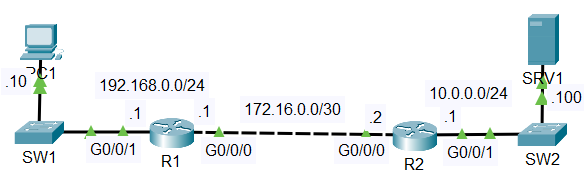
**Calidad del servicio (QoS)**

**Topología**

**In **

**Introducción**

* En este ejercicio, realizaremos algunas configuraciones básicas de QoS en R1. En esta red tenemos la PC1 conectada a R1 a través del switch SW1. Por otro lado, está el servidor SRV1 conectado a R2 vía SW2.
* Aunque solo hay dos PCs conectadas en R1, vamos a simular que tenemos muchas PCs y que la red se está congestionando.

**Configuración de las reglas de QoS**

* Configuraremos QoS en **R1** para garantizar que el tráfico más importante reciba un tratamiento de alta prioridad.
* Marcaremos el tráfico:
  + **HTTPS** como **AF31** y nos aseguraremos de que tenga un **ancho de banda mínimo** del **10%** en una **cola de prioridad**.
  + **HTTP** como **AF32** y nos aseguraremos de que obtenga un **ancho de banda mínimo** del **10%**, pero no estará en una cola de prioridad.
  + **ICMP** como **CS2** y nos aseguraremos de que obtenga un **ancho de banda mínimo** del **5%**.
* El diseño real de QoS de una red depende completamente de la red. Qué tráfico debe priorizarse, cuánto ancho de banda necesita, etc., varía mucho.
* Los valores seleccionados en esta práctica son valores aleatorios que se usan solamente para mostrar la configuración básica de los comandos utilizados en QoS. Dar prioridad al tráfico HTTPS, por ejemplo, probablemente no sea tan común. Por lo general, las colas de prioridad se utilizan para el tráfico de voz.

**Aplicar las reglas en la interfaz de salida de R1**

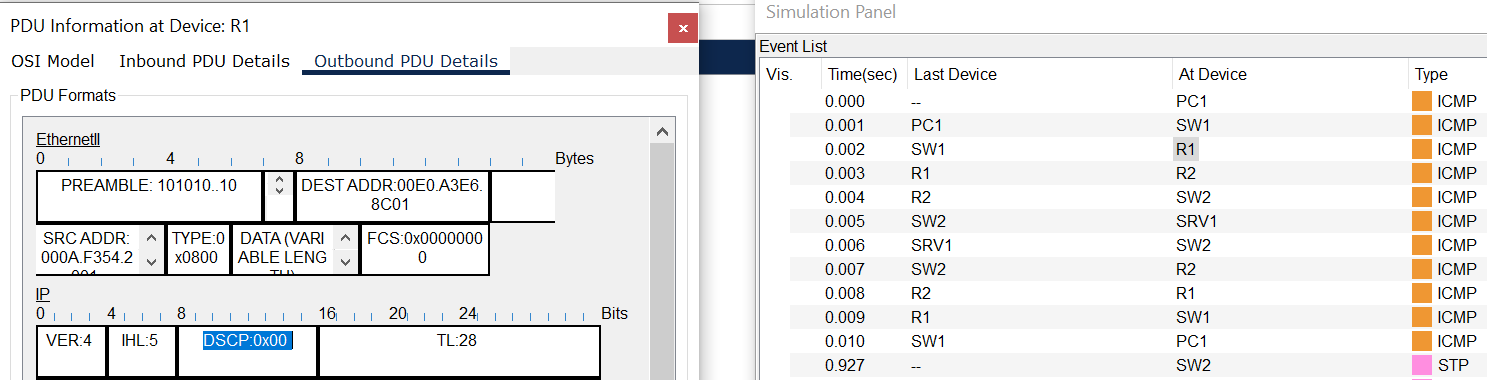
* Después de configurar las reglas de QoS, las aplicaremos al tráfico que se reenvía desde la interfaz G0/0/0 de **R1**.
* Cuando **R1** reenvía el tráfico de G0/0/0 a **R2**, se priorizará de acuerdo con nuestras configuraciones.
* Una vez que el tráfico llega a **R2**, es importante que también esté configurado para marcar los paquetes de alta prioridad, de lo contrario tratará a todos los paquetes por igual. No se puede simplemente configurar un ruteador para priorizar el tráfico y luego esperar a que el resto de la red también lo priorice. Se debe configurar QoS en toda la red, donde sea necesario.

