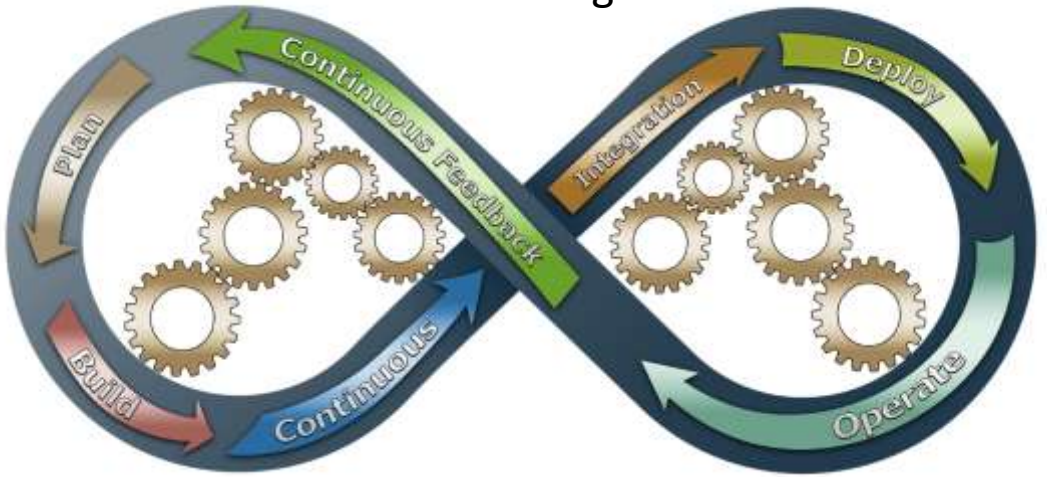
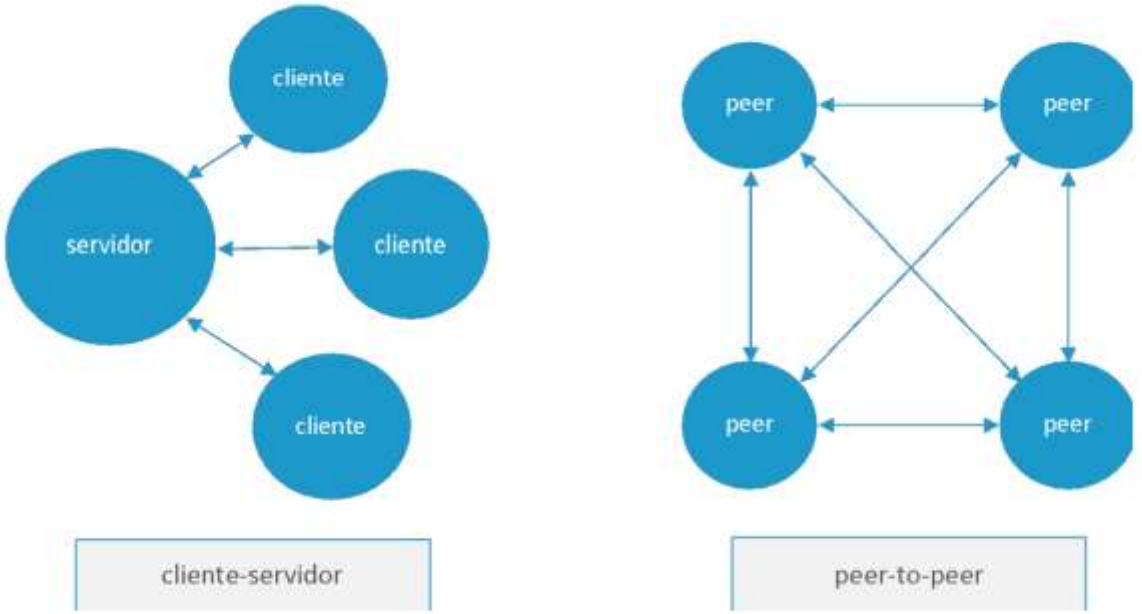
edix
Digital Workers

UF-4 REDES Y SEGURIDAD

Profesor Raúl Salgado Vilas



- Arquitectura de red: Este criterio se refiere al rol que cumplen los dispositivos que pertenecen a la red:
  - ✓ Puede haber uno o más sistemas que actúen como servidores. Un cliente solicita al servidor que atienda las solicitudes. El servidor toma y procesa la solicitud en nombre de los clientes.
  - ✓ Dos sistemas pueden estar conectados punto a punto y formar una red peer-to-peer. Si ambos residen en el mismo nivel, se les denomina pares.



Tipos de redes según su arquitectura.

- Arquitectura en capas En la arquitectura en capas, todo el proceso de intercambio de tráfico de red se divide en pequeñas tareas. Cada tarea se asigna a una capa determinada que funciona de manera dedicada para procesar esa tarea específica:

