

1) a- Pasos:

Primer paquete que envía: consultas dns:

1. El cliente consulta primero a su DNS Local sobre la IP (reg A) del host www.redes.unlp.edu.ar.
2. Si este no puede obtenerla de su caché entonces el DNS Local consulta de forma iterativa al Root-Server más cercano, que le responde (de forma iterativa) el NS de .ar.
4. DNS Local consulta al NS de .ar, que responde con el NS de .edu.ar.
5. DNS Local consulta al NS de .edu.ar, que responde con el NS de unlp.edu.ar.
6. DNS Local consulta al NS de unlp.edu.ar, que responde con el NS **autoritativo** de redes.unlp.edu.ar.
7. DNS Local consulta por el registro **A** al NS **autoritativo** de redes.unlp.edu.ar, que le responderá con la IP de www.redes.unlp.edu.ar.
8. El DNS Server cacheará la respuesta y le responde al resolver del cliente con la IP de www.redes.unlp.edu.ar (el resolver también la cacheará).
9. Una vez que el cliente tiene la IP, realiza el requerimiento HTTP.

b- No, la respuesta que recibe el cliente NO es autoritativa porque proviene de su servidor DNS Local, que obtiene respuestas de otros servidores DNS (actúa como un resolver no autoritativo, que realiza una serie de consultas iterativas).

El servidor autoritativo en este caso es el NS redes.unlp.edu.ar, que contiene la información precisa del dominio www.redes.unlp.edu.ar.

La consulta al DNS Local es recursiva, y la respuesta es recursiva (ya que realiza todas las consultas iterativas).

c- i. Se podría haber usado HTTP/1.0 o /1.1, pero como en las cabeceras se encuentra el Connection: keep-alive, me inclino a pensar que es HTTP/1.0, ya que el keep-alive es el comportamiento por defecto de HTTP/1.1.

ii. GET contacto.html HTTP/1.0

Host: www.redes.unlp.edu.ar -> no va el https -> se negocia el protocolo en la capa de transp

d- i. Se usará el que tenga mayor prioridad (= < n°), o sea, el 5.

ii. No, para enviar no es necesario tener un MX, para recibir sí.

e- Este código de respuesta significa que la URI del recurso solicitado ha sido cambiada. Dicha nueva URI es devuelta en la respuesta.

2) Dominio principal: redes.unlp.edu.ar

redes.unlp.edu.ar. IN SOA ns1.redes.edu.ar. admin.redes.unlp.edu.ar.

redes.unlp.edu.ar IN NS ns1.redes.edu.ar.

redes.unlp.edu.ar IN NS ns2.redes.edu.ar.

primerafecha.redes.unlp.edu.ar IN NS ns1.primera.redes.edu.ar.

ns1.redes.edu.ar IN A ip_ns1

ns2.redes.edu.ar IN A ip_ns2

ns1.primera.redes.edu.ar IN A ip_ns1_primera

www.redes.unlp.edu.ar IN A ip_servidor_web

mail.redes.unlp.edu.ar IN A ip_servidor_correo

primerafecha.redes.unlp.edu.ar IN A ip_primera

redes.unlp.edu.ar IN MX 5 mail.redes.unlp.edu.ar

3) a- 10 -> 200 -----> [SYN] Seq = 0

200 -> 10 -----> [SYN, ACK] Seq = 0 Ack = 1

10 -> 200 -----> [ACK] Seq = 1 Ack = 1

10 -> 200 -----> [ACK] Seq = 1 Ack = 1 Len = 653

200 -> 10 -----> [ACK] Seq = 1 Ack = 654

200 -> 10 -----> [ACK] Seq = 1 Ack = 654 Len = 1460

200 -> 10 -----> [ACK] Seq = 1461 Ack = 654 Len = 1460

b- i. 2000 - (1176 + 44) = 780.

ii. Lo que hubiese pasado sería que el servidor no debería poder enviar datos, esto gracias al mecanismo de control de flujo (evita que el remitente envíe más datos de los que el receptor puede manejar).