En este ejercicio, vamos a configurar **R1**, para mostrar las configuraciones básicas de QoS.

1. Hay tres pasos principales para configurar **QoS**.
2. Identificar los tipos de tráfico a los que se desea aplicar un tratamiento especial.
3. Especificar qué tipo de tratamiento queremos dar a cada tipo de tráfico.
4. Aplicar el mapa de políticas usando una política de servicio.

## Parte 1: Mostrar marcas predeterminadas en R1

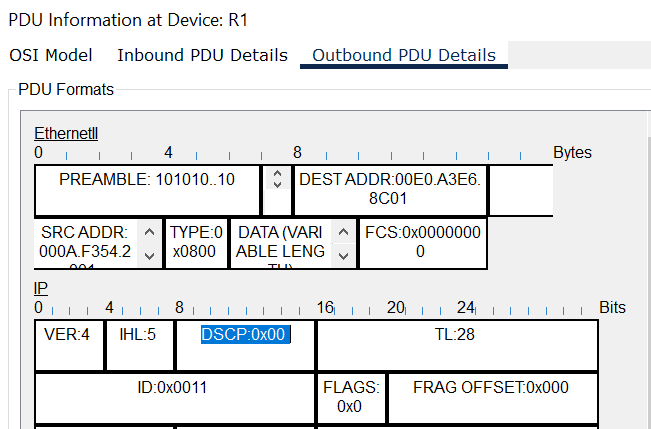
* Antes de la configuración, se mostrarán las marcas predeterminadas del tráfico que se reenvía desde **R1**.
* Cambiar a modo simulación y hacer un ping desde la **PC1** al servidor **SERVER1** .
* Ir a la ventana de **Outbound PDU Details,** revisar el valor de **DSCP (Differentiated Services Code Point)** que hace referencia al segundo byte del encabezado de los paquetes IP y se utiliza para diferenciar la calidad en la comunicación que tienen los datos que se transportan.
* El [marcado **DSCP**](https://en.wikipedia.org/wiki/Differentiated_services) de calidad del servicio (QoS) determina la **clasificación del tráfico** para los datos de red.



* Observe que el campo **DSCP** está escrito en hexadecimal 00. Por cuestiones de tiempo, no se mostrará el tráfico HTTP y HTTPS, que también tendría un valor **DSCP** de **0**.

<https://www.youtube.com/watch?v=DJSgZ2-BpeM>

<https://itgala.xyz/how-to-set-the-dscp-flag-in-windows-and-linux/>



## Parte 2: Configure los mapas de clase para cada tipo de tráfico

Identificar los tipos de tráfico a los que se desea aplicar un tratamiento especial.

**1. Configura el mapa de clase para el tráfico HTTPS**

* Define el mapa de clase **HTTPS\_MAP**, que se usará para hacer coincidir el tráfico **HTTPS**.

**class-map HTTPS\_MAP**

* Este mapa de clase se usará para identificar el tráfico **HTTPS**.

**match protocol https**

**2. Configura el mapa de clase para el tráfico HTTP**

* Define el mapa de clase **HTTP\_MAP**, que se usará para hacer coincidir el tráfico **HTTP**.

**class-map HTTP\_MAP**

* Este mapa de clase se usará para identificar el tráfico **HTTP**.

**match protocol http**

**3. Configura el mapa de clase para el tráfico ICMP**

* Define el mapa de clase **ICMP\_MAP**, que se usará para hacer coincidir el tráfico **ICMP**.