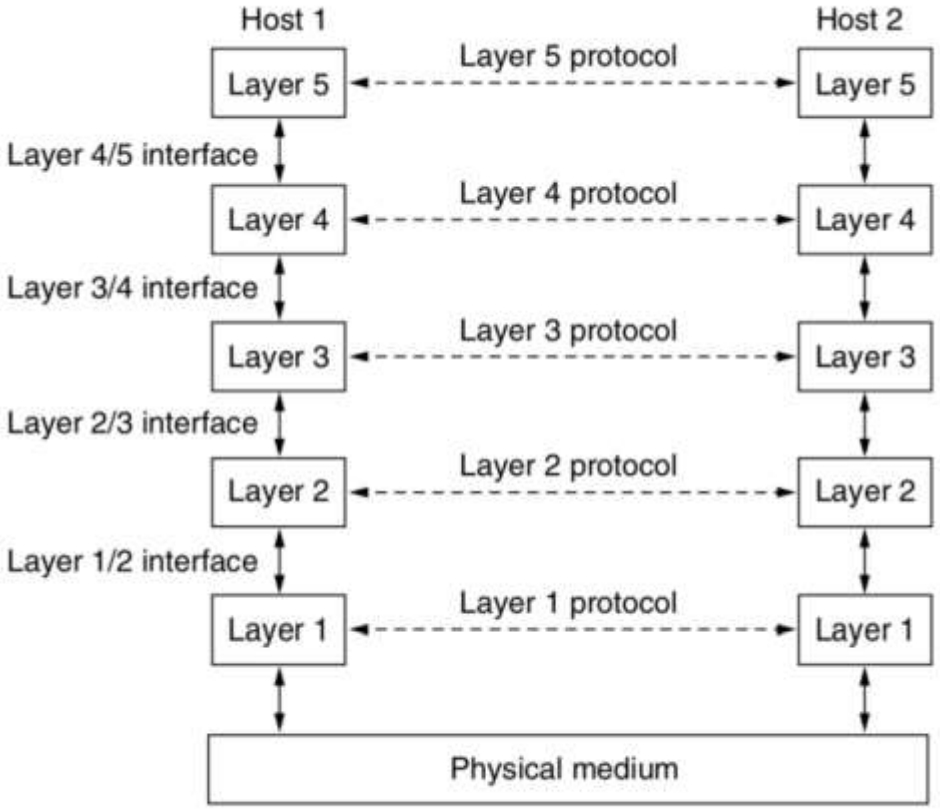


Imagen 1. Capas de red.

- Arquitectura en capas En la arquitectura en capas, todo el proceso de intercambio de tráfico de red se divide en pequeñas tareas. Cada tarea se asigna a una capa determinada que funciona de manera dedicada para procesar esa tarea específica:

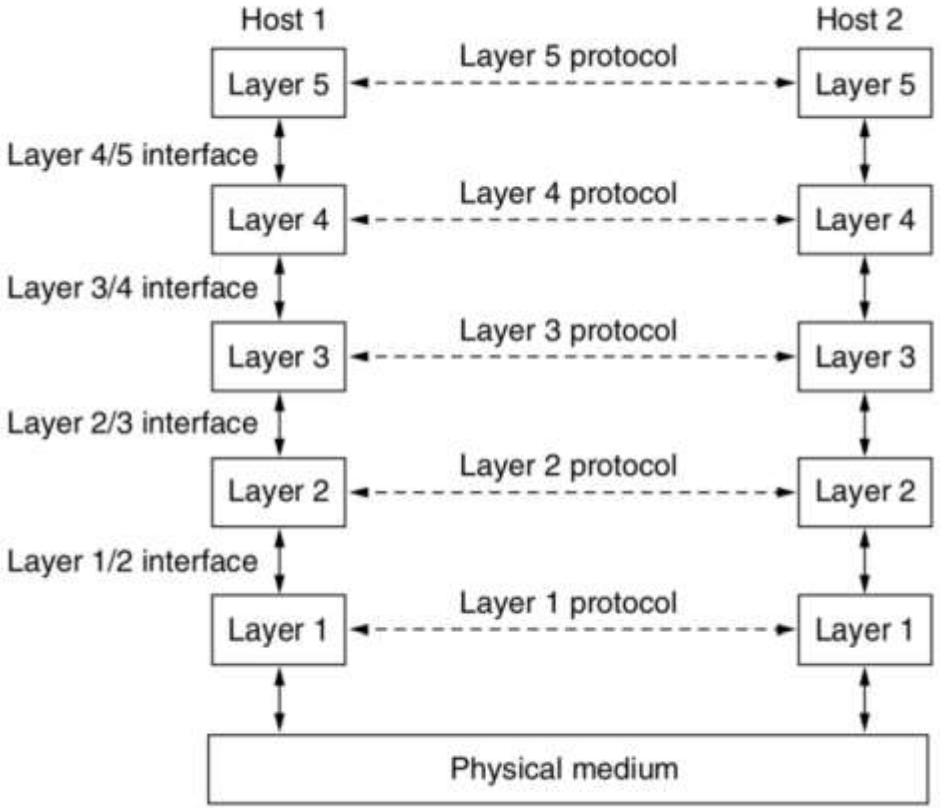


Imagen 1. Capas de red.

- Todas las capas identifican a sus peers (contrapartes, la capa equivalente en el host del otro extremo) encapsulando el contenido en tramas (frames) con una cabecera que incluye los parámetros específicos de esa capa. El diagrama de la Imagen 2 muestra cómo cada capa añade una cabecera, de forma que los datos más la cabecera de la capa N se convierten en los datos de la capa N-1:

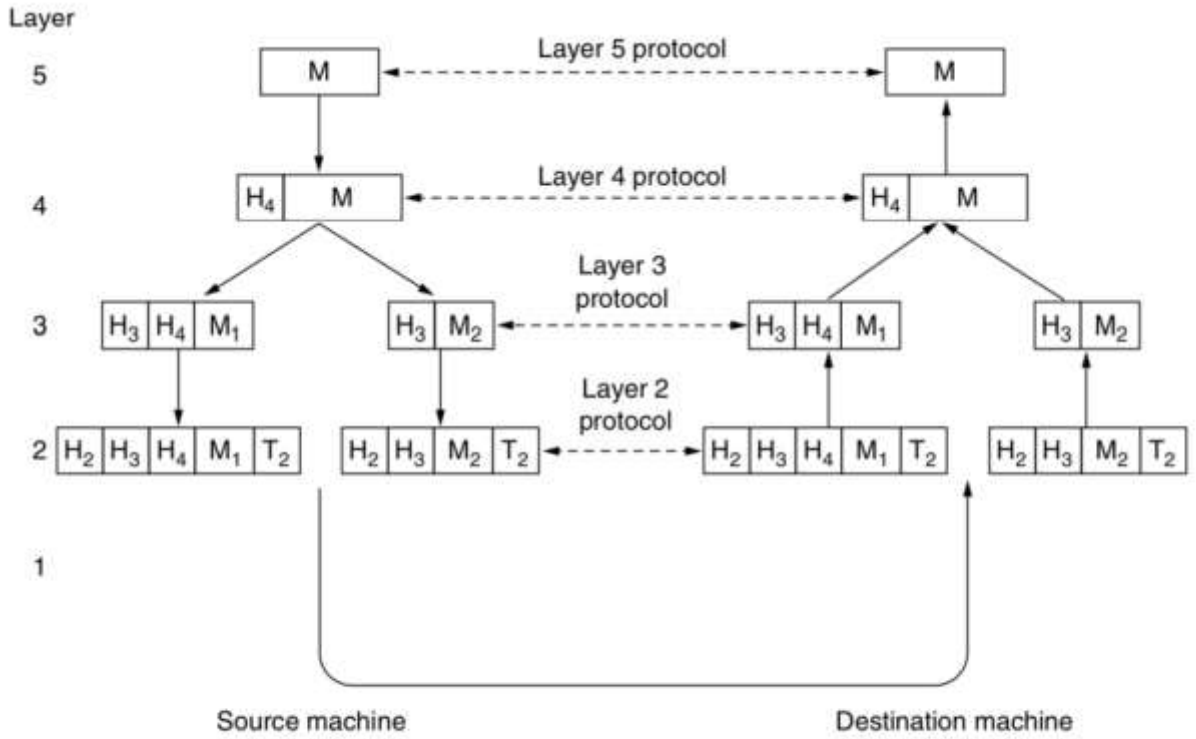


Imagen 2. Cabeceras añadidas en cada capa de red.

- De manera formal, cada capa ofrece un servicio a la capa superior. Por ejemplo, la capa N puede ofrecer garantías de entregas sin errores a la capa N+1. Las reglas para el funcionamiento interno de una capa definen un protocolo.

- Modelo OSI La ISO, International Organization for Standardization (en español, Organización Internacional de Estandarización) publicó en 1980 el estándar OSI, de Open Systems Interconnection. OSI es un estándar abierto para cualquier sistema de comunicación formado por siete capas detalladas en la siguiente imagen:

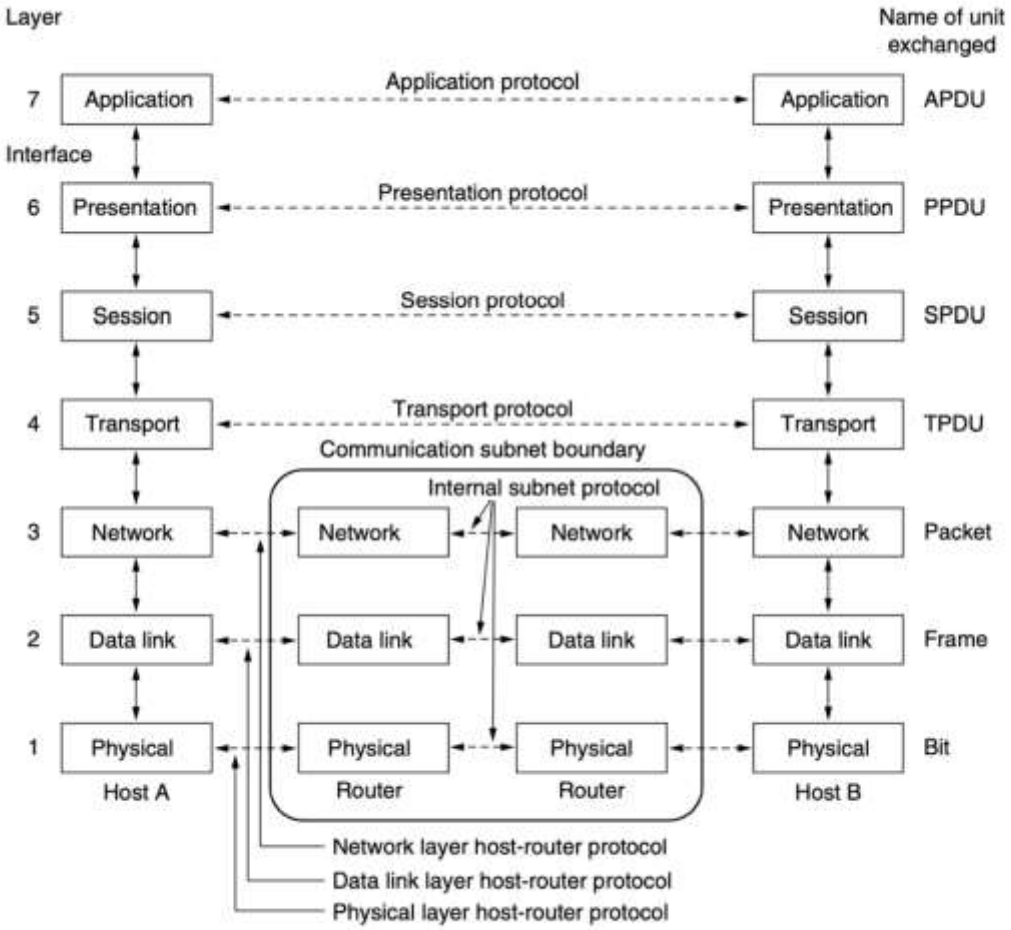
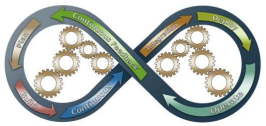


Imagen 3. Modelo de referencia OSI.



