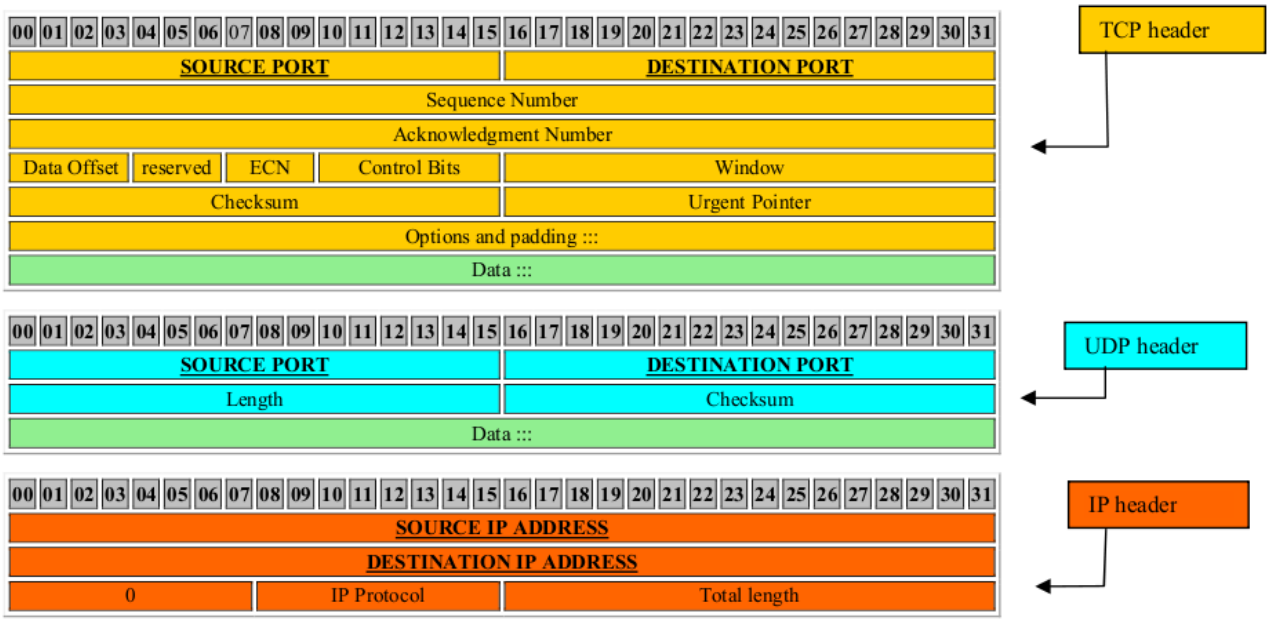
Introducción teórica a NAT:

NAT (Network Address Translation), o sea, traducción de direcciones de red, permite cambiar las **direcciones** y los **puertos** de los **paquetes IP** y los **paquetes TCP ó UDP** que un paquete IP encapsula.

A continuación se muestran dos paquetes completos con su **cabecera MAC**, que **encapsula** una **cabecera IP** que posteriormente **encapsula** una **cabecera TCP o UDP** que contiene dentro datos de aplicación. A continuación se muestra el detalle de una cabecera TCP, una UDP y una IP:



Los campos que aparecen **SUBRAYADOS Y EN NEGRITA Y MAYÚSCULAS** son los que NAT puede transformar, es decir:

- **En un paquete IP**: la dirección de destino y la dirección de origen
- **En un paquete TCP ó UDP**: el puerto de origen y el puerto de destino.

Existen dos tipos básicos de NAT:

- **DNAT** (Destination NAT): que transforma la dirección o puerto de destino.
- **SNAT** (Source NAT): que transforma la dirección o puerto de origen.

Usos de SNAT:

- **Masquerading** (Enmascaramiento). Se usa para que los equipos de una LAN que utilizan direcciones privadas puedan acceder a INTERNET con una única dirección IP pública para toda la LAN.

