

## Unidad 8

---



# Configuración de acceso a internet desde una LAN

## Planificación y Administración de Redes



# Índice



## 8.1. ACL. Listas de control de acceso

- 8.1.1. Definición y características
- 8.1.2. Funcionamiento de las ACL
- 8.1.3. ACL estándar
- 8.1.4. ACL extendida

## 8.2. Firewall y DMZ

- 8.2.1. Definición
- 8.2.2. Arquitectura de firewalls

## 8.3. Técnicas de conmutación

- 8.3.1. Conmutación de circuitos
- 8.3.2. Conmutación de paquetes

## 8.4. Tecnologías de acceso a Internet

- 8.4.1. La red digital de servicios integrados (RDSI)
- 8.4.2. La línea de abonado digital (DSL)
- 8.4.3. FTTx. Accesos mediante fibra
- 8.4.4. La red inalámbrica (WiMAX)

## 8.5. Traducción de direcciones. NAT/NAPT

- 8.5.1. Funcionamiento de NAPT
- 8.5.2. Configuración de NAPT en routers CISCO



## Introducción

La conexión de una red de área local no se basa simplemente en la navegación online. En los temas anteriores hemos ido viendo como configurar y preparar toda nuestra estructura para poder tener una red disponible, pero nos queda lo más importante, conectarnos a Internet.

Para poder realizar la conexión de manera segura, lo primero que deberemos de hacer será establecer unas reglas de acceso a nuestros sistemas, y de igual modo, de salida de información por si puede haber riesgo de fuga.

Aunque estos son buenos mecanismos, hay veces que necesitaremos de otros tipos de barreras como aplicaciones de terceros frente a ataques, los firewalls.

Una vez protegido el sistema, debemos de saber como trabaja la red, basando en las técnicas de conmutación.

Existen diferentes tecnologías de acceso a Internet y debemos de saber cual es la más adecuada para nuestra red y en la medida de lo posible, como funcionan.

Por último, es de vital importancia tener claro la diferencia entre IP privada y pública, y como pasar de una a otra sin que el exterior lo sepa.

## Al finalizar esta unidad

- + Conoceremos que son las listas de control de acceso y cómo funcionan.
- + Sabremos las diferencias entre ACL estándar y extendida y como trabajar con ellas.
- + Podremos determinar las definiciones de Firewall y de zona militarizada en temas de ciberseguridad.
- + Seremos capaces de distinguir los tipos de arquitecturas de firewall más usados.
- + Distinguiremos los dos tipos más usados en técnicas de conmutación.
- + Sabremos cuales son las tecnologías de acceso a internet más usadas.
- + Podremos determinar como funciona la tecnología basada en fibra óptica.
- + Seremos capaces de definir la tecnología NAT y de configurarla en los routers CISCO.



# 8.1.

## ACL. Listas de control de acceso

### 8.1.1. Definición y características

Las ACL o listas de control de acceso las hemos visto e introducido en otra asignatura de este mismo grado, pero en los *routers*, su función es diferente. En un enrutado, las ACL son las listas con las condiciones que van a hacer que el tráfico que pasa por ese *router* tome un camino u otro también.

Las ACL toman valores como la dirección de origen y destino, los números de puerto y el protocolo superior para decidir si un paquete es aceptado o rechazado por un *router*.

En cada paquete que llega al *router* se realiza una serie de comprobaciones en las que el orden importa, ya que, si la primera condición ya se cumple, se sale de la ACL, sin comprobar el resto de las condiciones.

A modo de resumen, podríamos decir que las ACL son normas que se usan para el procesamiento de los paquetes que entran y salen del *router*.

Cada una de las interfaces de un *router* puede servir para salida y para entrada de paquetes, pero su ACL no tiene por qué ser el mismo para ambos, de hecho, suele ser diferente.

Lógicamente, si no hay ACL configuradas, se permite el paso de todos y cada uno de los paquetes por nuestro *router*.

Si queremos modificar una ACL, lo preferible es eliminarla y crearla de cero.

Por último, cabe destacar que en general la última línea de una ACL no se suele poner explícitamente y siempre es *denegar cualquiera*.

Hay varios tipos de ACL:

- > **Estándar:** solo se comprueba la dirección de origen del paquete.
- > **Extendidas:** se comprueba la dirección de origen, la dirección de destino, el protocolo y los puertos.
- > **Dinámicas:** en este caso, se exige que haya autenticación en el *router* por parte del usuario vía *Telnet*.
- > **Reflexivas:** en este tipo, se permite el tráfico saliente, pero se limita el tráfico de regreso a modo de respuestas al tráfico que se inicia en el *router*.
- > **Basadas en tiempo:** se define un intervalo de tiempo real, donde se valida el tráfico de paquetes que pasa por nuestro enrutador.

Solo vamos a ver los dos primeros tipos, porque son las más usadas en la actualidad.



### 8.1.2. Funcionamiento de las ACL

Cada vez que llega un paquete al *router*, se valida si cumple las sentencias de la ACL en el orden en el que se han creado.

En cuanto cumpla con alguna de las sentencias, se para la comprobación.

```
Router(config)#  
Router(config)#access-list 1 permit any  
Router(config)#
```

Imagen 1. Creación de ACL en router Cisco

Siempre se suele colocar al final de la lista, por defecto, la sentencia implícita **deny any**, que indica que, si ninguna regla anterior se ha cumplido, se deniega el paquete.

Para cada una de las tramas que se registran, el proceso con las ACL es el siguiente:

1. Si la trama se acepta, se procede a des encapsularla y se comprueba si hay alguna ACL funcionando en esa interfaz de entrada.
2. Si existe una ACL y además el paquete es denegado, se descarta.
3. Si no existe ninguna ACL o, aunque existente, el paquete es aceptado, se busca cual es la interfaz de salida en la tabla de enrutamiento.
4. Vemos si la interfaz que va a dar salida al paquete tiene asociada alguna ACL.
5. Si existe una ACL y además el paquete es denegado, se descarta.
6. Si no existe ACL, o se acepta el paquete con las sentencias, se da salida al paquete de datos.

