

# Guión de laboratorio – Desarrollo de un cliente ARP

## Objetivo

Aprender a crear, enviar y recibir mensajes ARP mediante la librería `librawnet` y las implementaciones base de los protocolos Ethernet (`eth_base.tgz`) y direcciones IPv4 (`ipv4_base.tgz`) proporcionados. Desarrollar un módulo que implemente un cliente del *Address Resolution Protocol* (ARP) [1], que pueda ser utilizado en futuros proyectos.

## Especificaciones

Se pide desarrollar un programa (`arp_client.c`) basado en la librería `librawnet` que muestre en pantalla la dirección MAC asociada a la dirección IPv4 que se pasa como parámetro en su línea de comandos, junto con el interfaz que se desea emplear.

Para ello debe implementar la función “`int arp_resolve (eth_iface_t * iface, ipv4_addr_t ip_addr, mac_addr_t mac_addr)`” que, dada la dirección IPv4 “`ip_addr`”, envíe una petición ARP por el interfaz Ethernet especificado (“`iface`”) y rellene la dirección “`mac_addr`” con la respuesta obtenida, o devuelva un valor de error distinto de 0 si la respuesta no ha llegado después de 2 segundos. Como todavía no se dispone de una dirección IPv4 configurada, la dirección de red origen del mensaje ARP debe ser 0.0.0.0.

El funcionamiento correcto del cliente desarrollado se probará utilizando el servidor ARP implementado por la pila de protocolos TCP/IP estándar de los ordenadores del laboratorio.

Así, el resultado de ejecutar “`arp_client eth0 163.117.144.241`” debería ser:

```
163.117.144.241 -> 00:10:DC:D9:83:2B
```

## Parte opcional 1

Modificar la función anterior para que, si tras enviar una petición ARP no se recibe su respuesta ARP en 2 segundos, se retransmita la petición ARP. Si pasados otros 3 segundos tampoco se recibe respuesta, la función debe devolver un código de error.

## Parte opcional 2

Implementar una cache ARP que almacene las últimas direcciones solicitadas. La cache ARP debe tener una capacidad limitada y las entradas ARP deben expirar pasado cierto tiempo. Se valorará la política de reemplazo de entradas en la cache y la renovación eficiente de las entradas expiradas.

## Parte opcional 3

Implementar un servidor ARP (`arp_server.c`) que responda a las peticiones ARP enviadas por el cliente desarrollado, así como por cualquier otro cliente ARP estándar.

## Referencias

[1] “RFC826: *An Ethernet Address Resolution Protocol*”, Noviembre 1982.