Usos de DNAT:

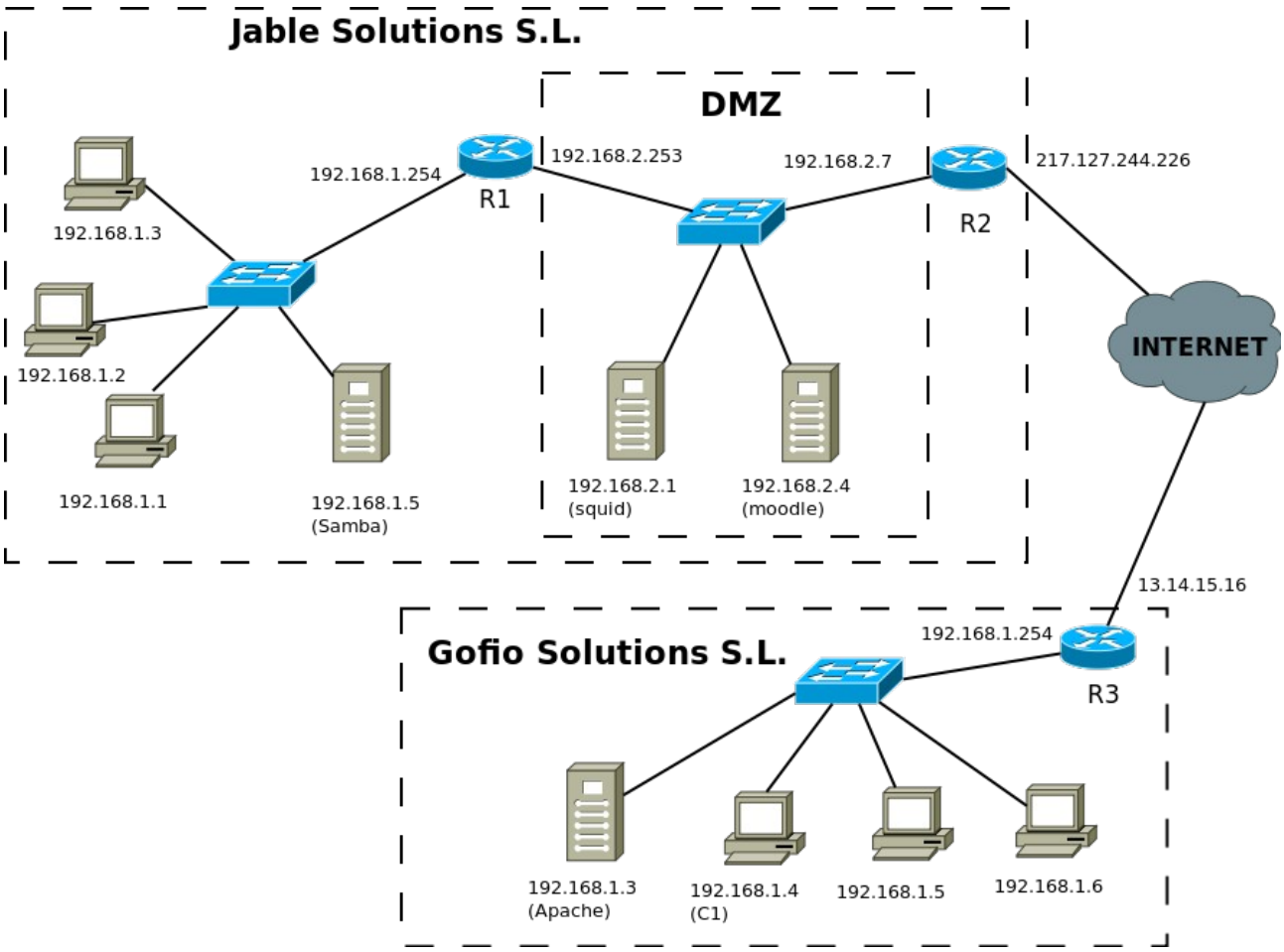
- **Port Forwarding** (Redirección de puertos). Permite que dentro de una LAN existan equipos que proporcionen servicios de INTERNET a equipos de INTERNETaunque dentro de la LAN se utilicen direcciones IP privadas.
- **Balanceo de carga**. Si se da un rango de direcciones IP, la dirección a usar se elegirá basándose en la menos utilizada para las conexiones de las que sabe

nuestra máquina. Esto nos permite hacer un balanceo de carga primitivo.

- **Proxy transparente**. Consiste en **redirigir las peticiones web** al Puerto y direcciónpor el que escucha un **Proxy** en la LAN de manera que no haya que configurar en los clientes el navegador para que acceda a través del Proxy.