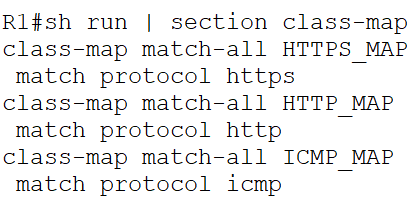
**class-map ICMP\_MAP**

* Este mapa de clase se usará para identificar el tráfico ICMP.

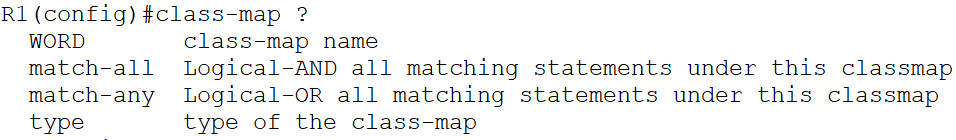
**match protocol icmp**

## Parte 3: Verifica la configuración de los mapas de clase

* Ejecuta el comando **sh run | section class-map.** Se encuentran los tres mapas de clase que se configuraron. En todos los mapas de clase, se aplicó el modo default **match-all.** Esto significa que, para que coincida con este mapa de clase, el tráfico debe coincidir con todas las declaraciones de coincidencia en ese mapa de clase. En este caso, cada mapa de clase solo tiene una declaración MATCH, por lo que no importa, pero si hay varias declaraciones de coincidencia, deben cumplirse todas.



* + - * Ejecuta el comando **class-map ?** en modo de configuración global.



## Parte 4: Configure el mapa de políticas

**1. Configura el mapa de políticas**, para definir qué tipo de tratamiento queremos dar a cada tipo de tráfico.

* + - * El comando es **policy-map** y lo nombraremos **g0/0/0\_OUT**, ya que después aplicaremos este mapa de políticas en la interfaz **g0/0/0** de salida.

**policy-map G0/0/0\_OUT**

**2. Configura el mapa de políticas para el tráfico HTTPS**

* + - * Luego use **class**, seguido del nombre de uno de los mapas de clase que acabo de configurar, HTTPS\_MAP. Use **class**, seguido del nombre del mapa de clase: **HTTPS\_MAP**.

**class HTTPS\_MAP**

* Ahora podemos decirle a **R1** qué hacer con el tráfico que coincide con este mapa de clases. Marca cualquier paquete HTTPS con un valor **DSCP** de **AF31**

**set ip dscp AF31**

* Luego, le daremos a **HTTPS** una cola de prioridad con al menos el **10% del ancho de banda** de los enlaces durante los períodos de congestión.

**priority percent 10**

**3. Configura el mapa de políticas para el tráfico HTTP**

* + - * Use **class**, seguido del nombre del mapa de clase: **HTTP\_MAP**.

**class HTTP\_MAP**

* Marca cualquier paquete **HTTP** con un valor **DSCP** de **AF32**

**set ip dscp AF32**

* Ahora le daremos un **ancho de banda mínimo del 10%** en momentos de congestión y no está en una cola de prioridad.

**bandwidth percent 10**

**4. Configura el mapa de políticas para el tráfico ICMP**

* + - * Use **class**, seguido del nombre del mapa de clase: **ICMP\_MAP**.

**class ICMP\_MAP**

* Marca cualquier paquete **ICMP** con un valor **DSCP** de **cs2**

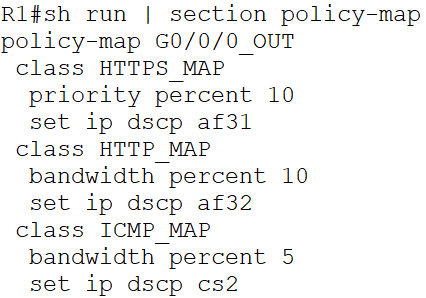
**set ip dscp cs2**

* Ahora le daremos un **ancho de banda mínimo del 5%** en momentos de congestión y no está en una cola de prioridad.