El siguiente diagrama resume este proceso:

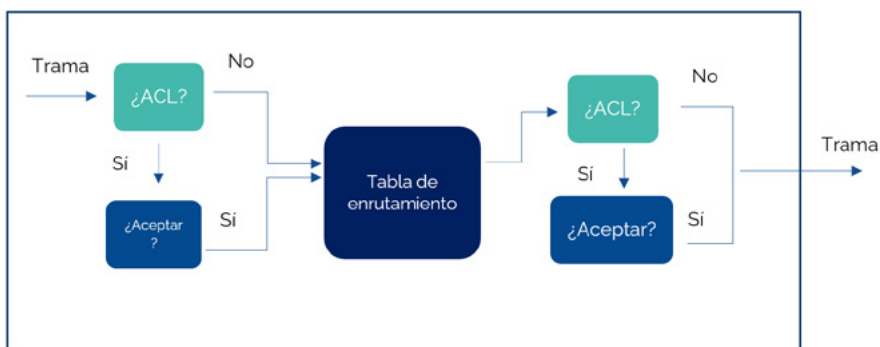


Imagen 2. Procesamiento de tablas con ACL





### 8.1.3. ACL estándar

---

#### Numeración

Para identificar una ACL, se usa un número único que no se puede repetir.

El número que una ACL tenga asignado, determina de que tipo se trata:

- > Las ACL estándar se numeran del 1 al 99 y del 1300 al 1999.
- > Las ACL extendidas se numeran del 100 al 199 y del 2000 al 2699.

#### Máscara wildcard

Este tipo de máscara es lo contrario a la máscara de red tradicional, ya que se intercambian los 0 por los 1. Si tomamos la dirección 10.24.0.1 que tiene una máscara de red 255.0.0.0, su máscara *wildcard* será 0.255.255.255.

En la máscara *wildcard*, los 0 indican los bits comparables y 1 para indicar los bits que son descartables.

#### Creación

Vamos al modo de configuración global del *router* y usamos el comando:

```
access list número_ACL {permit | deny} dirección_origen

Router(config) #
Router(config) #access-list 1 permit 10.20.0.1
Router(config) #
```

Imagen 3. Creación de ACL 2

El comando creará una ACL en caso de no existir, para definir una regla de permiso o no del tráfico de la red. En una primera instancia, cuando se crea, no están asignadas a ninguna interfaz del *router*.

Tenemos tres formatos diferentes para poder indicar cual es la dirección de origen a la que se le aplica la regla:

- > **host dirección\_IP**: se representa un único host.
- > **dirección\_de\_red máscara\_wildcard**: los dos valores en conjunto representan una dirección de red y su máscara *wildcard*.
- > **any**: se representa cualquier equipo de la red.



## Asignación de la ACL a una interfaz

Cuando hemos creado una ACL, esta no va a funcionar hasta que no se la asigne a una interfaz. Para poder realizar esta asignación, debemos de usar los siguientes comandos en un *router* CISCO:

1. Seleccionamos la interfaz con `interface interfaz`.
2. Lanzamos el comando siguiente para asignar la ACL específica a la interfaz:

```
ip access-group número_ACL {in | out}

Router(config)#
Router(config)#interface GigabitEthernet0/0
Router(config-if)#ip access-group 1 in
Router(config-if)#
```

Imagen 4. Asignar ACL a una interfaz

Podemos ver que, en el segundo comando, se pueden diferenciar al final dos parámetros usables;

- > **In:** el tráfico a filtrar es el que **entra** por la interfaz que hemos seleccionado.
- > **Out:** el tráfico a filtrar es el que **sale** por la interfaz que hemos seleccionado.

## Ubicación de la ACL estándar

La ACL estándar debe de encontrarse lo más cerca posible de su destino.

Esto es debido a que las ACL estándar solo usan la dirección de origen para el filtrado del tráfico.

## Comprobación

Para comprobar donde se encuentra los errores, o comprobar las ACL que han hecho que no pase cierto tráfico, se usa el comando:

```
show access-list
```

```
Router(config-if)#exit
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console
show access-list
Standard IP access list 1
  10 permit host 10.24.0.1

Router#
```

Imagen 5. Comprobar ACL



## Eliminar una ACL

Para eliminar una ACL, se debe de usar el comando:

```
no access-list número_ACL
```

```
Router(config)#no access-list 1
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console
show access-list
Router#
```

Imagen 6. Eliminar ACL

## Liberar una interfaz de su ACL

Para liberar una interfaz de su ACL, es decir, desasignársela, se usa el comando:

```
no ip Access-group número_ACL {in | out}
```

```
Router(config)#
Router(config)#interface GigabitEthernet0/0
Router(config-if)#no ip access-group 1 in
Router(config-if)#
```

Imagen 7. Liberar una interfaz de una ACL

## Ejemplo

Se necesita que a través de una **ACL** el equipo del Alumno1 no pueda llegar a ningún equipo de la sala del profesorado, sin embargo, el Alumno2 si va a poder.

### 1. Introducción de la ACL.

Como el objetivo es que el alumno 1 no pueda comunicarse con ningún ordenador en la sala de profesores, tenemos que crear una ACL que permita realizar esta condición.

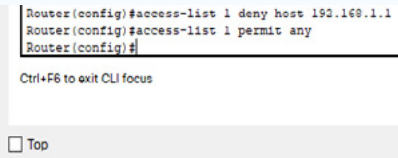
Creación de la ACL	
<pre>Router (config)# access-list (nº de lista) (permit   deny)</pre> <p>Podemos generarlo de diferentes maneras:</p> <p><b>host</b> (especificar un único equipo)</p> <p>&lt;dir_red&gt; &lt;máscara_wildcard&gt;</p> <p><b>Any</b> (cualquier equipo)</p> 	<p><b>PASO 1:</b></p> <p>Crearemos una ACL estándar en el <b>router</b>. Esta denegará el tráfico de <b>alumno1</b> y permitirá el tráfico de los demás equipos.</p> <p><b>NOTAS:</b></p> <p>El orden en el que se escriben las sentencias de una ACL es primordial, por lo que el orden es lo más importante.</p>

Imagen 8. Ejemplo ACL 1

Denegamos el tráfico del equipo del **alumno 1** y permitimos el resto.