Ejemplo detallado en el que se emplea Enmascaramiento y Redirección de puertos:

El ejemplo se explicará a partir del siguiente esquema de red:



Supongamos que el **equipo 192.168.1.1** del Jable Solutions S.L. quiere acceder al **servidor Apache** que se encuentra en la dirección 192.168.1.3 de GOFIO SOLUTIONS S.L. Veremos de forma detallada los cambios que van sufriendo el paquete de **ida** con la **petición** de una página web y el paquete de **vuelta** con la **página web pedida**.

Se trata de un ejemplo para entender el enmascaramiento y la redirección de puertos y no tendremos en cuenta su implementación que realizaremos posteriormente en LINUX.

Pasos seguidos por el paquete de ida:

1. Un usuario en el equipo 192.168.1.1 en el Jable Solutions S.L. abre el navegador y teclea la URL de la página web a la que quiere acceder. Esta URL será **http://13.14.15.16**, es decir, la página de inicio del servidor web de GOFIO SOLUTIONS S.L.

Observa que no se puede indicar la dirección 192.168.1.3 puesto que es la dirección IP privada dentro de la LAN de GOFIO SOLUTIONS S.L.

2. El sistema operativo del equipo 192.168.1.1 selecciona un puerto TCP libre para la conexión. Supongamos que es el 40124.

3. El siguiente paquete sale por la tarjeta de red del equipo 192.168.1.1, en el que solamente se indican los campos relevantes para esta explicación, pero se debe tener en cuenta que realmente van encapsulados en la estructura indicada anteriormente:

Dirección origen	Puerto origen	Dirección destino	Puerto destino
192.168.1.1	40124	13.14.15.16	80

4. Se supone que las tablas de encaminamiento del propio equipo 192.168.1.1 y del router R1 están configuradas para que el paquete llegue al router R2.

5. Una vez que el paquete llega a R2 éste se transforma por medio de NAT para que contenga únicamente direcciones IP públicas y así pueda ser enrutable en INTERNET:

Dirección origen	Puerto origen	Dirección destino	Puerto destino
217.127.244.226	15328	13.14.15.16	80

Para ello el software o hardware del router (NAT) ha realizado lo siguiente:

- **Transformar la dirección origen** inicialmente privada en la dirección **IP pública** del Jable Solutions S.L., de manera que cuando el servidor APACHE de GOFIO SOLUTIONS S.L. devuelva la página web pedida ésta pueda **llegar** hasta el router de Jable Solutions.
- Buscar un **puerto TCP libre** en el propio router R2, es decir, el **15328** en nuestro caso.
- Crear una entrada en una **tabla de conexiones abiertas** en el propio router cuya funcionalidad es recordar qué equipo de la LAN local es el que ha iniciado la conexión:

Dirección origen	Puerto origen	Dirección destino	Puerto destino	Puerto creado router
192.168.1.1	40124	13.14.15.16	80	15328

6. Las **tablas de encaminamiento de INTERNET** hacen que el paquete llegue al router R3 de GOFIO SOLUTIONS S.L.

7. En el router R3 de GOFIO SOLUTIONS S.L. se consulta la **tabla de redirección de puertos** (Port Forwarding) que debe tener una entrada como la siguiente:

Puerto origen	Dirección Destino	Puerto destino
80	192.168.1.3	8080

8. El paquete que llega a R3 se transforma entonces en el siguiente:

Dirección origen	Puerto origen	Dirección destino	Puerto destino
217.127.244.226	15328	192.168.1.3	8080

Para ello el software o hardware del router ha realizado lo siguiente:

- **Transformar** la **dirección destino**, que inicialmente era la dirección IP públicade GOFIO SOLUTIONS S.L., en la dirección **IP privada del servidor Apache**en la LAN. Esto se ha hecho teniendo en cuenta la **tabla de redireccionamiento** de puertos del router R3.
- También se ha **transformado el puerto** pues así lo indicaba la tabla de redireccionamiento de puertos del router R3. El servidor **Apache** 192.168.1.3 se supone por tanto que está escuchando por el **puerto 8080**.
- Crear una entrada en una **tabla de conexiones abiertas** en el propio router cuya funcionalidad es recordar qué paquete de INTERNET había entrado en la LAN local:

Dirección origen	Puerto origen	Dirección destino	Puerto destino	Puerto creado router
217.127.244.226	15328	13.14.15.16	80	8080

9. El paquete llega finalmente al servidor Apache 192.168.1.3 de GOFIO SOLUTIONS S.L. que escucha por el puerto 8080.

Pasos seguidos por el paquete de vuelta:

10. El servidor Apache 192.168.1.3 que escucha por el puerto 8080 en la LAN de GOFIO SOLUTIONS S.L. genera un paquete de respuesta con la página web pedida.