4) Red C: 223 (+2) hosts -> necesito 8 bits = 256 hosts

192.168. 1100 1000 . 0000 0000 = .200.0

255.255. 1111 1110 . 0000 0000 -> máscara /23

255.255. 1111 1111 . 0000 0000 -> nueva máscara /24

Queda 1 bit para asignar a subredes.

192.168. 1100 1000 . 0000 0000 – 192.168.200.0/24 – Asignada a la Red C

192.168. 1100 1001 . 0000 0000 – 192.168.201.0/24 – Libre.

Red D: 110 (+2) hosts -> necesito 7 bits = 128 hosts

192.168. 1100 1001 . 0000 0000 = .201.0

255.255. 1111 1111 . 0000 0000 -> máscara /24

255.255. 1111 1111 . 1000 0000 -> nueva máscara /25

Queda 1 bit para asignar a subredes.

192.168. 1100 1001 . 0000 0000 – 192.168.201.0/25 – Asignada a la Red D

192.168. 1100 1001 . 1000 0000 – 192.168.201.128/25 – Libre.

Red B: 62 (+2) hosts -> necesito 6 bits = 64 hosts

192.168. 1100 1001 . 1000 0000 = .201.128

255.255. 1111 1111 . 1000 0000 -> máscara /25

255.255. 1111 1111 . 1100 0000 -> nueva máscara /26

Queda 1 bit para asignar a subredes.

192.168. 1100 1001 . 1000 0000 – 192.168.201.128/26 – Asignada a la Red B

192.168. 1100 1001 . 1100 0000 – 192.168.201.192/26 – Libre.

Red A: 27 (+2) hosts -> necesito 5 bits = 32 hosts

192.168. 1100 1001 . 1100 0000 = .201.192

255.255. 1111 1111 . 1100 0000 -> máscara /26

255.255. 1111 1111 . 1110 0000 -> nueva máscara /27

Queda 1 bit para asignar a subredes.

192.168. 1100 1001 . 1100 0000 – 192.168.201.192/27 – Asignada a la Red A

192.168. 1100 1001 . 1110 0000 – 192.168.201.224/27 – Asignada a la Red E

REDES ENTRE ROUTERS: usar 10.10.20.0/27

Red RA-RBorder, RA-RB, RB-RD, RBorder-RD, RD-RC: 2 (+2) hosts -> necesito 2 bits = 4 hosts

10.10.20. 0000 0000 = .0

255.255.255. 1110 0000 -> máscara /27

255.255.255. 1111 1100 -> nueva máscara /30

Quedan 3 bits para asignar a subredes.

10.10.20. 0000 0000 – 10.10.20.0/30 – Asignada a la Red RA-RBorder

10.10.20. 0000 0100 – 10.10.20.4/30 – Asignada a la Red RA-RB

10.10.20. 0000 1000 – 10.10.20.8/30 – Asignada a la Red RBorder-RD

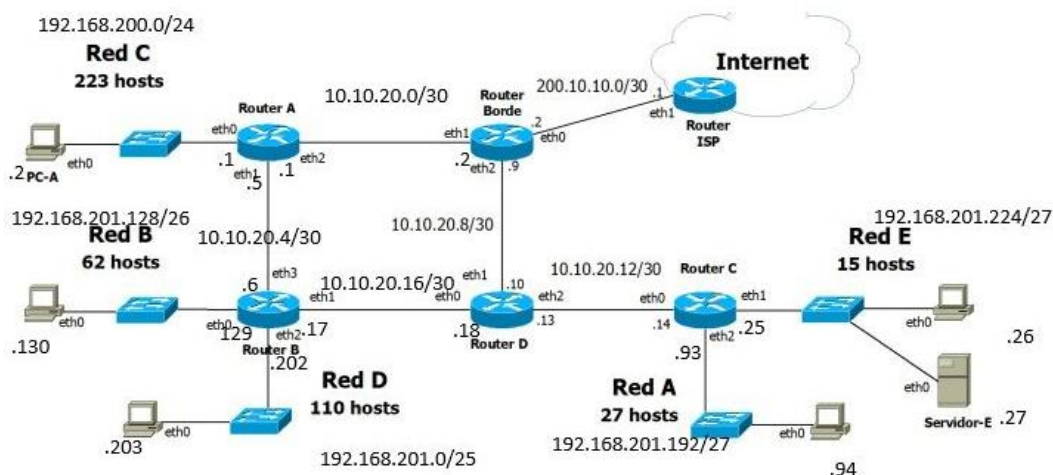
10.10.20. 0000 1100 – 10.10.20.12/30 – Asignada a la Red RD-RC