- Modelo TCP/IP Internet utiliza la suite de protocolos TCP/IP, por lo que este modelo se puede denominar también modelo de internet.
- los criterios que dieron pie a TCP/IP fueron:
  - ✓ Resistencia a la pérdida de elementos de red sin afectar al tráfico existente.
  - ✓ Arquitectura flexible capaz de soportar tanto transferencia de ficheros como tráfico en tiempo real. - Interconexión de redes heterogéneas: cableadas, radio, satélite…
  - ✓ Posibilidad de mantener una conexión, incluso en el caso de caída de nodos intermedios, entre la fuente y el destino
- Las capas definidas en este modelo son:
  - ✓ Enlace Este nivel no cuenta con mucha documentación incluso en el modelo original: su propósito principal está en definir qué deben hacer los enlaces, como conexiones Ethernet o conexiones serie, para poder usarse por la capa superior para establecer comunicaciones sin necesidad de una conexión permanente. Es poco más que el enlace entre el hardware y la capa que veremos ahora.
  - ✓ Internet Esta capa, cuyo nombre ha dado nombre a la red que todos conocemos, es la encargada de llevar paquetes de una red a otra, en cualquier orden y a través de cualquier medio. Utiliza para esto el protocolo IP (protocolo de internet) y se apoya en un protocolo de control llamado ICMP (Internet Control Message Protocol). Esta capa define el direccionamiento (cómo identificar a diferentes hosts en una red determinada), el enrutamiento (cómo llevar un paquete, salto a salto, hasta una red destino) y la gestión de errores a nivel de paquete.

- Transporte Esta capa se construye sobre el protocolo IP y ofrece dos protocolos principales:
  - ✓ TCP es un protocolo de transporte orientado a conexión que garantiza que dos entidades puedan compartir streams de bytes, que llegan de forma ordenada y sin congestionar al destino. –
  - ✓ UDP (User Datagram Protocol) es otro protocolo de transporte, sin conexión y sin garantías de entrega, lo que se denomina un protocolo best-effort. Su ventaja frente a IP es la mayor velocidad al no tener la compleja capa de gestión de conexión, control de flujo y entrega ordenada de paquetes. La gran desventaja es, sin duda, que no se puede confiar que un paquete llegue al destino: esto es muy útil en transmisión de vídeo en tiempo real, donde lo importante es que los paquetes lleguen a tiempo (si llegan tarde, ya no son necesarios).

- El protocolo IP El protocolo IP es el más ampliamente extendido en la capa de nivel 3. Actualmente hay dos versiones funcionando simultáneamente: IPv4, que ha gobernado el mundo de internet durante décadas, pero que se está quedando sin espacio de direcciones; e IPv6, que se creó para reemplazar IPv4 y se espera que también mitigue las limitaciones de IPv4:

172	.	16	.	254	.	1
10101100	.	00010000	.	11111110	.	00000001

255	.	255	.	0	.	0
11111111	.	11111111	.	00000000	.	00000000

172	.	16	.	0	.	0

- Por ejemplo, la dirección hexadecimal de 32 bits AC10FE01 se escribe como 172.16.254.1. Si además sabemos que pertenece a una red con un prefijo de 16 bits, el prefijo de red se indicaría como 172.16.0.0/16

- Subredes Las direcciones IP están divididas en tres categorías principales:
  - Clase A: utiliza el primer octeto para las direcciones de red y los últimos tres octetos para el direccionamiento del host.
  - Clase B: utiliza los primeros dos octetos para las direcciones de red y los dos últimos para el direccionamiento del host.
  - Clase C: utiliza los primeros tres octetos para las direcciones de red y el último para el direccionamiento del host.

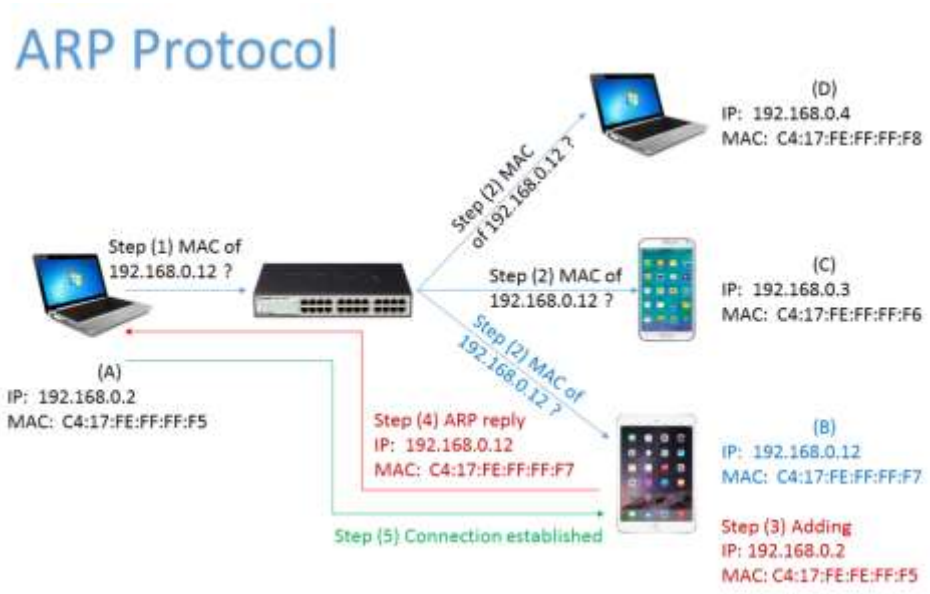
Clase	Tamaño prefijo	#redes	#direcciones	Ejemplo de rango
A	8	128	16.777.216	10.0.0.0 - 10.255.255.255
B	16	16384	65.536	192.168.1.0 - 192.168.1.255
C	24	2097152	256	10.5.1.0 - 10.5.1.255

- IPv4 también tiene espacios de direcciones bien definidos para ser utilizados como direcciones privadas (no enrutables en internet) y direcciones públicas (proporcionadas por los ISP y enrutables en internet):

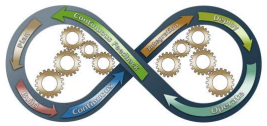
CIDR	Rango de direcciones	# direcciones	Descripción
10.0.0.0/8	10.0.0.0 – 10.255.255.255	16.777.216	Una única red de clase A.
172.16.0.0/12	172.16.0.0 – 172.31.255.255	1.048.576	16 redes de clase B
192.168.0.0/16	192.168.0.0 – 192.168.255.255	65.536	256 redes de clase C

Tabla 2. Rangos de redes privadas.