Dirección origen	Puerto origen	Dirección destino	Puerto destino
192.168.1.3	8080	217.127.244.226	15328

11. El paquete de respuesta llega al router R3 que es transformado por el software o hardware NAT del mismo, para que pueda llegar a su destino a través de INTERNET.

Dirección origen	Puerto origen	Dirección destino	Puerto destino
13.14.15.16	80	217.127.244.226	15328

Esta transformación ha sido posible porque el router R3 había **guardado la transformación NAT** realizada anteriormente en el **paso 8**. Es decir, se buscó una transformación de un paquete que **venía de 217.127.244.226** por el **puerto 15328** y que fue a parar al **puerto 8080**. El resultado es que esa conexión se realizó al router por el **puerto 80** originalmente, por lo que en el paquete de respuesta se cambia la dirección de origen a **13.14.15.16** y el **puerto de origen al 80**.

12. El paquete de respuesta llega al router R2 que es transformado por el software o hardware NAT del mismo, para que pueda llegar a su destino LAN correspondiente.

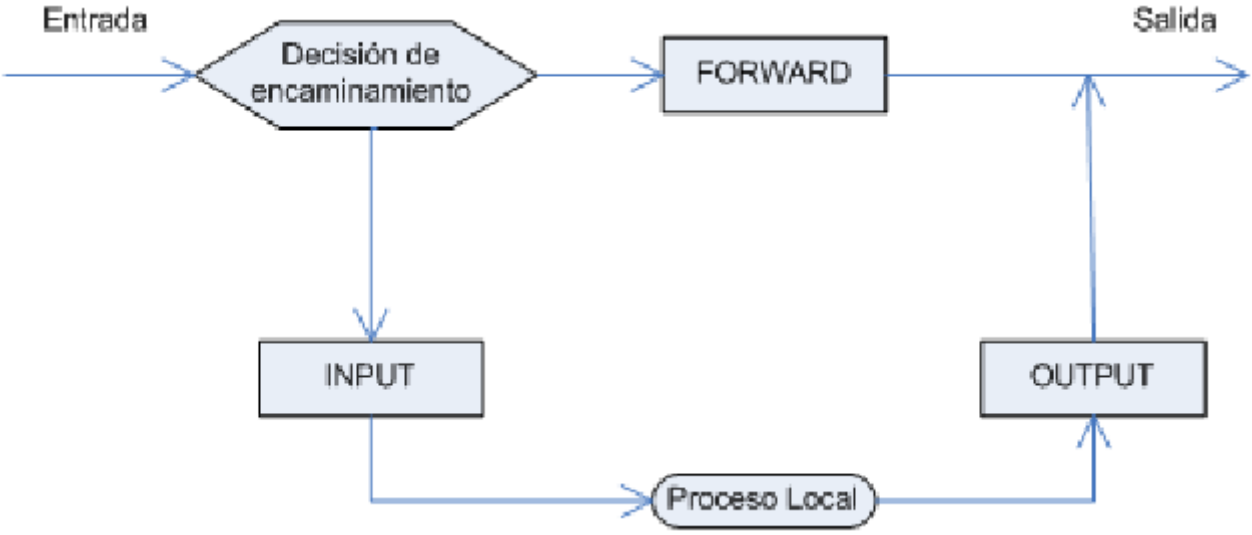
Dirección origen	Puerto origen	Dirección destino	Puerto destino
13.14.15.16	80	192.168.1.1	40124

Esta transformación ha sido posible porque el router R2 había **guardado la**

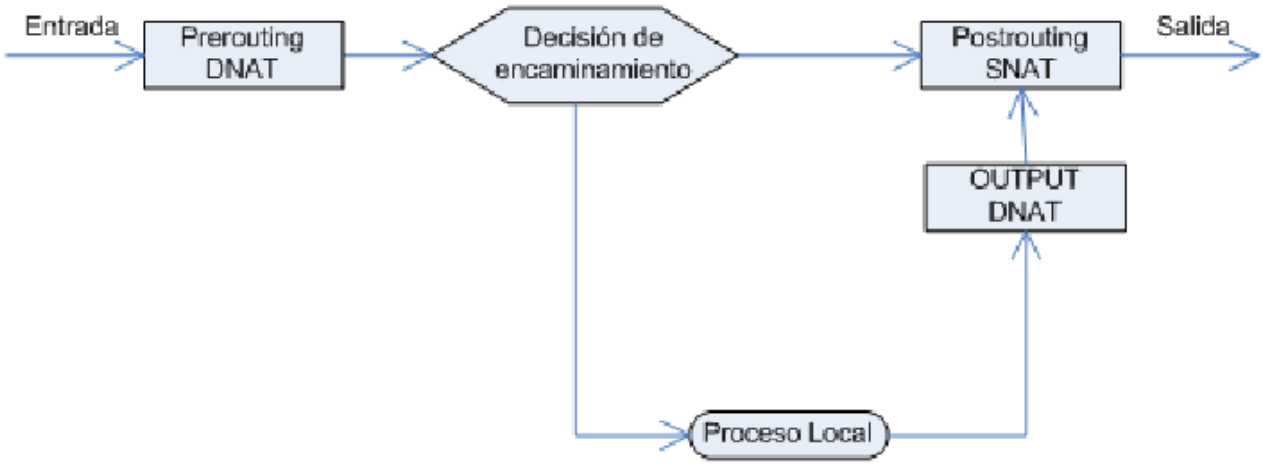
transformación NAT realizada anteriormente en el **paso 5**. Es decir, se buscó una transformación de un paquete que iba hacia **13.14.15.16** por el **puerto 80** y cuyo puerto de origen fue transformado a **15328**. El resultado es que esa conexión provenía del equipo **192.168.1.1** por el puerto **40124** originalmente, por lo que en el paquete de respuesta se cambia la **dirección de destino** a 192.168.1.1 y el **puerto de destino** al 40124.

13. El paquete de respuesta llega finalmente al equipo 192.168.1.1 de la LAN de Jable Solutions S.L. y el navegador muestra la página web.

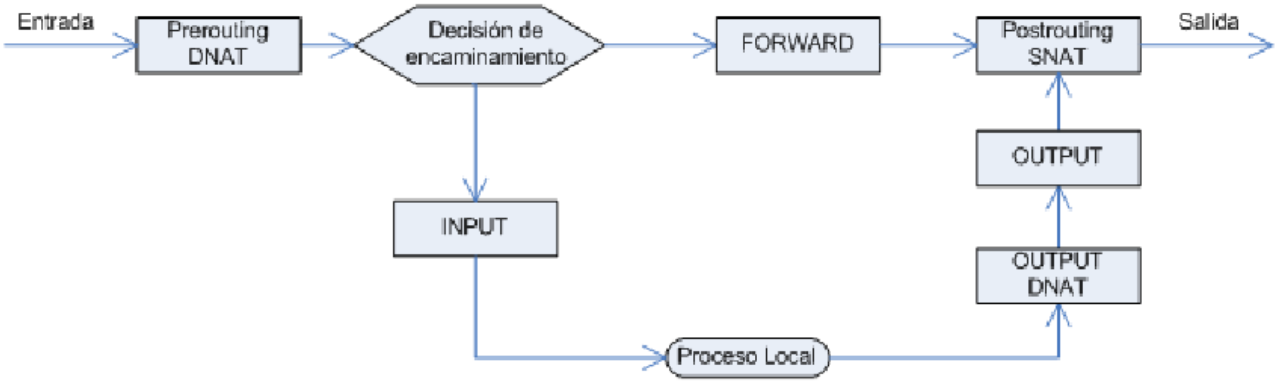
Funcionamiento del filtrado de paquetes en LINUX con iptables:



Funcionamiento de NAT en LINUX con iptables:



Funcionamiento conjunto de filter y NAT en LINUX con iptables:



Como se puede observar por el esquema anterior:

- Las traducciones de direcciones de origen se realizan **justo antes de salir** del equipo en la **cadena POSTROUTING** de la **tabla NAT**.
- Las traducciones de direcciones de destino para paquetes que entran por una interfaz de red se realizan incluso antes de la toma de decisión de encaminamiento en la **cadena PREROUTING** de la **tabla NAT**.
- Las **traducciones de direcciones** de destino para los **paquetes generados por el propio equipo** se realizan justo después de generarse en la **cadena OUTPUT** de la **tabla NAT**.

Trabajar con iptables puede llegar a ser realmente complejo, aunque también muy útil como veremos más adelante.