### Asignación de la ACL a una interfaz

#### Asignación

Cuando es creada la ACL, es necesario asignarla a una interfaz, si no, la ACL, no tendrá ningún efecto.

**Router (config)# interface <FastEthernet/Gig>**

**Router(config-if)# ip access-group <número\_ACL>**  
**<in | out>**

**in** = tráfico que entra por la interfaz seleccionada.

**out** = tráfico que sale por la interfaz seleccionada.

```
Router>
Router>config t
% Invalid input detected at '^' marker.

Router>enable
Router#config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface fas
Router(config)#interface FastEthernet 0/1
Router(config-if)#ip acc
Router(config-if)#ip access-group 1 out
Router(config-if)#exit
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console

Router#show acc
Router#show access-lists
Standard IP access list 1
10 deny host 192.168.1.1
20 permit any

Router#
```

Imagen 9. Ejemplo ACL 2

Para acabar, con el comando **show access-list** podemos ver las listas generadas.

#### PASO 1:

Como norma general, la ACL, se instala lo más cerca posible del destino, es por ello que aquí, la asignamos a la interfaz de FastEthernet 0/1

#### NOTAS:

Las ACL estándar solo especifican la dirección de origen del tráfico.

### Eliminar una ACL

Como la configuración de la ACL es tan delicada con el orden de las normas establecidas, cuando se necesite modificar, lo mejor es eliminar la ACL y volver a realizarla.

### Eliminación de una ACL

Para eliminar una ACL, simplemente hay que añadir:

**Router (config)# no access-list <nº de list>**

```
Router(config)#no access-list 1
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console

Router#show access-list
Router#
```

Imagen 10. Ejemplo ACL 3



### Liberar una interfaz de su ACL

Para liberar a una interfaz de una ACL, simplemente hay que añadir:

```
Router (config)# interface <FastEthernet/Gig>  
Router0(config-if)# no ip access-group <número_ACL>  
<in|out>
```

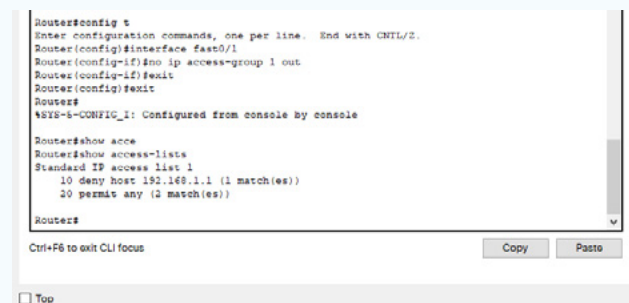


Imagen 11. Ejemplo ACL 4

Aquí vemos como el ping que realiza el Alumno 1, ahora sí puede llegar a la sala de profesores.

Llegados a este punto, ya tenemos terminada la práctica. Teniendo un *router* capaz de poder restringir el tráfico.

## 8.1.4. ACL extendida

### Creación

Para crear una ACL extendida, lanzamos los siguientes comandos:

```
access-list número_ACL [Dynamic nombre] {deny | permit} protocolo dirección_de_origen [operador puerto] dirección_de_destino [operador puerto] [tipo_de_icmp] [established] [precedence p] [tos t] [time-range tiempo] [remark comentario]
```

Con este comando lo que hacemos es crear, en caso de no existir, una ACL extendida para bloquear o permitir el tráfico. Solo puede existir una ACL extendida para cada protocolo e interfaz. Cuando creamos la ACL, esta aún no está asociada a ninguna interfaz.

### Parámetros

- > **num\_ACL:** número de ACL en el rango de 100 a 199 o de 2000 a 2699.
- > **dynamic:** se le asigna un nombre a la ACL.
- > **protocolo:** los que se suelen usar son:
  - » eigrp
  - » gre
  - » icmp
  - » ip
  - » ospf
  - » tcp
  - » udp



- > **dirección\_de\_origen y dirección de destino:** igual que en las ACL estándar, lo único que ahora también se especifica la dirección de destino.
- > **operador puerto:** se pueden usar operadores en algunos protocolos como tcp o udp, la siguiente tabla ilustra los operadores:

Operadores aplicables al puerto en ACL extendida	
Operador	Se aplica a paquetes...
<b>eq puerto</b>	Por un número de puerto igual que el indicado
<b>gt puerto</b>	Por un número de puerto mayor que el indicado.
<b>lt puerto</b>	Por un número de puerto menor que el indicado.
<b>neq puerto</b>	Por un número de puerto distinto que el indicado.
<b>range puerto1 puerto2</b>	Por un rango de puertos indicados.

- > **tipo\_icmp:** una vez que se selecciona el protocolo ICMP, se pueden indicar los tipos de mensaje que ya se vieron en unidades anteriores.
- > **established:** el tráfico TCP está permitido siempre que el paquete use una conexión establecida.
- > **precedence:** se filtra el tráfico dependiendo del nivel de precedencia.
- > **tos:** dependiendo del tipo de servicio, se filtrará el tráfico de un modo u otro.
- > **time\_range:** se establece un intervalo de tiempo en el cual la ACL permanece activa
- > **remark:** se añaden comentarios a la ACL.

### Ubicación de la ACL extendida

Las ACL extendidas deberían de estar lo más cerca posible del origen del tráfico que se deniega. Esto se hace con la intención de que el tráfico que no queremos sea desechado lo antes posible, así no consume recursos o genera peligros de manera innecesaria.



# 8.2.

## Firewall y DMZ

### 8.2.1. Definición

Es una de las medidas de seguridad más importantes a la hora de configurar una red LAN conectada a internet. Analiza todos los paquetes de datos que entran y salen por sus interfaces y, apoyándose en un conjunto de reglas, determinan de qué manera actuar con cada uno de ellos, es decir, *aceptando o rechazando* el mensaje.