- ARP o Address Resolution Protocol: Durante la comunicación, un host necesita la dirección de capa 2, conocida como MAC, del siguiente salto del paquete. Este salto puede ser el host de destino, si la comunicación no sale de la propia subred, o el gateway que el host ha decidido usar para salir de su subred si el destino está en una red remota. La dirección MAC está físicamente grabada en la tarjeta de interfaz de red de un equipo y nunca cambia. En un entorno virtualizado, una NIC virtual recibe una MAC del hipervisor. Esto puede provocar que haya MACs duplicadas, ya que la asignación es aleatoria, pero los hipervisores tienen mecanismos para evitar que las MACs se repitan en una misma red virtual. En cualquier caso, la NIC virtual de una VM no suele cambiar a lo largo de su vida.



- Para conocer la dirección MAC del host remoto en una subred, el ordenador de origen envía un mensaje broadcast ARP preguntando qué host tiene dirección de IP. Debido a que es un mensaje broadcast, todos los hosts del segmento de red, que equivale a un dominio de broadcast, reciben este paquete y lo procesan. El paquete ARP contiene la dirección IP del host de destino al que el host de origen quiere contactar. Cuando un host recibe un paquete ARP que se le ha destinado, responde con su propia dirección MAC. Una vez que el host recibe la dirección MAC de destino, puede comunicarse con el host remoto utilizando el protocolo de enlace. Este mapeo del MAC al IP se guarda en una caché de ambos hosts. Si en el futuro necesitan comunicarse, pueden directamente dirigirse a su caché ARP.

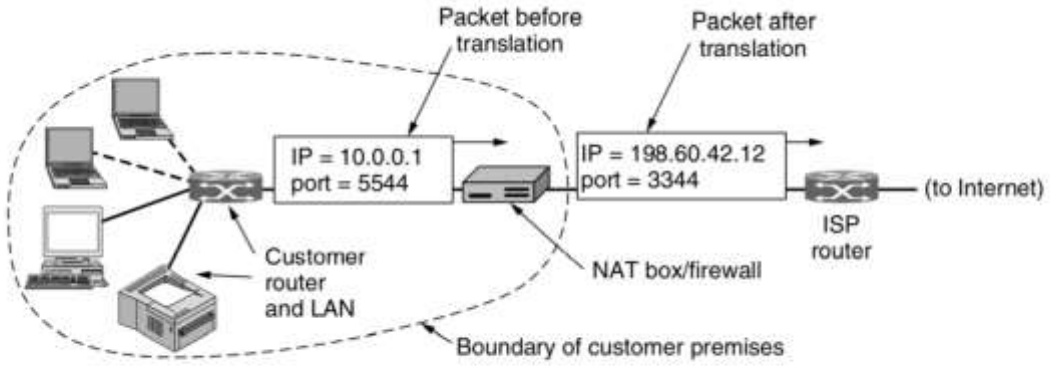


- Internet Control Message Protocol (ICMP): ICMP es un protocolo de diagnóstico y notificación de errores de red. Pertenece al protocolo IP y utiliza IP como protocolo de encapsulamiento. Después de construir el paquete ICMP, se encapsula en un paquete IP. Los hosts usan ICMP para informar si se produce algún error en la red. Este ofrece opciones para informar de que contiene docenas de mensajes de diagnóstico y reporte de errores. Los mensajes de ICMP-echo y ICMP-echo-reply son los mensajes ICMP más comúnmente utilizados para comprobar la accesibilidad de los hosts de extremo a extremo, utilidad expuesta en la mayoría de los sistemas operativos en la utilidad ping:

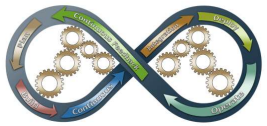
```
C:\Users\rauls>ping  www.edix.com
Haciendo ping a edix.com [213.149.230.164] con 32 bytes de datos:
Respuesta desde 213.149.230.164: bytes=32 tiempo=110ms TTL=56
Respuesta desde 213.149.230.164: bytes=32 tiempo=24ms TTL=56
Respuesta desde 213.149.230.164: bytes=32 tiempo=6ms TTL=56
Respuesta desde 213.149.230.164: bytes=32 tiempo=7ms TTL=56

Estadísticas de ping para 213.149.230.164:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 6ms, Máximo = 110ms, Media = 36ms
```

- NAT Las direcciones IPv4 son escasas. Si un ISP dispone de un bloque /16, dándole 65.534 direcciones, el número de clientes que puede tener está limitado a ese número. Un mecanismo para resolver este problema es NAT (Network Address Translation, o traducción de direcciones de red). La idea básica detrás de NAT es que un ISP asignará a cada cliente una única dirección IP pública para navegar por la red y, dentro de la subred interna del cliente, cada equipo obtiene una dirección IP única que se utiliza para enrutar el tráfico local:



- Ahora viene el problema: cuando el paquete de respuesta vuelve al origen (por ejemplo, desde un servidor web), se dirige naturalmente a la dirección 198.60.42.12. Entonces ¿cómo sabe el dispositivo NAT con qué dirección interna reemplazarla? Las cabeceras IP no soportan NAT nativamente, por lo que la dirección interna no se puede incluir como parte del paquete. La solución en NAT es hacer uso de los puertos de origen y destino (el capítulo siguiente aclarará las dudas sobre los puertos TCP y UDP).



