ha dado lugar a cambios en los valores de los objetos MIB.