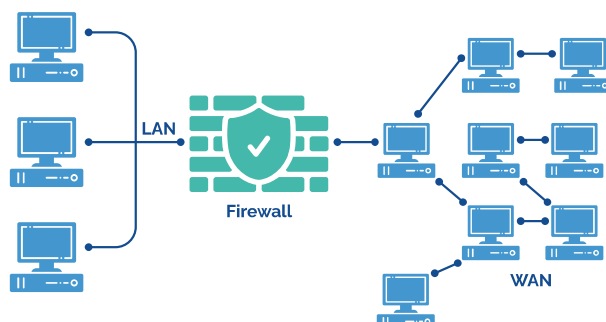


Imagen 12 Reglas presentes en una red LAN. El firewall suele representar como un símbolo de un muro

Los *routers* contienen un cortafuegos en su interior, por lo que podrían considerarse como *cortafuegos hardware*. La diferencia entre uno y otro reside en que el *software* necesario (*firmware*), permita al administrador a crear un conjunto de *reglas* de filtrado.

Por ejemplo, los *routers* domésticos proporcionan un conjunto de reglas muy reducido preestablecido por el cortafuegos, aparte de la redirección de puertos.

Los cortafuegos en este tipo de dispositivos deben preocuparse únicamente de los paquetes de datos que reciben (INPUT), y los que generan ellos mismos (OUTPUT), y los más importantes, los que pasan a través de ellos (FORWARD).

Así mismo, todos los sistemas operativos incorporan su propio *cortafuegos software*, el cual controla las conexiones entrantes y salientes. Desde la versión 8 de Windows, se permite definir reglas según puerto, aplicaciones que van a recibir las conexiones, etc. Los sistemas Linux, en cambio, incorporan un cortafuegos orientado únicamente a los paquetes que entran y salen por las interfaces de red. Es muy potente, pero difícil de configurar, ya que se debe hacer mediante líneas de comandos.

En el caso de los cortafuegos *hardware*, se pone a disposición del usuario un sistema operativo interno para poder instalarlo en equipos PC, brindando la posibilidad de crear un cortafuegos *software*. Como es el caso de Mikrotik. Al igual que este, existen también cortafuegos *open source* basados en Linux/UNIX, como pfSense.



Imagen 13. Cortafuegos *hardware* Barracuda. Fuente: [commons.wikimedia.org](https://commons.wikimedia.org)



Un cortafuegos sirve para detener conexiones de red no deseadas. No se debe pensar que un cortafuegos protege contra todas las amenazas que hemos descrito anteriormente por varios motivos:

Si se abre un puerto específico, ya hay un canal por el cual recibir ataques. Esto implica que se deben tomar medidas *adicionales* para proteger los ataques destinados a ese puerto en concreto.

Contiene un número limitado de reglas que protegen ante cierto número de atacantes, dicho de otra manera, estará indefenso ante otros tipos de ataques no contemplados en esas reglas. Por eso, la administración es la encargada de actualizar dichas reglas.

Algunos ataques, como el conocido DoS (denegación de servicio) es imposible de evitar, aunque según las medidas de seguridad incorporadas, ayudan a mitigar sus efectos.

Para terminar, podemos decir que un cortafuegos es una medida de protección más, aunque podemos afirmar que estamos ante una de las medidas más importantes porque *reduce la superficie de ataque de la red local*.

### 8.2.2. Arquitectura de firewalls

Dependiendo de su misión y funcionamiento, podemos esklar tres tipos de arquitecturas de *firewalls*.

#### Host Dual-Homed

La arquitectura *dual-homed host* se constituye con un equipo que cuenta con dos interfaces de red y que usa un *software* específico para filtrar los paquetes. Se suele denominar esta arquitectura como **bastión** y su función es actuar como *router* entre las redes conocidas.

Los paquetes de una red a la otra no son enrutados de manera directa. La red interna y externa no se pueden comunicar entre sí, pero ambas tienen comunicación directa con el *dual-homed host*.

Este sería un primer nivel de seguridad en cuanto a arquitectura de *firewalls* se refiere.

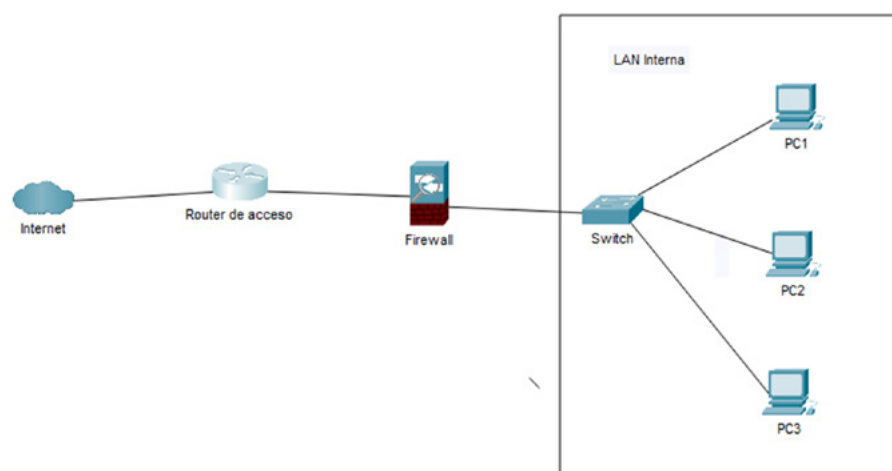


Imagen 14. Host Dual-Homed



### Arquitectura Screened Host

El también llamado *firewall* de 3 patas, esta técnica lo que trata de hacer es una combinación entre un *screening router* y un *host bastión*. En esta arquitectura, el principal nivel de filtrado proviene del filtrado de los paquetes mediante las ACL.

Estas ACL deben de haberse definido previamente por el administrador de la red, usando las direcciones IP de origen y destinos y los servicios usados.

Se configura el filtrado del tráfico en el *screening router* de tal modo que el *host bastión* se trata de un sistema que es accesible desde la red externa, siendo este el único elemento de la red interna accesible desde Internet.

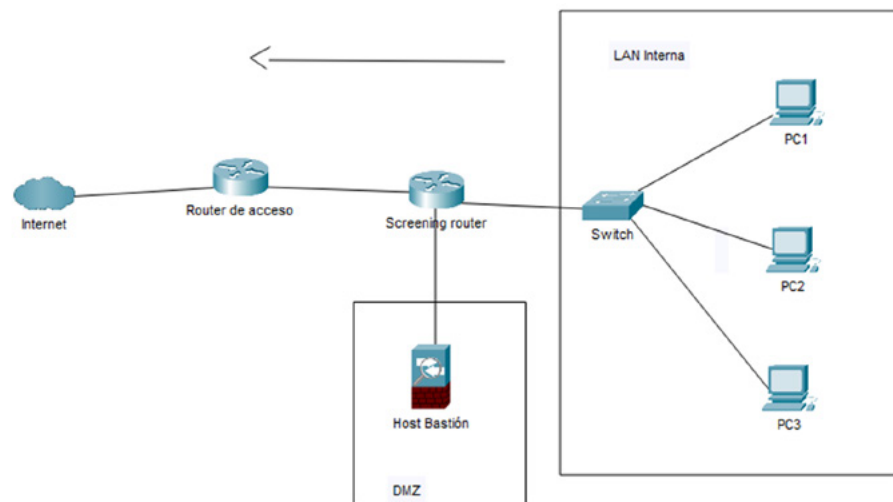


Imagen 15. Screened Host

Es una característica de los *routers* domésticos que se utiliza para delegar la tarea de cortafuegos en otro equipo. Al definir sus valores, se indica que todos los paquetes entrantes se re-envíen a la IP especificada.



Por otro lado, si se necesita dar acceso desde el exterior a un protocolo diferente a TCP o UDP, es una configuración inevitable a la hora de gestionar las conexiones.



Imagen 16. Configuración DMZ en router Vodafone.

### Arquitectura Screened Subnet

Esta es la arquitectura más segura de todas, porque se aumenta un nivel más de seguridad sobre la anterior arquitectura al añadir un perímetro a la red que la asila de Internet. El host bastión, aunque pueden ser protegido, es la máquina más vulnerable ya que se puede visualizar desde la red externa.

Cuando usamos una red intermedia o un perímetro para asilar el host Bastión, lo que conseguimos es que, si es atacado, el impacto del ataque sea menor. En este tipo de arquitectura, tenemos dos *screening router* conectados cada uno de ellos al perímetro, a los extremos para ser más concretos, el primero entre el perímetro y la red externa y el segundo entre el perímetro y la red interna. Si vamos a recibir un ataque, el atacante debe de pasar por ambos *routers* para poder llegar a la red interna.

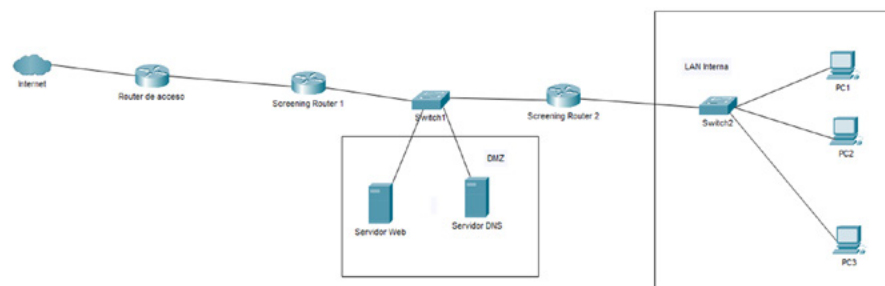


Imagen 17. Screened Subnet



# 8.3.

## Técnicas de conmutación

Las **técnicas de conmutación** son herramientas o mecanismos que nos ayudan a establecer conexiones entre equipos en el contexto de redes WAN, repartiendo el medio físico entre todas las conexiones para poder emitir los datos incluso a largas distancias.

Estas técnicas se crean derivadas de la imposibilidad de la conexión de todos los terminales punto a punto y teniendo en cuenta que la difusión de las LAN también es imposible debido a la distancia.

### 8.3.1. Conmutación de circuitos

Cuando usamos esta técnica de conmutación, debemos de tener entre los dos equipos que se intente comunicar, una conexión constante establecida, sin decaer. Para esta conexión no siempre tiene que ser el mismo cable o medio, puede haber varias líneas que se interconecten entre sí para conseguir siempre tener conexión.

Antes de comenzar con la comunicación de paquetes, cada uno de los dos equipos deben de estar conectados mediante una llamada.

En esta técnica de conmutación, se crea un camino físico entre origen y destino durante el tiempo que la transmisión ocurra.

El camino es exclusivo para los extremos, **no se comparte** con demás usuarios, por lo que, si no hay comunicación o hay muy poca, realmente se está desaprovechando el medio.

Tenemos dos tipos de conmutación de circuitos, o mediante conmutación por división en el espacio o mediante conmutación por división en el tiempo.

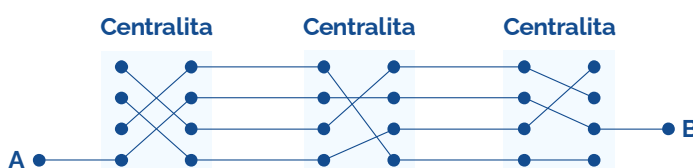


Imagen 18. Conmutación de circuitos por división en el espacio

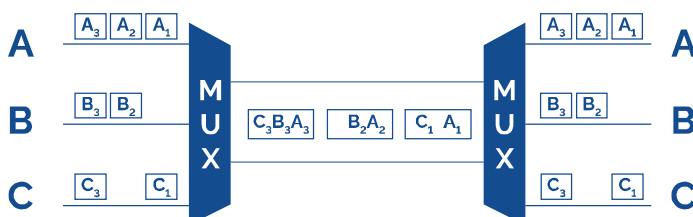


Imagen 19. Conmutación de circuitos por división en el tiempo



### 8.3.2. Conmutación de paquetes

Este tipo de conmutación no necesita que exista una misma conexión entre emisor y receptor durante toda la transmisión.

La red de transporte creada en esta conmutación se trata de una malla entera de varios nodos que envían y reciben mensajes de comunicación diferenciados en pequeñas partes llamadas **paquetes**. En estos paquetes no solo se almacenan datos, también una cabecera con información adicional.

La red conmutada de paquetes tiene varias conexiones entre todos los *routers* de la red, de modo que cada vez que uno de estos recibe un paquete, es su decisión cual será la salida de dicho paquete fijándose en la cabecera.

Al ser los paquetes una unidad más pequeña que el mensaje entero, no se almacena en discos, sino en memoria central del *router* mientras que se procede a su reenvío.

Como se ha dicho en unidades anteriores, no hay una razón que dictamine que los paquetes deben de llegar en orden, de hecho, nunca es así, o casi nunca.

Cada uno de los *routers* realiza las funciones de almacenar y transmitir y la de encaminar la información.

Este tipo de conmutación no es la adecuada si queremos transmitir datos sin voz, porque los datos se suelen transmitir a ráfagas, lo que hace que se encuentre la línea inactiva durante el tiempo que no hay información que transmitir.

Tenemos dos tipos de conmutación de paquetes: modo datagrama y modo circuito virtual.

#### Modo Datagrama

La red conmutada de paquetes que funciona en **modo datagrama** es la que se da cuando el equipo que recibe la información es también el encargado de reorganizar y ordenar los paquetes que se reciben, por lo que, si hay algún paquete que no sabemos su orden o que está perdido, es el receptor quien debe solicitar su reenvío.

Como mayor ejemplo de este tipo de red conmutada tenemos Internet.

#### Modo Circuito virtual

Si es la propia red la que ordena los paquetes antes de entregarlos al receptor del mismo modo en que salieron, se trata de una red conmutada de paquetes funcionando en modo **circuito virtual**.

El circuito virtual simula la red de conmutación de circuitos usando como medio de transporte una red de conmutación de paquetes.

Para cada mensaje que queramos enviar, se establecerá previamente un camino virtual hasta el destino que pasará por los *routers* necesarios dependiendo del tráfico que tenga la red en cada momento, siguiendo todos los paquetes del mensaje la misma ruta de modo consecutivo, uno tras otro.

Los paquetes directamente llegan ordenados y si por cualquier casualidad uno de los paquetes no está, este se retransmitirá inmediatamente.

Las fases de la comunicación son:

1. Establecimiento de la conexión.
2. Transferencia de datos.
3. Liberación de la conexión.

#### NOTA

No es lo mismo una conmutación de paquetes de circuito virtual que la conmutación de circuitos. En la segunda no compartimos la línea, mientras que en la primera sí.

Ejemplos de redes de conmutación de paquetes con circuitos virtuales son: X.25, *Frame Relay* y ATM.



## 8.4.

### Tecnologías de acceso a Internet

Cuando en casa, oficina, aula, etc., deseamos tener conexión a Internet, necesitamos de un dispositivo intermedio que nos comunique nuestra red privada con la centralita que haya más cercana. Por lo general, estos dispositivos nos los proveen los proveedores de servicios, ISP, y suele ser un *router*, aunque su nombre técnico es DCE, equipo circuito de datos. La línea que une la centralita más próxima con nuestro DCE se llama bucle local o de abonado.

En las redes de telecomunicaciones podemos distinguir tres niveles:

- > **Red de transporte o red troncal.** Esta red se encuentra formada por centrales o por nodos primarios que están conectados mediante una fibra óptica en forma de anillo, que es lo mejor para las largas distancias.
- > **Red de distribución o agregación.** Esta se encuentra formada por nodos secundarios con el mismo medio, Fibra óptica. Es en esta red donde surge la conmutación de circuitos o de paquetes.
- > **Red de acceso.** Se encuentra formada por el bucle local y es la que permite que el cliente final pueda conectarse a la red de agregación. En este caso el medio puede ser, fibra óptica, cable de cobre o el medio inalámbrico.

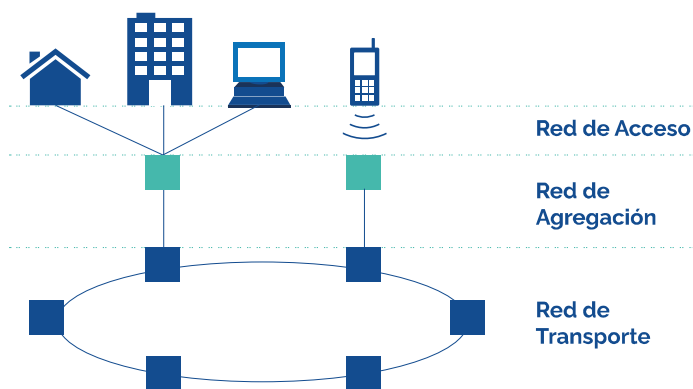


Imagen 20. Red de telecomunicaciones

#### 8.4.1. La red digital de servicios integrados (RDSI)

Esta red que usa el protocolo HDLC a nivel de enlace permite que haya conexiones digitales a cada extremo para poder proporcionar tanto servicio de voz como de datos.

EL DCE es un dispositivo al que llamamos NT1 y su bucle local es un par trenzado por el que viajan las señales digitales. EL acceso básico DSI está compuesto por 2 canales de comunicación de 64 kbps cada uno de ellos. Cada canal puede usarse tanto para voz como par datos. Además de estos dos, existe un tercer canal de 16 kbps usado para señalización y provisión de servicios suplementarios.



### 8.4.2. La línea de abonado digital (DSL)

La red DSL se trata de una familia de tecnologías de acceso donde destaca ADSL, que añade al principio una A para señalar que es asimétrica. Esto es debido a que tiene una mayor tasa de descarga que de subida.

El ADSL divide el ancho de banda del cable, 1 MHz, en tres bandas distintas:

1. **Va de 0 a 4 kHz.** Es usada para el servicio telefónico regular al que llamamos POTS.
2. **De 25 a 138 kHz.** Es la usada para el tráfico de los datos de subida.
3. **De 138 kHz a 1 MHz.** Es la que usamos para el tráfico de los datos de bajada.

#### Tipos de ADSL

##### > ADSL

- » Usa el estándar ITU-T G.992.1.
- » El módem de ADSL se conecta con el módem de la central situando en el otro lado el bucle local.
- » Su velocidad máxima de subida es de 1 Mbps.
- » Su velocidad máxima de bajada es de 8 Mbps.
- » Realmente, en el día a día, las velocidades de subida están sobre los 0,8 Mbps y las de bajada sobre los 2 Mbps.

##### > ADSL2

- » Usa el estándar ITU-T G.992.3 y el 4.
- » Se mejora la eficiencia en la modulación, con menos ruido y atenuación, dando lugar a la posibilidad de hasta 9 km de distancia con la central.
- » Su velocidad máxima de subida es de 1 Mbps.
- » Su velocidad máxima de bajada es de 12 Mbps.

##### > ADSL2+

- » Usa el estándar ITU-T G.992.5
- » Su frecuencia máxima se incrementa hasta los 2,2 MHz
- » Su velocidad máxima de subida sigue siendo 1 Mbps.
- » Su velocidad máxima de bajada aumenta hasta los 24 Mbps.

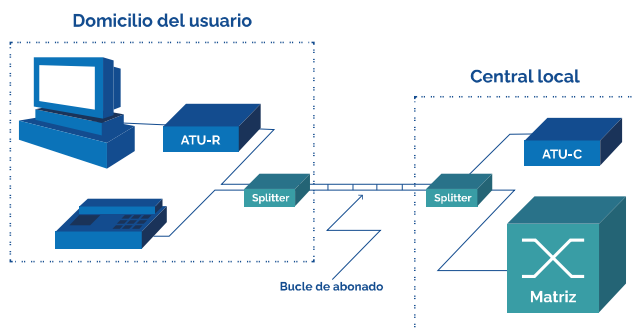


Imagen 21. Funcionamiento ADSL



### 8.4.3. FTTx. Accesos mediante fibra

La fibra óptica es el mejor medio cableado para la transmisión de datos gracias a que es tremendamente resistente frente a las interferencias además de que tiene un gran ancho de banda. Dependiendo de que tipo de instalación queramos hacer, se instalará más o menos cantidad de fibra debido a su gran coste.

En la siguiente imagen podemos ver una clasificación de la fibra dependiendo de hasta donde se extienda su instalación:

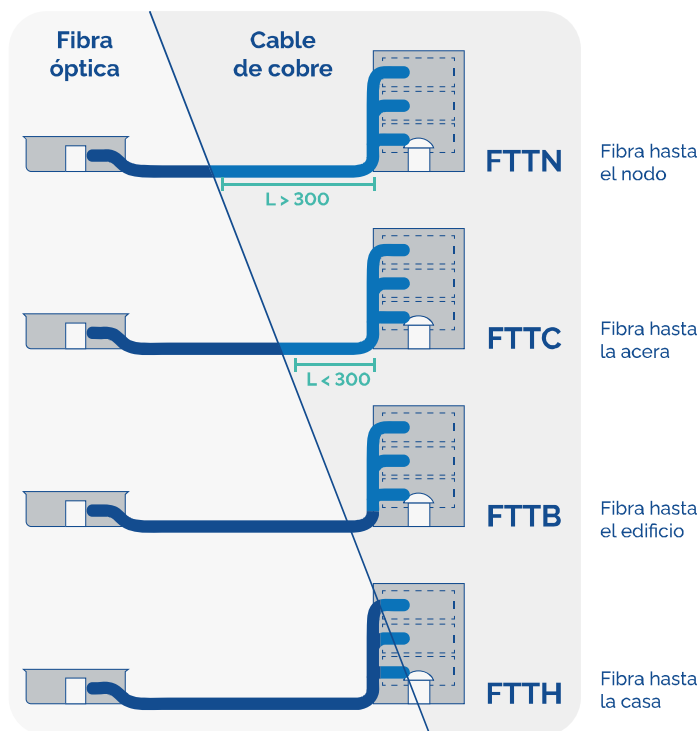


Imagen 22. Tipos de Fibra óptica según cercanía al usuario

La última de las tecnologías que se nos presentan en la imagen anterior, la tecnología FTTH es la que permite que la fibra llegue hasta el usuario final. En este caso, la red de acceso entre el usuario y el último nodo de distribución se podría realizar con una o dos fibras para cada usuario o con una red óptica pasiva o PON en estructura de árbol con una fibra en el lado de la red y varias para el lado del usuario.

La red óptica pasiva se forma en principio por:

- > Un módulo OLT en el nodo central.
- > Un divisor óptico o *splitter*.
- > Varias ONT en el final de la línea.





#### 8.4.4. La red inalámbrica (WiMAX)

Las redes WiMAX son las que usan esta misma tecnología que se trata de una tecnología de acceso a redes WAN. Esta funciona con microondas y ondas de radio que se mueven en las frecuencias de 2,3 a 3,5 GHz y son capaces de alcanzar hasta 50km.

Esta tecnología se diseñó en 2004 y era especialmente para poder disponer de conexión a Internet en sitios donde los cables no llegan. Se encuentra definida y regulada por el estándar IEEE 802.16.

Las tecnologías WiMAX funcionan mediante la instalación de estaciones base. Estas estaciones base se conectan con múltiples usuarios mediante ondas de radio, y permiten transportar voz, datos y video. Los usuarios pueden estar situados a grandes distancias, y se conectan gracias a unos paneles que hay instalados en el exterior de las localizaciones.

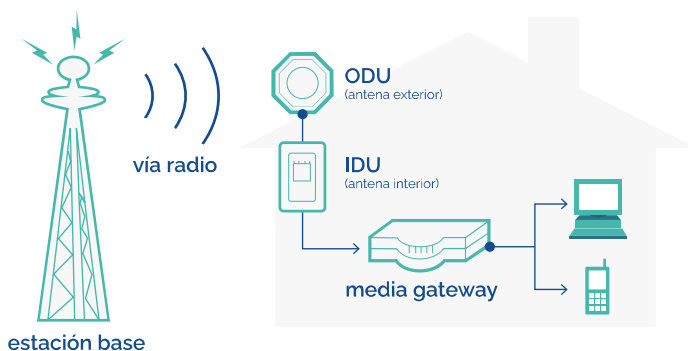


Imagen 23. Funcionamiento WiMAX

Dentro del estándar que define las redes WiMAX, tenemos dos variantes:

> **802.16d para acceso fijo:**

En este caso el enlace está establecido por ondas de radio entre la estación base y la antena instalada en la localización de conexión. Tiene una velocidad máxima de 100 Mbps para un canal de 20 MHz, aunque realmente, en función de la distancia, la velocidad es mucho menor.

> **802.16e para acceso móvil:**

El usuario se puede mover de una manera bastante similar a la de la telefonía móvil con internet, pero aún se encuentra en desarrollo.

Solo pueden certificar el cumplimiento de este estándar y la operabilidad de las redes los participantes del *Wimax Forum*.



# 8.5.

## Traducción de direcciones. NAT/NAPT

### 8.5.1. Funcionamiento de NAPT

Cada vez que una empresa u organización quiere tener conexión a internet, necesita de una o varias direcciones IP públicas que sean visibles en la red. Por otro lado, los hosts internos de una red no tienen direcciones públicas, sino privadas, por lo que habrá que realizar una conversión de estas direcciones a la hora de mandar información al exterior. Esta transformación es la conocida como NAT, *Network Address Translation*.

Cuando lo que hacemos es conectarnos a través de una única dirección pública con el exterior, como en el caso de una casa con un único *router*, se usa además algún puerto que nos ayude a dirigir el tráfico hacia el equipo deseado, que es una implementación de NAT conocida como NAPT, *Network Address and Port Translation*.

La tecnología NAPT nos ayuda resolver el problema planteado desde hace ya mucho tiempo con el agotamiento de direcciones públicas de IPv4, y hasta que se use de manera definitiva IPv6. Otra ventaja que tenemos con NAPT es que es un mecanismo más seguro ya que desde el exterior no pueden visualizar la verdadera dirección IP de nuestros dispositivos.

Los mapeos que realiza NAPT son del siguiente tipo:

**IP privada: n° puerto local → IP pública: n° puerto asignado**

El número de puerto asignado es el que elige el *router*, siendo posible usarlo únicamente cuando se termina la conexión que el equipo local ha generado.

### 8.5.2. Configuración de NAPT en routers CISCO

Tenemos que seguir los siguientes pasos para poder configurar NAPT en nuestro *router*:

1. Lo primero que debemos hacer es entrar en el modo de configuración del *router*.
2. Ahora, definimos una ACL que deje pasar a las direcciones locales que vamos a traducir.

```
Router(config)#  
Router(config)#access-list 5 permit 10.0.0.14 0.255.255.255  
Router(config)#
```

Imagen 24. NAPT en Cisco 1



- Lo siguiente que debemos hacer es crear la traducción, indicando la ACL que hemos creado antes y el puerto externo del *router* porque se va a salir la dirección, para esto, usamos el comando:

```
ip nat inside source list n interface puerto_ex-  
terno overload
```

```
Router(config)#  
Router(config)#ip nat inside source list 5 interface Gig0/0/1 overload  
Router(config)#
```

---

Imagen 25. NAT en Cisco 2

- Ahora debemos de indicar la interfaz interna del *router* y la marcamos como *inside* con el comando:

```
ip nat inside
```

```
Router(config)#interface GigabitEthernet0/0/2  
Router(config-if)#ip nat inside  
Router(config-if)#exit  
Router(config)#
```

---

Imagen 26. NAT en Cisco 3

- A la interfaz externa del *router*, la marcamos como *outside*. El comando es:

```
ip nat outside
```

```
Router(config)#  
Router(config)#interface GigabitEthernet0/0/1  
Router(config-if)#ip nat outside  
Router(config-if)#exit  
Router(config)#exit  
Router#
```

---

Imagen 27. NAT en Cisco 4

- Por último, si por lo que sea, tenemos un servidor en nuestra red que es accesible desde el exterior, establecemos una regla como la siguiente para que el tráfico que llegue por el puerto 80 sea redirigido a nuestro servidor interno:

```
Router(config)#  
Router(config)#ip nat inside source static tcp 10.0.0.14 80 17.0.127.1 80  
Router(config)#
```

---

Imagen 28. NAT en Cisco 5

*En este caso, como dirección pública hemos usado una del rango 17.0.0.0, que en la actualidad están todas en propiedad de la empresa Apple, Inc.*



## Gestión de una configuración NAT

En las siguientes imágenes veremos una serie de comandos de los *routers* Cisco que son interesantes para la configuración NAT en ellos:

```
Router#  
Router#show ip nat translations  
Pro Inside global      Inside local      Outside local      Outside  
global  
tcp 17.0.127.1:80      10.0.0.14:80      ---               ---
```

Imagen 29. Mostrar tabla NAT

```
Router#show ip nat statistics  
Total translations: 1 (1 static, 0 dynamic, 1 extended)  
Outside Interfaces: GigabitEthernet0/0/1  
Inside Interfaces: GigabitEthernet0/0/2  
Hits: 0 Misses: 0  
Expired translations: 0  
Dynamic mappings:  
Router#
```

Imagen 30. Mostrar estadísticas NAT

```
Router#  
Router#debug ip nat  
IP NAT debugging is on  
Router#no debug ip nat  
IP NAT debugging is off  
Router#
```

Imagen 31. Debug & undebug NAT

```
Router#  
Router#clear ip nat translation *  
Router#
```

Imagen 25. Eliminar todos los registros NAT



 [www.universae.com](http://www.universae.com)