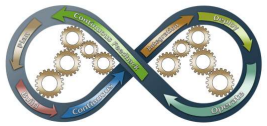
- Protocolo TELNET: TELNET (TELecommunications NETwork) es un protocolo de comunicación, genérico y bidireccional, con envío de información sin cabeceras extra ni cifrado de ningún tipo desde el cliente al servidor. Su uso habitual es para establecer una sesión de terminal virtual de red (NVT) a través de una red, de modo que se puedan enviar comandos al puerto destino donde el servidor los interpretará como sea necesario.

```
C:\Users\rauls>telnet loalhost 1521
```

- TELNET no es seguro: envía toda su información en claro (sin cifrado de ningún tipo), por lo que cualquier analizador de tráfico puede interceptar lo que se envía. Por ello, el protocolo de inicio de sesión remoto por excelencia es SSH (Secure shell), donde se garantiza la confidencialidad de los datos mediante técnicas criptográficas

- Protocolo SSH Es el protocolo por defecto para las conexiones seguras a sistemas remotos para tareas de administración y gestión mediante NVTs. Está encaminado a reemplazar a telnet y otras aplicaciones de login/copia de ficheros no seguras, como rcp. Por defecto, SSH escucha el puerto 22. Por estas razones, se puede usar ssh para crear un túnel seguro entre dos hosts y comunicarse mediante otros protocolos a través del mismo, ya de forma segura: el siguiente comando crea un túnel ssh en segundo plano, que conecta el puerto local 2020 con el puerto 8000 de la máquina remota:

```
$ ssh -l vagrant 192.168.33.10 -NfL 2020:127.0.0.1:8000
```



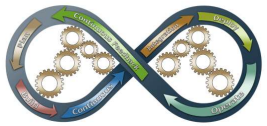
- Protocolo TELNET: TELNET (TELecommunications NETwork) es un protocolo de comunicación, genérico y bidireccional, con envío de información sin cabeceras extra ni cifrado de ningún tipo desde el cliente al servidor. Su uso habitual es para establecer una sesión de terminal virtual de red (NVT) a través de una red, de modo que se puedan enviar comandos al puerto destino donde el servidor los interpretará como sea necesario.

```
C:\Users\rauls>telnet loalhost 1521
```

- TELNET no es seguro: envía toda su información en claro (sin cifrado de ningún tipo), por lo que cualquier analizador de tráfico puede interceptar lo que se envía. Por ello, el protocolo de inicio de sesión remoto por excelencia es SSH (Secure shell), donde se garantiza la confidencialidad de los datos mediante técnicas criptográficas

- Protocolo SSH Es el protocolo por defecto para las conexiones seguras a sistemas remotos para tareas de administración y gestión mediante NVTs. Está encaminado a reemplazar a telnet y otras aplicaciones de login/copia de ficheros no seguras, como rcp. Por defecto, SSH escucha el puerto 22. Por estas razones, se puede usar ssh para crear un túnel seguro entre dos hosts y comunicarse mediante otros protocolos a través del mismo, ya de forma segura: el siguiente comando crea un túnel ssh en segundo plano, que conecta el puerto local 2020 con el puerto 8000 de la máquina remota:

```
$ ssh -l vagrant 192.168.33.10 -NfL 2020:127.0.0.1:8000
```



- Protocolo FTP File Transfer Protocol (FTP): Es el protocolo más utilizado para la transferencia de archivos en la red, o al menos lo era antes de la popularización de las redes de intercambio peer-to-peer.

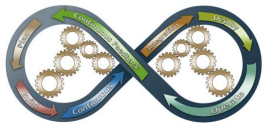
FTP utiliza TCP como capa de transporte y utiliza diferentes puertos en función del modo de funcionamiento:

- El cliente siempre inicia una conexión de control al puerto TCP 21.
- En modo activo, el cliente abre un socket en un puerto aleatorio e indica al servidor que inicie la transferencia de datos en ese puerto.
- En modo pasivo, es el servidor el que abre un socket adicional en un puerto aleatorio y le indica al cliente que lo use para sus transferencias.
- El modo activo no funcionará en situaciones con firewalls o en las que el cliente está detrás de un NAT, ya que el puerto en el cliente no será accesible desde la red del servidor

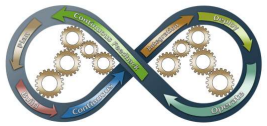
- Protocolo SMTP: El Protocolo de transferencia de correo simple (SMTP) se utiliza para el intercambio de mensajes de correo electrónico entre un cliente y un servidor o entre servidores. En un cliente de escritorio, un agente de la aplicación se encarga de gestionar el envío al servidor mediante SMTP. Mientras que el usuario final utiliza SMTP solo para enviar los correos electrónicos, los servidores normalmente utilizan SMTP tanto para enviar como para recibir correos. La recepción de correos en los clientes se suele llevar a cabo con POP o IMAP, como se verá en un capítulo posterior. Los clientes usan el puerto TCP 587 como destino para el envío de mensajes, mientras que los servidores usan el puerto TCP 25 tanto para el envío como para la recepción.

- Protocolo DNS El protocolo de nombres de dominio (Domain Name System, o DNS) facilita el mapeo entre nombres de equipos y direcciones IP. Los programas de software pueden trabajar con direcciones IP sin problema, pero para usuarios finales es más fácil recordar un nombre de equipo como ftp.rediris.es que su IP 130.206.13.2.