**bandwidth percent 5**

**5. Verifica la configuración de los mapas de políticas**

* Ejecuta el comando **sh run | section class-map.** Todavía no se ha aplicado el mapa de política en la interfaz de salida en **G0/0/0**, pero después de hacerlo, verificará todo el tráfico que se reenviará desde **G0/0/0**.



Si es un paquete:

* **HTTPS**, lo marcará como **DSCP AF31** y le dará una **cola de prioridad** con al menos el **10% del ancho de banda** del enlace.
* **HTTP**, lo marcará como **DSCP AF32** y le dará al menos el **10% del ancho de banda** del enlace.
* **ICMP**, lo marcará como **DSCP CS2** y le dará al menos el **5% del ancho de banda** del enlace.
* El resto del tráfico, que no coincida con ninguno de estas políticas, no se marcará y se reenviará sin ningún tratamiento especial de QoS.

## Parte 5: Aplique este mapa de políticas usando una política de servicio

**1. Aplique el mapa de políticas a la interface g0/0/0 de salida**

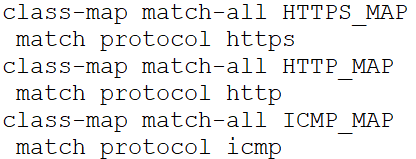
**int g0/0/0**

**service-policy output g0/0/0\_OUT**

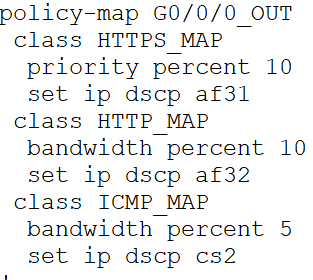
Nuestra configuración QoS está completa.

**2. Verifica la configuración**

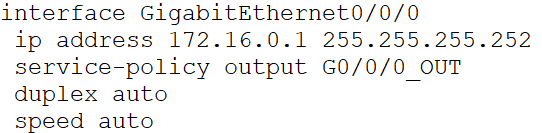
* Ejecuta el comando **sh run.**
* Revisa los **mapas de clase**, identificando los tipos de tráfico que queremos hacer coincidir.



* Revisa el **mapa de políticas**, que establece la acción que queremos realizar sobre el tráfico coincidente en los mapas de clase.



* Finalmente, se utilizó una **política de servicio** para aplicar el mapa de políticas en la interface **G0/0/0** de salida.



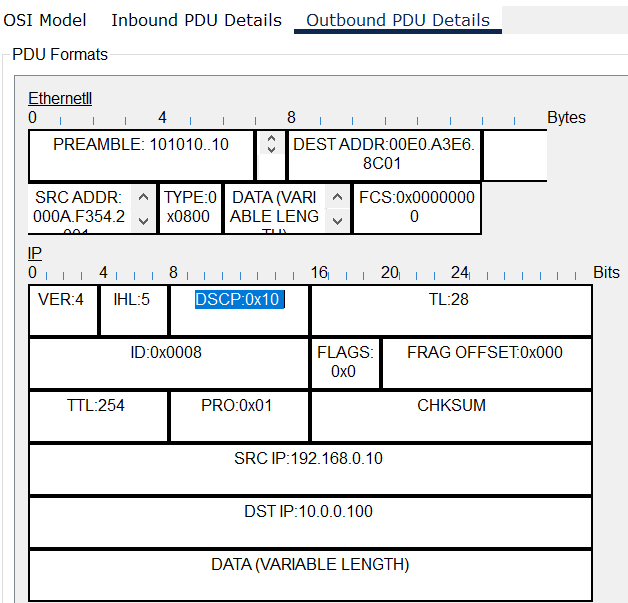
## Parte 6: Envíe tráfico ICMP, HTTP y HTTPS al servidor SERVER1 y verifique las marcas