10.10.20. 0001 0000 – 10.10.20.16/30 – Asignada a la Red RB-RD

10.10.20. 0001 0100 – 10.10.20.20/30 – Libre

10.10.20. 0001 1000 – 10.10.20.24/30 – Libre

10.10.20. 0001 1100 – 10.10.20.28/30 – Libre



b- TABLA DE RUTEO Router B:

Destination	Mask	Gateway	Iface
0.0.0.0	/0	10.10.20.5	eth3
10.10.20.4	/30	-	eth3
10.10.20.16	/30	-	eth1
192.168.201.128	/26	-	eth0
192.168.201.0	/25	-	eth2
10.10.20.8	/30	10.10.20.18	eth1
10.10.20.12	/30	10.10.20.18	eth1
192.168.201.192	/27	10.10.20.18	eth1
192.168.201.224	/27	10.10.20.18	eth1
192.168.200.0	/24	10.10.20.5	eth3
10.10.20.0	/30	10.10.20.5	eth3
200.10.10.0	/30	10.10.20.5	eth3

Puedo **sumarizar** 10.10.20.8/30 y 10.10.20.12/30, ya que son contiguas y tienen el mismo Gateway y la misma Interface en:

10.10.20. 0000 0100 = .8/30

10.10.20. 0000 0110 = .12/30

10.10.20. 0000 0100 = **.8/29**

Puedo **sumarizar** 192.168.201.192/27 y 192.168.201.224/27, ya que son contiguas y tienen el mismo Gateway y la misma Interface en:

192.168.201. 1100 0000 = .192/27

192.168.201. 1110 0000 = .224/27

192.168.201. 1100 0000 = **.192/26**

La tabla de ruteo me termina quedando:

Destination	Mask	Gateway	Iface
0.0.0.0	/0	10.10.20.5	eth3
10.10.20.4	/30	-	eth3
10.10.20.16	/30	-	eth1
192.168.201.128	/26	-	eth0
192.168.201.0	/25	-	eth2
10.10.20.8	/29	10.10.20.18	eth1
192.168.201.192	/26	10.10.20.18	eth1
192.168.200.0	/24	10.10.20.5	eth3
10.10.20.0	/30	10.10.20.5	eth3
200.10.10.0	/30	10.10.20.5	eth3

5) a- Los paquetes van a entrar en un loop infinito y simplemente se esperara que haya ttl para descartar el paquete.

b- Depende de cómo estén configurados los demás routers -> sigue pudiendo usar el DEFAULT GATEWAY (0.0.0.0).

c- TCP: ICMP "Destination Unreachable". No recibe un RST porque el RST lo tendría que enviar el destino, pero NO llegó. El router es el que contesta, y este NO tiene capa de transporte.

UDP: ICMP red inalcanzable.

6) a- Enviar un paquete ICMP Echo Request al puerto 22 puede hacerse por medio de TCP o UDP. **F** -> los paquetes ICMP (Internet Control Message Protocol) son independientes de los protocolos TCP y UDP, ya que pertenecen a la capa de red y no utilizan puertos. Enviar un ICMP Echo Request no implica usar un puerto (como el 22), ya que los puertos son específicos de los protocolos de la capa de transporte (TCP/UDP). Por lo tanto, no es posible enviar un paquete ICMP al puerto 22.

b- Los registros MX al tener prioridad distinta, generan balanceo de carga. **F** -> los registros MX con diferentes prioridades se utilizan para especificar el orden en el que los servidores de correo deben ser contactados.

c- Si hay un servidor web atendiendo solicitudes por HTTP 1.1 para diferentes dominios, serán identificadas por medio del GET y el nombre de la página. **F** -> el GET identifica el recurso solicitado, pero el dominio es especificado por la cabecera Host, no por el GET.

d- En IPv6 la dirección 4002::1981::7/64 es válida. **F** -> no puede haber 2 :: en la misma dirección.