Registros DNS	
A	Dirección de un host
NS	Nombre de servidor DNS acreditado para esta zona
CNAME	Alias de un nombre de host
MX	Nombre de servidor de correo asociado al dominio
TXT	Cadena de texto



- El protocolo trabaja sobre UDP usando el puerto 53 por defecto. La elección de UDP se debe principalmente a razones de rendimiento. El número de peticiones DNS necesarias para resolver un único nombre es elevado y el contenido de cada petición es relativamente pequeño, así que el tamaño de las cabeceras TCP es comparable y añade una sobrecarga importante. Además, las peticiones DNS son una herramienta auxiliar antes de iniciar el protocolo deseado, por lo que es deseable una reducción en la latencia inicial. Aunque formalmente DNS no es confiable, no hay más que usar internet para comprobar que funciona como se espera. Aunque las peticiones de resolución de nombres usan UDP como protocolo de transporte, los servidores DNS se comunican entre ellos mediante TCP en el puerto 53. Esta comunicación se usa para el intercambio y sincronización de zonas DNS. Este mecanismo permite que una actualización de un registro DNS se pueda replicar a otros servidores sin intervención manual.
- Protocolo HTTP El protocolo quizá más conocido a este nivel, pues es el que nos permite navegar por diferentes páginas de internet, abriendo sus recursos en diferentes navegadores web o usando aplicaciones específicas para móvil que ofrecen servicios personalizados por este canal.

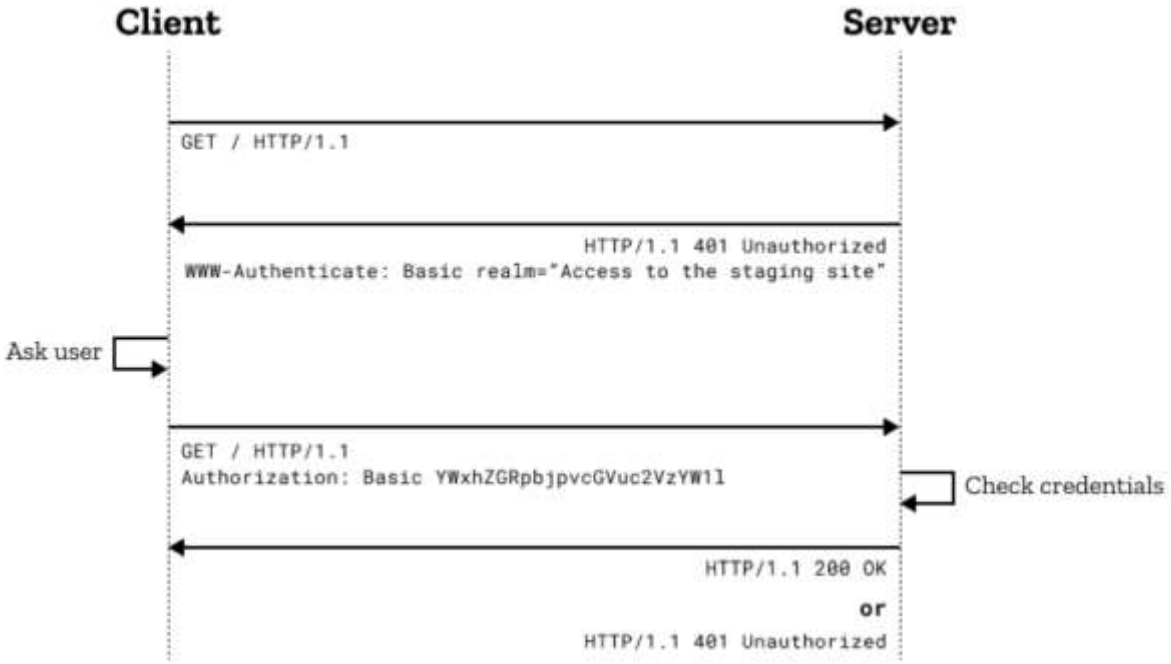


```
protocolo://[user:pass@]host[:puerto]/ruta/al/recurso[?param1=value1[&param2=value2…]][#fragmento]
```