**1. Envíe tráfico ICMP de la PC1 al servidor SRV1 y verifique las marcas**

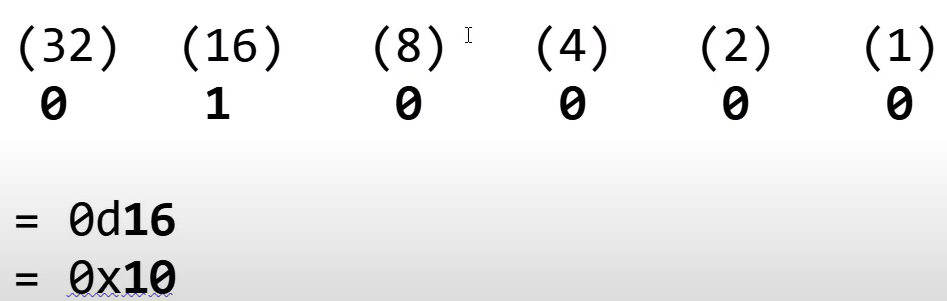
* Abra el modo de simulación y haga un ping al servidor **SERVER1**.

**ping 10.0.0.100**

* En la pestaña de **outbound PDU details**, la marca de **DSCP** es **10 hexadecimal**.

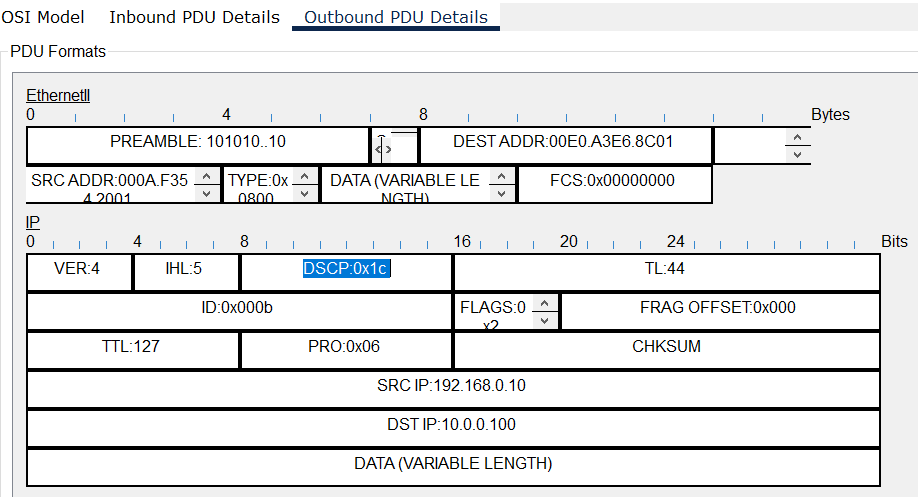


* Los paquetes **ICMP** fueron marcados como **CS2:**
  + Hexadecimal: 10
  + Binario: 01 0000
  + Decimal: 16

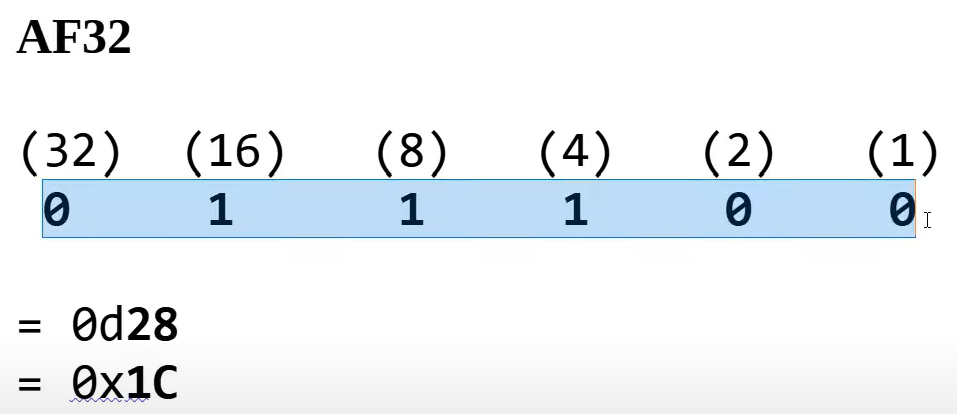


**2. Envíe tráfico HTTP de la PC1 al servidor SRV1 y verifique las marcas**

* En la pestaña de **Outbound PDU Details**, la marca de **DSCP** es **1 C**.

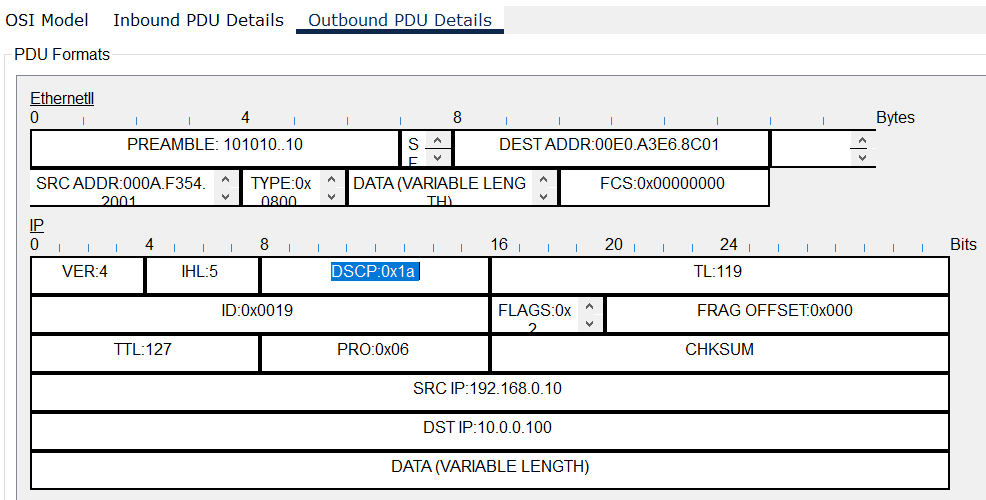


* Los paquetes **HTTP** fueron marcados como **AF32:**
  + Hexadecimal: 1C
  + Binario: 0001 1100
  + Decimal: 28

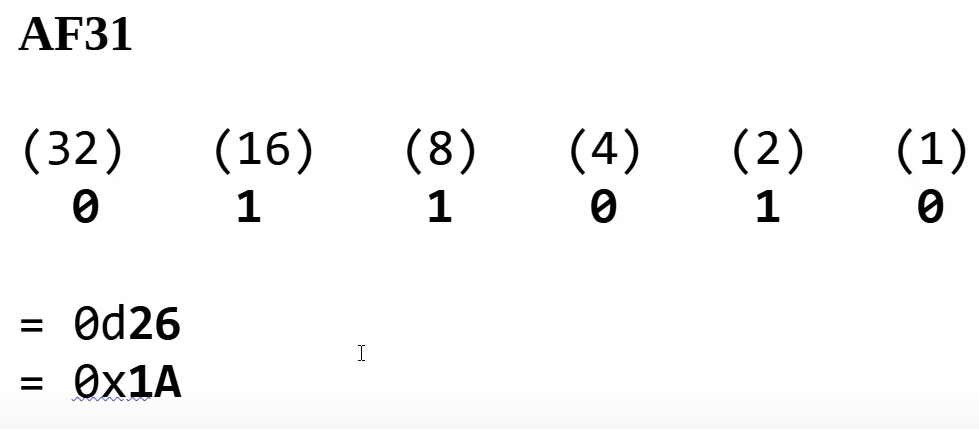


**3. Envíe tráfico HTTPS de la PC1 al servidor SRV1 y verifique las marcas.**

* En la pestaña de **outbound PDU details**, la marca de **DSCP** es **1A**.



* Los paquetes **HTTPS** fueron marcados como **AF31:**
  + Hexadecimal: 1A
  + Binario: 0001 1010
  + Decimal: 26



**Resumen:**

* Los **mapas de clases** identifican el tráfico (**class maps**)
* Los **mapas de políticas** especifican qué acciones realizar con ese tráfico (**policy maps**).
* Las **políticas de servicio** aplican los **mapas de políticas** a las **interfaces**.