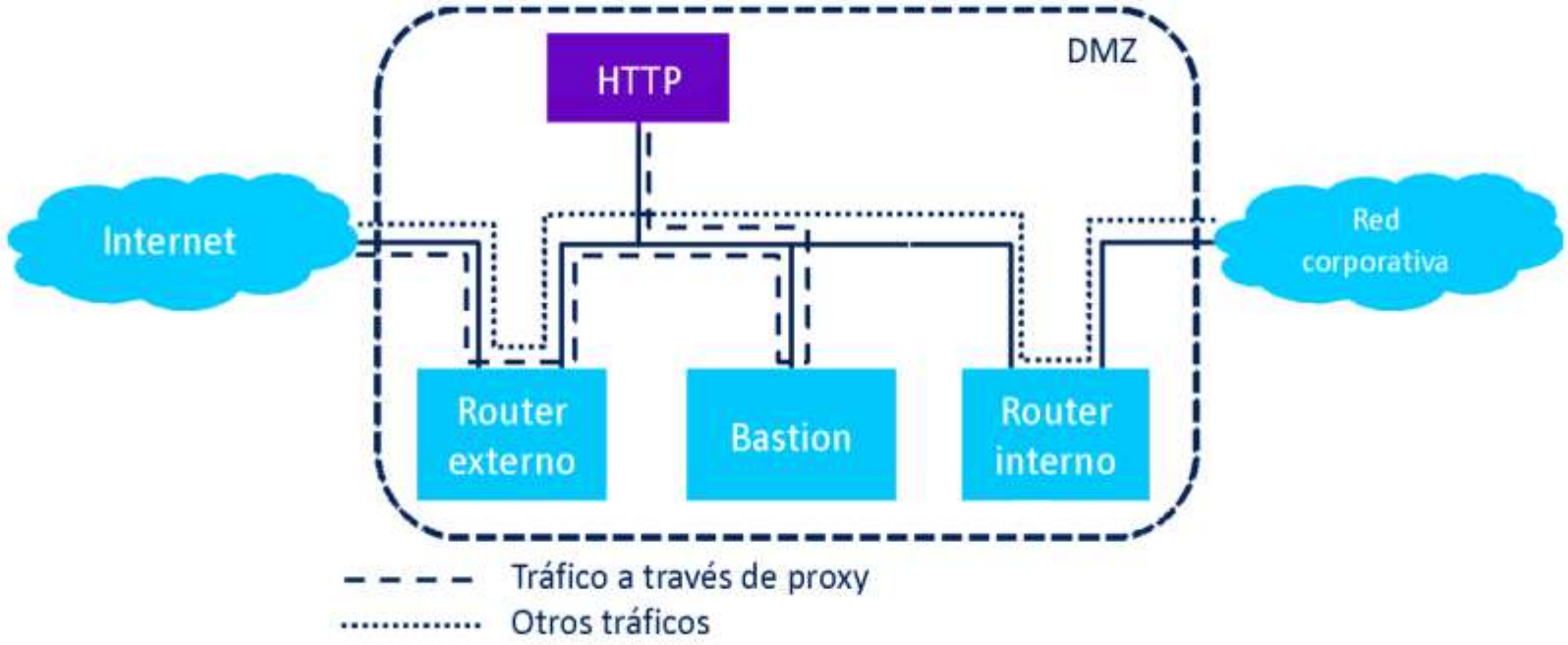
- Protocolo: define el protocolo de aplicación a usar para solicitar el recurso. Valores usuales son http, https, ftp, mailto… También pueden ser aplicaciones específicas, como chrome://, spotify://… Estas URLs suelen hacer que el recurso se maneje de una manera determinada al abrirlo con determinados dispositivos (como, por ejemplo, abrir Spotify en el móvil).
- User/pass: en caso de que el servicio necesite autenticación, se puede enviar directamente en la URL (ojo, no es seguro).
- Host: el nombre canónico del servidor donde se aloja, cuya IP se consultará usando DNS.
- Puerto: en caso de que no se use el puerto por defecto para el protocolo elegido, se especifica aquí dónde se escucha.
- /ruta/al/recurso: el path que usará el servidor para ubicar el recurso.
- Parámetros extra: se pueden añadir parámetros extra a partir del símbolo ?, separados por ampersands (&). Estos parámetros serán empleados por el servicio para lo que decida, no forman parte de ningún protocolo.
- Fragmento: Si se especifica, permite referenciar tan solo una parte del recurso. Por ejemplo, nos puede llevar a una parte concreta de una página web o a un instante concreto de un vídeo

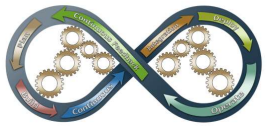
- HTTP Secure (HTTPS): Es básicamente una versión segura de HTTP que corre sobre TLS (Transport Layer Security) para que las comunicaciones estén apropiadamente cifradas. En esta sección no entraremos en detalles relativos a la seguridad, pero podemos adelantar que TLS proporciona los mecanismos de seguridad necesarios para que las comunicaciones sean ya confiables. El puerto por defecto para HTTPS es el 443, y el esquema de las URLs cambia cuando se usa, comenzando estas por https://.

$curl https://admin:1234@88.45.12.61:3030/api/v1/users



- Cortafuegos de subred o DMZ: El cortafuegos de subred se puede considerar una extensión del cortafuegos de host de rastreo. También incorpora un router de rastreo, denominado externo, y un host bastión. Sin embargo, este cortafuegos crea una capa adicional de seguridad añadiendo una red de perímetro que aísla a la red privada de internet. Esta capa define una DMZ (de demilitarized zone o zona desmilitarizada) demarcada por el router externo y un router interno. Este último está localizado más cerca de la red privada que del enrutador externo. El bastión y los servidores de acceso público se encuentran entonces dentro de la DMZ y es la **arquitectu**





- Protocolos de tunneling Hay cuatro protocolos de tunelización utilizados para establecer VPNs. Pueden clasificarse de forma general en dos grupos: PPTP, L2F y L2TP son protocolos de tunelización de capa 2, mientras que IPSec es un protocolo de tunelización de capa 3.

- Criptografía simétrica: Esta familia de técnicas, también conocida como criptografía convencional, usa la misma clave para cifrar y para descifrar los mensajes. Por tanto, requiere que ambos extremos de la comunicación (es decir, los usuarios o los agentes de software) compartan la clave. Estas técnicas tienen un punto débil en el intercambio de la clave: necesitan un canal seguro para compartirla. Si se ha podido compartir en secreto (por un segundo canal confiable o en persona), el algoritmo puede ser tan seguro como permita técnicamente. Es decir, el problema del intercambio de la clave no es una propiedad de seguridad de cada algoritmo simétrico, sino un problema de esta familia en general. La siguiente familia de algoritmos viene a dar una solución al problema.

- Criptografía asimétrica: Estos algoritmos también se conocen como de clave pública. Resuelven el problema del intercambio de claves definiendo un algoritmo que usa dos claves (conocidas como key pair o pareja de claves), cada una de las cuales puede usarse para cifrar un mensaje. Cuando se cifra un mensaje con una de las claves hay que usar la otra para descifrarlo. Si el usuario A quiere que solo el usuario B lea el contenido del mensaje, cifrará el mensaje con la clave pública de B. Es decir, B siempre compartirá la misma clave, considerada pública, con cualquier usuario de quien necesite recibir mensajes. Deberá mantener la clave privada en secreto.

❑ Pila Protocolos SSH: